แผนบริหารการสอนประจำวิชา

รายวิชา โปรแกรมภาษาจาวาเบื้องต้น (Basic Java Programming) จำนวนหน่วยกิต-ชั่วโมง เวลาเรียน

รหัสวิชา วศ 5111306 (EN 5111306) 3 หน่วยกิต (2-2-5) 64 ชั่วโมง / ภาคการศึกษา

คำอธิบายรายวิชา

พื้นฐานการโปรแกรมภาษาจาวา การโปรแกรมเชิงวัตถุด้วยจาวา การทำงานของแอพเพ็ต การ ทำงานเทรด และแอนนิเมชัน การติดต่อกับผู้ใช้เบื้องต้น และการติดต่อกับผู้ใช้ขั้นสูง การประยุกต์แบบ ผู้ใช้คนเดียว แนะนำการโปรแกรมบนเครือข่ายด้วยจาวา

วัตถุประสงค์ทั่วไป

- 1. เพื่อให้มีความรู้เกี่ยวกับหลักการโปรแกรมคอมพิวเตอร์
- 2. เพื่อให้สามารถคิด และคำนวณทางโปรแกรมภาษาจาวาได้
- 3. เพื่อให้เกิดทักษะในการทดลองด้านการโปรแกรมภาษาจาวา
- 4. เพื่อให้สามารถเขียนบันทึกผลการทดลองได้
- 5. เพื่อให้มีทัศนคติที่ดีต่อการเรียนวิชาทางการโปรแกรมภาษาจาวา
- เพื่อให้สามารถนำความรู้ทางโปรแกรมภาษาจาวามาประยุกต์ใช้ประโยชน์ใน ชีวิตประจำวัน

เนื้อหา

8 ชั่วโบง แผนบริหารการสอนประจำบทที่ 1 ับทที่ 1 หลักการเขียนโปรแกรมคอมพิวเตอร์ การเขียนโปรแกรมคอมพิวเตอร์ กระบวนการทำงานการเขียนโปรแกรมแบบโครงสร้าง ความเป็นมาภาษาจาวา กระบวนการทำงานของโปรแกรมภาษาจาวา องค์ประกอบของเทคโนโลยีจาวา แพลตฟอร์มของจาวา การติดตั้ง และเรียกใช้งานโปรแกรมภาษาจาวา กำหนดเส้นทางการทำงาน (Path) ให้กับระบบปฏิบัติการวินโดว์ การติดตั้งโปรแกรม Edit Plus หลักการเขียนโปรแกรมภาษาจาวา สรุป แบบฝึกหัด เอกสารอ้างอิง

แผนบริหารการสอนประจำบทที่ 2 4 ชั่วโบง บทที่ 2 การเขียนโปรแกรมเชิงวัตถุ คลาส วัตถุ หรือ ออบเจกต์ การปิดบังข้อมูล การสืบทอดคุณสมบัติ นามธรรม โพลิมอฟิซึม พื้นฐานการเขียนโปรแกรมภาษาจาวา ส่วนประกอบของโปรแกรมภาษาจาวา สรุป แบบฝึกหัด เอกสารอ้างอิง 4 ชั่วโมง แผนบริหารการสอนประจำบทที่ 3 บทที่ 3 ตัวแปรและตัวดำเนินการ ตัวแปร ชนิดข้อมูล ตัวดำเนินการ ตัวดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม ตัวดำเนินการทางคณิตศาสตร์เลขทศนิยม ตัวดำเนินการทางคณิตศาสตร์ร่วมกับการกำหนดค่า ตัวดำเนินการทางคณิตศาสตร์การเพิ่มค่าและการลดค่า สรุป แบบฝึกหัด เอกสารอ้างอิง แผนบริหารการสอนประจำบทที่ 4 8 ชั่วโมง บทที่ 4 ประโยคควบคุมการทำงาน ประโยคตามลำดับ ประโยคการตัดสินใจ ประโยค if-else การเขียนรูปแบบลัด if ประโยค switch-case ประโยคการวนลูป ประโยค for ประโยค while ประโยค do/while ประโยคการควบคุมอื่นๆเพิ่มเติม ประโยค break

ประโยค continue ประโยค return สรุป แบบฝึกหัด เอกสารอ้างอิง 8 ชั่วโมง แผนบริหารการสอนประจำบทที่ 5 บทที่ 5 คลาสและอ็อบเจกต์ คลาส class name การสร้างคลาสจากคลาสเดิม แพ็กเกจ อ็อบเจกต์ ประกาศ instance class ประกาศ instance สร้าง อ็อบเจกต์ การเรียกใช้งานแอตทริบิวต์ การส่งข้อมูลด้วยค่าของตัวแปร การส่งข้อมูลด้วยการอ้างอิงของวัตถุ คอนสตรัคเตอร์ สแตติคดาต้าเมมเบอร์ สแตติคเมธอดเมมเบอร์ สรุป แบบฝึกหัด เอกสารอ้างอิง 8 ชั่วโมง แผนบริหารการสอนประจำบทที่ 6 **บทที่ 6** คลาสและอ็อบเจกต์ วงจรชีวิตแอพเพ็ต เมธอด init() เมธอด start() เมธอด stop() เมธอด destroy() เมธอด paint() การเขียนโค๊ดแอพเพ็ตแทรกในภาษาไฮเปอร์เท็กต์มาร์คอัพ (HTML) แอพเพ็ตกับกราฟิก แอพเพ็ตกับเสียง แอพเพ็ตกับภาพ แอพเพ็ตกับเทรดสำหรับภาพเคลื่อนไหว การสร้างเทรด

การสร้างเทรดสำหรับภาพเคลื่อนไหว สรุป แบบฝึกหัด เอกสารอ้างอิง ผนบริหารการสอนประจำบทที่ 7 4 ชั่วโมง **บทที่ 7** ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) ด้วย AWT ความสัมพันธ์ระหว่าง AWT และ Swing การโปรแกรม GUI ด้วย Swing และ AWT การจัดวางองค์ประกอบ FlowLayout BorderLayout GridLayout สร้าง Basic Event Handling สรุป แบบฝึกหัด เอกสารอ้างอิง แผนบริหารการสอนประจำบทที่ 8 4 ชั่วโมง **บทที่ 8** ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) ด้วย AWT ไฟล์สตรีม ไบท์สตรีม การใช้คำสั่ง try ... catch / finally คาร์แร็คเตอร์สตรีม สรุป แบบฝึกหัด เอกสารอ้างอิง แผนบริหารการสอนประจำบทที่ 9 8 ชั่วโมง **บทที่ 9** การประยุกต์การติดต่อฐานข้อมูลกับเครือข่าย เน็ตบีนส์ (Netbeans) มายเอ็สคิวแอล (MySql) การเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล การกำหนดคุณลักษณะมายเอ็สคิวแอล สรุป แบบฝึกหัด เอกสารอ้างอิง 8 ชั่วโมง แผนบริหารการสอนประจำบทที่ 10 **บทที่ 10** การพัฒนาฐานข้อมูลด้วยจาวา สร้างและทำการเชื่อมต่อฐานข้อมูลให้พร้อมใช้งาน สร้างตารางให้กับฐานข้อมูล

การสั่งงานตารางข้อมูล การสั่งการงานฐานข้อมูลด้วย Script แนะนำการโปรแกรมบนเครือข่ายด้วยจาวา สรุป แบบฝึกหัด เอกสารอ้างอิง

ผนวก

การทดลองที่ 1 การทดลองที่ 2 การทดลองที่ 3 การทดลองที่ 4 การทดลองที่ 5 การทดลองที่ 6 การทดลองที่ 7 การทดลองที่ 8 การทดลองที่ 9

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

1.1 สอนแบบบรรยาย โดยการนำเสนอข้อมูลด้วยสื่ออิเล็กทรอนิกส์โปรแกรมพาว-เวอร์พอยท์ (power point program) ประกอบการบรรยาย และตั้งคำถามระหว่างบรรยายเพื่อให้ ผู้เรียนได้คิดและตอบคำถามในประเด็นที่สำคัญ

1.2 สอนแบบอภิปราย โดยแบ่งผู้เรียนออกเป็นกลุ่ม ๆ และมอบหมายให้แต่ละกลุ่ม
 ร่วมกันแสดงความคิดเห็น นำเสนอการอภิปรายของแต่ละกลุ่ม แล้วผู้สอนเป็นผู้สรุปผลการอภิปราย
 ให้เข้าใจตรงตามวัตถุประสงค์ของเนื้อหา

1.3 สอนแบบสาธิตการใช้โปรแกรมคอมพิวเตอร์ เครื่องคอมพิวเตอร์เป็นเครื่องมือ
 โดยผู้สอนแสดงขั้นตอนให้ผู้เรียนดูพร้อมอธิบายลำดับขั้นตอนในการปฏิบัติ จากนั้นให้ผู้เรียนได้ทดลอง
 ปฏิบัติ โดยแบ่งผู้เรียนออกเป็นกลุ่ม ๆ

1.4 สอนโดยให้ค้นคว้าหาความรู้ด้วยตนเอง โดยผู้สอนมอบหมายเนื้อหาในบทเรียน ให้ผู้เรียนสืบค้นข้อมูลจากตำรา เอกสาร สื่ออิเล็กทรอนิกส์ ทำรายงานและนำเสนอในห้องเรียน

 1.5 การทำรายงาน เป็นกิจกรรมกลุ่มที่ผู้สอนนำกระตุ้นให้ผู้เรียน นำปัญหาที่พบใน ชีวิตประจำวันมาเป็นโจทย์ เป็นการเรียนการสอนโดยยึดการเรียนรู้โดยใช้ปัญหาเป็นฐาน (PBL: Problem-Base Learning) ในการเขียนโปรแกรมคอมพิวเตอร์เพื่อใช้ในการแก้ปัญหา โดยให้ผู้เรียนได้ มีส่วนร่วมในการเลือกหัวข้อรายงาน และสมาชิกในการทำงาน เป็นการเรียนการสอนที่ยึดผู้เรียนเป็น สำคัญ

<u>สื่อการเรียนการสอน</u>

- 1. แฟ้มข้อมูลชนิดโปรแกรมพาวเวอร์พอยท์
- 2. ภาพสื่อผสมภาพนิ่ง หรือภาพเคลื่อนไหว จากระบบออนไลน์
- 3. เครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติที่สามารถใช้ระบบปฏิบัติการดอส(DOS Prompt)
- 4. ตัวโปรแกรมเขียนโค๊ด (Edit Plus)
- 5. ตัวโปรแกรมคอมไพเลอร์ภาษาจาวา (Java Compiler)
- 6. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 7. ระบบอินเทอร์เน็ตและหน่วยความจำภายนอก(External Hard disk)

การวัดและการประเมินผล

- 1. การวัดผล
 - 1.1 คะแนนระหว่างภาคการศึกษา

ร้อยละ 70 คะแนน

ร้อยละ 30 คะแนน

ร้อยละ 20 คะแนน

ร้อยละ 10 คะแนน

ร้อยละ 10 คะแนน

- 1.1.1 สอบกลางภาค ร้อยละ 20 คะแนน
 1.1.2 ความตั้งใจในการเข้าเรียน ร้อยละ 5 คะแนน
 1.1.3 แบบฝึกหัด ร้อยละ 5 คะแนน
- 1.1.4 ใบงาน
- 1.1.5 ศึกษาค้นคว้ารายงาน
- 1.1.6 นำเสนอรายงาน
- 1.2 คะแนนสอบปลายภาคการศึกษา

2. การประเมินผล

ได้คะแนนระหว่างร้อยละ 80-1	00 ได้ระดับ A	ดีเยี่ยม (Excellent)
ได้คะแนนระหว่างร้อยละ 75-7	9 ได้ระดับ B+	ดีมาก (Very Good)
ได้คะแนนระหว่างร้อยละ 70-7	4 ได้ระดับ B	ดี (Good)
ได้คะแนนระหว่างร้อยละ 65-6	9 ได้ระดับ C+	ดีพอใช้ (Fairy Good)
ได้คะแนนระหว่างร้อยละ 60-6	4 ได้ระดับ C	พอใช้ (Fairy)
ได้คะแนนระหว่างร้อยละ 55-5	9 ได้ระดับ D+	อ่อน (Poor)
ได้คะแนนระหว่างร้อยละ 50-5	4 ได้ระดับ D	อ่อนมาก (Very Poor)
ได้คะแนนระหว่างร้อยละ 0-49	ได้ระดับ E	ตก (Fail)

แผนบริหารการสอนประจำบทที่ 1

หัวข้อเนื้อหา

หลักการเขียนโปรแกรมคอมพิวเตอร์

การเขียนโปรแกรมคอมพิวเตอร์

- กระบวนการทำงานการเขียนโปรแกรมแบบโครงสร้าง
- ความเป็นมาภาษาจาวา
 - กระบวนการทำงานของโปรแกรมภาษาจาวา
 - องค์ประกอบของเทคโนโลยีจาวา
 - แพลตฟอร์มของจาวา
- การติดตั้ง และเรียกใช้งานโปรแกรมภาษาจาวา
 - กำหนดเส้นทางการทำงาน (Path) ให้กับระบบปฏิบัติการวินโดว์
- การติดตั้งโปรแกรม Edit Plus
- หลักการเขียนโปรแกรมภาษาจาวา

วัตถุประสงค์เชิงพฤติกรุรม

เมื่อผู้เรียนได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. เกิดทักษะการติดตั้งโปรแกรมภาษาจาวาและโปรแกรม Edit Plus
- 2. อธิบายความแตกต่างหรือสัมพันธ์ระหว่างโปรแกรมภาษาจาวามีกับภาษาแบบโครงสร้างได้
- 3. บอกขั้นตอนการทำงานของโปรแกรมภาษาจาวาได้
- 4. เขียนโปรแกรมภาษาจาวาตามรูปแบบภาษาได้

วิธีการสอนและกิจกรรมการเรียนการสอน

- 1. วิธีสอน
 - 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อหลักการเขียนโปรแกรมคอมพิวเตอร์
- 1.2 ใช้วิธีสอนแบบปฏิบัติการเรื่องดาวน์โหลดตัวโปรแกรมและการติดตั้งโปรแกรม ภาษาจาวา และโปรแกรมอิดิเตอร์
 - 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดสมบัติโปรแกรมที่ได้ทำการติดตั้ง
- 2. กิจกรรมการเรียนการสอน

2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการดาวน์โหลดตัวโปรแกรม และการ ติดตั้งโปรแกรม

- 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
- 2.3 ทำแบบฝึกหัดเกี่ยวกับเรื่องหลักการเขียนโปรแกรมคอมพิวเตอร์

สื่อการเรียนการสอน

- 8. แฟ้มข้อมูลการนำเสนอ ของบทที่ 1
- 9. ระบบอินเทอร์เน็ต สำหรับดาวน์โหลด
- 10. เครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติที่สามารถเข้าถึงระบบปฏิบัติการดอส (DOS Prompt)
- 11. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 12. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 13. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 14. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- 1. สังเกตจากการตอบคำถามและตั้งคำถามของนักศึกษา
- 2. สังเกตจากความกระตือรือร้นของนักศึกษา
- 3. สังเกตจากการฝึกและผลงานภาคปฏิบัติ
- 4. ตรวจแบบฝึกหัด

บทที่ 1 หลักการเขียนโปรแกรมคอมพิวเตอร์ (Principle of Computer Programming)

การเขียนโปรแกรมคอมพิวเตอร์

การเขียนโปรแกรมคอมพิวเตอร์ (Computer Programming) มีแนวคิดมาจากมนุษย์มีความ ต้องการที่นำงานที่เป็นลักษณะงานประจำ งานที่เป็นลักษณะงานซ้ำๆ ที่มีขั้นตอนทำงานที่แน่นอน ้มีระบบการตัดสินใจที่ชัดเจนแน่นอน มาให้ระบบคอมพิวเตอร์ทำงานแทนมนุษย์ ตามวัตถุประสงค์ใน การทำงานของผู้คิดค้นโปรแกรมคอมพิวเตอร์ ดังนั้นการเขียนโปรแกรมคอมพิวเตอร์จึงเป็นพื้นฐาน ที่จำเป็นต่อการทำงานของระบบคอมพิวเตอร์ ชนิดของโปรแกรมคอมพิวเตอร์สามารถแบ่งตามระดับ ้ความลึกของการเขียนโปรแกรมควบคุมอุปกรณ์คอมพิวเตอร์ได้เป็น 2 ชนิด คือ **ชนิดแรก**เป็นโปรแกรม ระบบปฏิบัติการ (Operating System) ทำหน้าที่ในการควบคุมอุปกรณ์พื้นฐานที่เป็นส่วนประกอบ ้คอมพิวเตอร์ทั้ง 4 หน่วย(Unit) ดังนี้คือ 1) หน่วยอินพุท (Input Unit) เช่น แป้นป้อนข้อมูล (Key Board) เมาส์ (Mouse) เป็นต้น 2) หน่วยเอาท์พุท (Output Unit) เช่น จอแสดงผล (Monitor) ลำโพง (Speaker) เป็นต้น 3) หน่วยความจำ (Memory Unit) เช่น หน่วยความจำชั่วคราว (RAM) หน่วยความจำภายนอก (External Memory) เช่น ฮาร์ดดิสก์ (Hard Disk) และหน่วยสุดท้ายหน่วย 4) หน่วยประมวลผลกลาง (CPU: Central Processing Unit) ด้วยเพราะเทคโนโลยีของอุปกรณ์พื้นฐาน ของคอมพิวเตอร์มีการแข่งขันสูงส่งผลทำให้โปรแกรมที่ทำหน้าที่ในการควบคุมอุปกรณ์คอมพิวเตอร์นี้ ต้องมีการพัฒนาอย่างต่อเนื่อง ดังนั้นโปรแกรมระบบปฏิบัติการจึงเป็นโปรแกรมที่สำคัญต่อการ เจริญก้าวหน้าของบริษัทที่ทำหน้าที่ในการผลิตเครื่องคอมพิวเตอร์ โปรแกรมชนิดนี้จึงเป็นเทคโนโลยีปิด นักเขียนโปรแกรมคอมพิวเตอร์จึงมีระดับการเข้าถึงในการกำหนดค่าการควบคุมอุปกรณ์ต่างๆ มีอย่าง จำกัดตามที่บริษัทที่พัฒนาโปรแกรมระบบปฏิบัติการอนุญาตให้เข้าถึงได้เท่านั้น โปรแกรมดังกล่าว เช่น โปรแกรมระบบปฏิบัติการวินโดว์ (Windows Operating System) โปรแกรมระบบปฏิบัติการ ลินุกส์ (Linux Operating System) เป็นต้น **ชนิดที่สอง**เป็นโปรแกรมภาษาคอมพิวเตอร์ ที่นักพัฒนา ้โปรแกรมคอมพิวเตอร์ สามารถเข้าถึงในการเขียนโปรกรมควบคุมอุปกรณ์ต่างๆ ได้อย่างกว้างขว้าง ้โปรแกรมชนิดภาษาคอมพิวเตอร์นี้ จึงเป็นโปรแกรมที่นักพัฒนาโปรแกรมสามารถเรียนรู้และสร้างสรรค์ ้งานได้อย่างต่อเนื่องตลอดเวลา ดังนั้นภาษาคอมพิวเตอร์จึงสามารถแบ่งชนิดตามลักษณะการมองปัญหา ของโจทย์ที่ต้องการให้การเขียนโปรแกรมทำการแก้ปัญหาได้เป็น 2 แบบ คือ แบบแรกเป็นการเขียน โปรแกรมแบบโครงสร้าง (Structure Programming) ตัวอย่างภาษาที่ใช้เขียนโปรแกรม เช่น โปรแกรม ภาษาซี (C Language Program) โปรแกรมภาษาปาสคาล (Pascal Language Program) โปรแกรม ภาษาเบสิค (Basic Language Program) เป็นต้น ส่วนแบบที่สองเป็นการเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming : OOP) ตัวอย่างภาษาที่ใช้เขียนโปรแกรม เช่น โปรแกรม ภาษาซีพลัสพลัส (C++ Language Program) โปรแกรมภาษาซีชาร์ป (C# Language Program) โปรแกรมพีเอสพี (PHP Program) โปรแกรมวิชวลเบสิค (Visual Basic Program) โปรแกรมภาษาจาวา (Java Language Program) เป็นต้น

กระบวนการทำงานการเขียนโปรแกรมแบบโครงสร้าง

ในการพัฒนาโปรแกรมแบบโครงสร้างหรือการเขียนโปรแกรมแบบโครงสร้าง ซึ่งมีกระบวนการ ในการสร้างคือ ตัวอย่างเช่น การโปรแกรมภาษาซี เริ่มต้นด้วยการจากทำการสร้างโปรแกรม (source program หรือ source code) ซึ่งต้องเขียนตามลักษณะโครงสร้างของภาษาซี ทำการบันทึก (save หรือ save as) เป็นไฟล์นามสกุล .c นำโค๊ดที่ได้ทำการเขียนทำการคอมไพล์ ทำการตรวจสอบ ข้อผิดพลาดว่ามีหรือไม่ ถ้ามีให้กลับไปแก้ไขที่โปรแกรม แต่ถ้าไม่มีข้อผิดพลาดตัวระบบภาษาซีจะสร้าง ไฟล์นามสกุล .obj แล้วทำการเชื่อมต่อไฟล์จากไฟล์ไรบารีและระบบจะทำการสร้างไฟล์นามสกุล .obj และทำการตรวจสอบความผิดพลาดในการเชื่อมต่อ ถ้าไม่มีข้อผิดพลาดจะได้ไฟล์นามสกุล .exe ซึ่งเป็น ไฟล์ที่พร้อมนำไปสั่งงานให้แสดงผลการทำงานตามที่กำหนดของโปรแกรม ไฟล์ที่เป็นนามสกุล .exe นี้ ไม่สามารถทำการแก้ไขได้ด้วยเพราะเป็นไฟล์ที่เป็นรหัสให้เครื่องคอมพิวเตอร์ทำงาน ดังนั้นถ้าต้องการ ปรับปรุงหรือแก้ไขโปรแกรมผู้เขียนโปรแกรมต้องมีไฟล์ที่เป็นนามสกุล .c มาเข้ากระบวนการประมวลผล ภาษาซีใหม่ตั้งแต่เริ่มต้น แสดงในภาพที่ 1.1



ภาพที่ 1.1 แสดงกระบวนการประมวลผลของภาษาซี

ความเป็นมาภาษาจาวา

การพัฒนาภาษาจาวาถูกพัฒนาโดย ซัน ไมโครซิสเต็มท์ บจก. (sun microsystem co., ltd) โดยขณะที่ทีมงานของบริษัททำการพัฒนาภาษามักนิยมดื่มกาแฟกันเป็นประจำ และกาแฟที่นิยมดื่ม มีชื่อว่า จาวา ทีมพัฒนาภาษาจาวาต้นแบบจึงตั้งชื่อภาษาที่พัฒนาว่า จาวา จึงเป็นที่มาเทคโนโลยีของ บริษัท ซัน ไมโครซิสเต็มท์ จำกัด เป็นต้นมา ซึ่งทำให้ บริษัท ซัน ฯลฯ เป็นผู้กำกับทิศทางไม่ให้มีการนำ ภาษาจาวาไปดัดแปลงประยุกต์ใช้ในทางที่เบี่ยงเบนออกจากจุดประสงค์เดิม การกำหนดทิศทางโดย ซัน เป็นไปเพื่อให้เกิดความชัดเจนในทิศทางการพัฒนาภาษาจาวา โดยภาษาจาวามีคำสั่งพื้นฐานคล้าย ภาษาซี พลัส พลัสเป็นอย่างมาก ทำให้นักเขียนโปรแกรมภาษาซี พลัส พลัส ที่มีอยู่มากที่สุดแล้ว ในขณะนั้นสามารถสร้างความคุ้นเคยได้อย่างรวดเร็ว จึงทำให้ภาษาจาวาเป็นที่ยอมรับอย่างกว้างขวาง อย่างรวดเร็ว และได้รับการจัดเป็นภาษาคอมพิวเตอร์เชิงวัตถุ เช่นเดียวกับภาษาซี พลัส พลัส ด้วย แต่สิ่งที่ทั้งสองภาษาต่างกัน คือ โปรแกรมภาษาจาวาต้องเขียนเป็นแบบเชิงวัตถุเท่านั้น ในขณะที่ภาษาซี พลัส พลัส สามารถเขียนได้ทั้งแบบเชิงวัตถุ และแบบโครงสร้างก็ได้ ที่เป็นเช่นนี้ได้เนื่องจากภาษาซี พลัส พลัส มต้นกำเนิดมาจากภาษาซี ซึ่งเป็นภาษาแบบโครงสร้าง ดังนั้นภาษาซี พลัส พลัส จึงต้องสนับสนุน การเขียนโปรแกรมแบบโครงสร้างด้วยเพื่อให้เข้ากันได้กับภาษาซี อย่างไรก็ตามปัจจุบันโปรแกรม ประยุกต์ล้วนแต่เขียนด้วยภาษาเชิงวัตถุทั้งสิ้น

กระบวนการทำงานของโปรแกรมภาษาจาวา

การทำงานโปรแกรมภาษาจาวา การสร้างไฟล์ต้นฉบับ (source file) หรือ โค๊ดต้นฉบับ(source code) แล้วทำการบันทึกไฟล์ดังกล่าวโดยบันทึก นามสกุล .java เก็บไว้ใน folder ที่ต้องการ แล้วทำ การ compiler ด้วย java compiler ในระบบปฏิบัติการ DOS เพื่อเปลี่ยนคำสั่งภาษาจาวาเป็นภาษา เฉพาะอย่างหนึ่งเพื่อให้จาวาเวอร์ชัวร์แมทชีนเข้าใจ ภาษาเฉพาะที่จาวาเวอร์ชัวน์แมทชีนเข้าใจนี้เรา เรียกว่าจาวาไบต์โค๊ด เป็นไฟล์ที่มีนามสกุล .class แล้วนำไฟล์ ดังกล่าวไปรันบนเครื่องคอมพิวเตอร์ที่มี จาวาเวอร์ชัวร์แมทชีน สรุปขั้นตอนการพัฒนาโปรแกรมด้วยภาษาจาได้แสดงดังภาพ 1.2



ภาพที่ 1.2 กระบวนการพัฒนาภาษาจาวา ที่มา (About the Java Technology, 2014)

ด้วยจาวาเวอร์ชัวน์แมทชีน มีความสามารถนำมาใช้กับระบบปฏิบัติการที่แตกต่างกันได้ โดยใช้คลาสที่คอมไพล์ตัวเดียวกัน นำมาทำงานบนระบบของไมโครซอฟท์วินโดว์ (Microsoft Windows) ระบบโซลาริส (Solaris) หรือ ระบบยูนิกส์ (Unix) รวมทั้งระบบของแม็ค (Mac) ได้ ดังแสดง ในภาพที่ 1.3



ภาพที่ 1.3 แสดงระบบจาวาเวอร์ชัวน์แมทชีนที่สามารถใช้ได้ทุกระบบปฏิบัติการ ที่มา (About the Java Technology, 2014)

องค์ประกอบของเทคโนโลยีจาวา

เทคโนโลยีภาษาจาวามืองค์ประกอบที่สำคัญในการทำงาน ประกอบด้วย 3 ส่วน คือ

1. จาวาเวอร์ชัวน์แมทชีน (Java Virtual Machine: JVM) เป็นส่วนที่ทำหน้าที่เป็น ตัวอินเตอร์ - พรีเตอร์ (interpreter) คือ จะทำการแปลจาวาไบต์โค๊ด ให้เป็นภาษาที่เครื่องเข้าใจ ซึ่งจา วาไบต์โค๊ดจะสามารถรันได้หลายแพลตฟอร์ม ถ้าแพลตฟอร์มนั้นมี จาวาเวอร์ชัวน์แมทชีน

2. จาวารันไทล์อินวิโรเมนต์ (Java Runtime Environment: JRE) เป็นส่วนที่ใช้ ในการรันโปรแกรมภาษาจาวา เป็นส่วนที่ใช้ในการรันโปรแกรม โดยจะทำงาน 3 ขั้นตอน ดังนี้ ขั้นตอน แรกโหลดไบต์โค็ดโดยใช้ Class loader คือการโหลดคลาสทุกคลาสที่เกี่ยวข้องในการรันโปรแกรม ขั้นตอนที่สองตรวจสอบไบต์โค๊ดโดยใช้การตรวจสอบไบท์โค๊ด (Byte code Verifier) คือการตรวจสอบ ว่าโปรแกรมจะต้องไม่มีคำสั่งที่ทำให้เกิดความผิดพลาดกับระบบ เช่น การแปลงข้อมูลที่ผิดพลาด หรือมี การแทรกแซงเข้าสู่ระบบภายในเป็นต้น ขั้นตอนที่สุดท้าย รันไบต์โค๊ด โดยใช้ตัวแปลรันไทล์ (Runtime Interpreter)

3. ชุดพัฒนาภาษาจาวา (Java 2 Software Developer Kit: J2SDK) เป็นชุด พัฒนาโปรแกรมภาษาจาวา ประกอบไปด้วยโปรแกรมต่างๆ แต่ไม่มีโปรแกรมที่ทำหน้าที่สร้างโค๊ด (Editor) รวมอยู่ด้วย แต่มีโปรแกรมคอมไพเลอร์ (javac.exe) และโปรแกรมอินเตอร์พรีเตอร์ (java.exe)

แพลตฟอร์มของจาวา (Platform of Java)

แพลตฟอร์มของจาวา หมายถึงลักษณะการนำภาษาจาวาไปสร้างเป็นโปรแกรมเพื่อ วัตถุประสงค์ใด วัตถุประสงค์หนึ่ง บนการประมวลผลของคอมพิวเตอร์ อาจเป็นการทำการประมวลผล โดยไม่มีการเชื่อมต่อกับเครื่องคอมพิวเตอร์เครื่องอื่นๆ หรือต้องการนำโปรแกรมภาษาจาวาไปใช้กับ ระบบเครือข่ายอินเทอร์เน็ต เป็นต้น ซึ่งแพลตฟอร์มของจาวาจึงมีการแบ่งรูปแบบเป็น 3 รูปแบบ คือ

1. แพลตฟอร์มจาวามาตรฐาน (Standard Edition: Java SE) เป็นแพลตฟอร์มที่ ใช้ในการพัฒนาโปรแกรมภาษาจาวากับเครื่องพีซีทั่วไป ซึ่งสามารถแยกย่อยออกเป็น 2 แบบ คือ Java Application เป็นโปรแกรมที่ใช้พัฒนาโปรแกรมประยุกต์ทั่วไป และ Java Applet เป็นโปรแกรมจาวา ที่ใช้พัฒนาโปรแกรมเพื่อรันบนเว็บบราวเซอร์

 แพลตฟอร์มจาวาอินเตอร์ไพรส์ (Enterprise Edition: Java EE) เป็น แพลตฟอร์มที่มุ่งเน้นในการพัฒนาโปรแกรมเครือข่าย สำหรับใช้งานในองค์กร โดยใช้โปรแกรม Application Server หรือ Web Server ใช้สำหรับพัฒนาโปรแกรมที่มีขนาดใหญ่

3. แพลตฟอร์มจาวาไมโคร (Micro Edition: Java ME) เป็นแพลตฟอร์มที่ใช้พัฒนา โปรแกรมเพื่อใช้งานกับอุปกรณ์อิเล็กทรอนิกส์ ที่มีทรัพยากรจำกัด เช่น โทรศัพท์มือถือ เป็นต้น

การติดตั้ง และเรียกใช้งานโปรแกรมภาษาจาวา

การพัฒนาโปรแกรมด้วยภาษาจาวา สำหรับคอมพิวเตอร์ส่วนบุคคล เครื่องคอมพิวเตอร์ จำเป็นต้องอาศัยชุดพัฒนา J2SDK แต่ด้วยชุดพัฒนาไม่มีโปรแกรมที่ทำหน้าที่สร้าง source code ซึ่งใน ระบบปฏิบัติการวินโดวส์สามารถใช้โปรแกรมโน๊ตแพด (notepad) ทำการสร้างโค๊ดภาษาจาวา เพื่อทำ การเก็บโค๊ดด้วยการบันทึกเป็นนามสกุล และส่วนขยาย .java ซึ่งชุดพัฒนา J2SDK ส่วนตัวคอมไพล์และ ตัวรันโปรแกรม สามารถทำการดาวน์โหลดและติดตั้ง J2SDK ลงบนเครื่องคอมพิวเตอร์ระบบปฏิบัติการ วินโดวส์ได้ที่ <u>http://java.sun.com/javase/downloads/index.jsp</u> (ภาพที่ 1.4)



ภาพที่ 1.4 แสดงแหล่งดาวน์โหลดชุดพัฒนาโปรแกรมจาวา ที่มา (java download, 2014)

การติดตั้งชุดพัฒนาโปรแกรมภาษาจาวา

การติดตั้งสามารถทำได้การแตกแฟ้มข้อมูลที่ได้ทำการดาวน์โหลดมา (ภาพที่ 1.5)



ภาพที่ 1.5 แสดงชุดพัฒนาโปรแกรมภาษาจาวา SDK

ทำการเลือกที่ jsk-6u14-windows-i586.exe เพื่อทำการติดตั้งเครื่องมือในการพัฒนา โปรแกรมด้วยภาษาจาวา และทำการเลือก Accept (ภาพที่ 1.6)

Java(TM)	SE Development Kit 6 Update 14 - License	x
نې از ا Java	License Agreement Please read the following license agreement carefully.	IN.
Sun Micro for the JA	systems, Inc. Binary Code License Agreement VA SE DEVELOPMENT KIT (JDK), VERSION 6	Â
SUN MICH SOFTWA THAT YOU CODE LIC (COLLEC CAREFUL ACCEPT SELECTIN AGREEM TERMS. S	ROSYSTEMS, INC. ("SUN") IS WILLING TO LICENSE THE RE IDENTIFIED BELOW TO YOU ONLY UPON THE CONDITION J ACCEPT ALL OF THE TERMS CONTAINED IN THIS BINARY ENSE AGREEMENT AND SUPPLEMENTAL LICENSE TERMS TIVELY "AGREEMENT"). PLEASE READ THE AGREEMENT .LY. BY DOWNLOADING OR INSTALLING THIS SOFTWARE, YOU THE TERMS OF THE AGREEMENT. INDICATE ACCEPTANCE BY NG THE "ACCEPT" BUTTON AT THE BOTTOM OF THE ENT. IF YOU ARE NOT WILLING TO BE BOUND BY ALL THE SELECT THE "DECLINE" BUTTON AT THE BOTTOM OF THE	4
	Decline Accept >	

ภาพที่ 1.6 แสดงชุดพัฒนาโปรแกรมภาษาจาวา ลิขสิทธิ์โดย Sun Microsystems บจม.

ตัวชุดพัฒนาโปรแกรมภาษาจาวา จะให้ผู้ติดตั้งโปรแกรมกำหนดลักษณะโปรแกรมที่ต้องการ (Custom Setup) ถ้าต้องการกำหนดตามข้อแนะนำให้ทำการเลือก Next ตามภาพที่ 1.7

ava c	ustom Setup Select the program features you v	want installed.
lect optional fe tallation by us	eatures to install from the list belo ing the Add/Remove Programs ut Development Tools Demos and Samples Source Code	w. You can change your choice of features af lity in the Control Panel Feature Description Java(TM) SE Development Kit 6 Update 14, including private JRE 6 Update 14. This will require 300MB on your hard drive.
tall to: Program Files	(x86)\Java\jdk1.6.0_14\	Change

ภาพที่ 1.7 แสดงการกำหนดค่าตำแหน่งที่ติดตั้งโปรแกรม

ตัวชุดพัฒนาจะทำการติดตั้งโปรแกรมตามที่กำหนดจากภาพ 1.7 โดยแสดงสถานะ การทำงาน ว่าสำเร็จเป็นจำนวนเท่าใด แสดงในภาพที่ 1.8



ภาพที่ 1.8 แสดงผลการติดตั้งโปรแกรมภาษาจาวา

<text>

เมื่อทำการติดตั้งสำเร็จสมบูรณ์เรียบร้อย แสดงผลภาพที่ 1.9 ทำการเลือก Finish



กำหนดเส้นทางการทำงาน (Path) ให้กับระบบปฏิบัติการวินโดว์

เมื่อทำการติดตั้งชุดพัฒนาโปรแกรมภาษาจาวา สิ่งที่สำคัญอีกประการหนึ่ง คือการกำหนด เส้นทางในการทำงานให้กับตัวโปรแกรมภาษาจาวาที่ต้องนำตัวโค๊ดจาวาที่ได้ทำการบันทึกในส่วนใดส่วน หนึ่งของหน่วยความจำบนเองคอมพิวเตอร์ที่มีระบบปฏิบัติการวินโดว์ได้โดยไม่มีปัญหาในการค้นหาโค๊ด จาวาที่ผู้เขียนโปรแกรมได้ทำการบันทึกได้อิสระ ส่งผมทำให้การทำงานโค๊ดจาวาได้อย่างเสร็จสมบูรณ์ การกำหนดเส้นทางสามารถทำได้โดยการคลิกเมาส์ขวาที่ไอคอน My Computer บนเดสก์ท็อป แล้ว เลือก Properties จากนั้นเลือกแท็ป Advanced และทำการเลือก Environment Variables ซึ่ง วินโดว์แต่ละรุ่นจะมีหน้าจอในส่วนนี้ไม่เหมือนกัน ตัวอย่างในภาพ 1.10 เป็นระบบปฏิบัติวินโดว์ 7

System Properties
Computer Name Hardware System Protection Remote
You must be logged on as an Administrator to make most of these changes.
Performance
Visual effects, processor scheduling, memory usage, and virtual memory
Settings
User Profiles
Desktop settings related to your logon
Settings
Startup and Recovery
System startup, system failure, and debugging information
Settings
Environment Variables
OK Cancel Apply

ภาพที่ 1.10 แสดงหน้าต่างของ My computer กำหนดเส้นทาง

ทำการกำหนดเส้นทางของตัวโปรแกรมภาษาจาวาให้โปรแกรมระบบปฏิบัติการวินโดว์ โดยทำ การบันทึกหรือสำเนาที่อยู่ของตัวโปรแกรมภาษาจาวา แล้วนำมาวางต่อในส่วนของ Path ด้วยการเลือก Edit เพื่อทำการแก้ไขเพิ่มเส้นทางเดิมที่ตัวระบบปฏิบัติได้กำหนดให้กับโปรแกรมอื่นๆ ที่ได้ทำการติดตั้ง มาก่อนหน้านี้ (ห้ามทำการลบ Path ของเดิมออกเด็ดขาด เพราะอาจทำให้ตัวโปรแกรมบางโปรแกรม ทำงานไม่ได้ ด้วยหาเส้นทางไม่พบ) แล้วทำการวางโดยก่อนวางที่อยู่ของโปรแกรมภาษาจาวาต้องวาง หน้าเครื่องหมาย เซมิโคร่อน (;) เสมอแสดงในภาพที่ 1.11

	bles	23
User variables fo	r Administrator	Edit User Variable
Variable	Value	
MGT_PHIGIN_ Path	P C:\Program Files (x86)\Foxit Software\ C:\Program Files\Intel\WiFi\bin\:C:\Prog	E Variable name: Path
TEMP	%USERPROFILE%\AppData\Local\Temp	Variable Value / Variable Value / Variable Value / Variable Value / Va
TMP	%USERPROFILE%\AppData\Local\Temp	
System variables	, ~	
Variable	Value	·
Variable asl.log	Value Destination=file	
Variable asl.log ComSpec	Value Destination=file C:\Windows\system32\cmd.exe	
Variable asl.log ComSpec FP_NO_HOST_	Value Destination=file C:\Windows\system32\cmd.exe C NO	
Variable asl.log ComSpec FP_NO_HOST_ NUMBER_OF_F	Value Destination=file C:\Windows\system32\cmd.exe C NO ' 4	

ภาพที่ 1.11 แสดงหน้าต่าง การกำหนด Path ของระบบปฏิบัติการวินโดว์ 7

การทำงานของโปรแกรมภาษาจาวาบนระบบดอส (DOS)

การสร้างซอร์สโค๊ดเบื้องต้นด้วยตัวอิดิเตอร์ที่ติดมากับระบบปฏิบัติการวินโดว์ คือ โน๊ตแพด (Notepad) ดังตัวอย่างซอร์สโค๊ด ภาพที่ 1.12 บันทึกชื่อ first เป็นนามสกุล .java (first.java)



ภาพที่ 1.12 แสดงซอร์สโค้ดชื่อ first.java บน Notepad

จากภาพที่ 1.12 สิ่งสำคัญในการจัดเก็บซอร์สโค๊ด ชื่อของแฟ้มข้อมูลต้องเป็นชื่อเดียวกันกับ ชื่อของคลาสในซอร์สโค๊ด ในที่นี้ชื่อ public class first ดังนั้นชื่อที่ใช้ในการจัดเก็บต้องเป็น first ด้วย เช่นกัน (first ไม่เท่ากับ First ด้วยเพราะรูปแบบของตัวอักษร f กับ F คนละตัวกัน)

เมื่อต้องการทำการคอมไพล์ซอร์สโค๊ด first.java ด้วยการคำสั่ง javac.exe บนระบบดอสทำได้ โดยออกสู่ Command Prompt แสดงภาพที่ 1.13 และภาพที่ 1.14 แสดงหน้าต่างระบบ DOS



ภาพที่ 1.13 แสดงการเข้าสู่ระบบ Dos ด้วย Command Prompt

and Administrator: Command Prompt	
Microsoft Windows [Version 6.1.7601] Copyright (c) 2009 Microsoft Corporation. All rights reserve C:\Users\Administrato.>cd C:\d: D>>cd java D:\Java/dir Volume in drive D has no label. Volume Serial Number is 547P-192D	ed.
Directory of D:\Java 10/12/2013 01:51 PM 〈DIR〉 10/12/2013 01:51 PM 〈DIR〉 10/12/2013 12:30 PM 122 first.java 1 File(s) 122 bytes 2 Dir(s) 201,061,535,744 bytes free	
D:\Java>_	•

ภาพที่ 1.14 แสดงการใช้คำสั่งพื้นฐานระบบ DOS

จากภาพที่ 1.14 เป็นส่วนของระบบ DOS ที่เปิดโอกาสให้สามารถสั่งการควบคุมขั้นพื้นฐานได้ เช่น คำสั่ง cd\ หมายถึง สั่งให้ระบบออกไดเร็คทรอรีปัจจุบัน (c:\Users\Administrator>) สู่ไดร์ฟซี (c:\>) คำสั่ง d: หมายถึง สั่งให้ระบบย้ายไปไดร์ฟดี (d:\>) คำสั่ง cd java หมายถึง การย้ายไปที่ไดเร็ค ทรอรีชื่อ java คำสั่ง dir หมายถึง การสั่งให้แสดงชื่อแฟ้มข้อมูลที่อยู่ในไดเร็คทรอรีปัจจุบัน (c:\java>) มีแฟ้มข้อมูลใดบ้าง ซึ่งพบแฟ้มข้อมูลชื่อ first.java สร้างขึ้นเมื่อ 12 ตุลาคม 2013 เวลา 12.30 PM ขนาด 122 ไบท์



ภาพที่ 1.15 แสดงการทำงานของโปรแกรมภาษาจาวา

จากภาพที่ 1.15 สามารถสั่งทำการให้ระบบ dos ทำการคอมไพล์ซอร์สโค๊ดแฟ้มข้อมูล first.java ด้วยคำสั่ง javac first.java ได้โดยไม่มีข้อผิดพลาด เพราะบรรทัดหลังจากสั่งคอมไพล์ไม่มีการ แจ้งข้อผิดพลาดใดๆ ซึ่งตัวคอมไพเลอร์จาวาทำการสร้างไบท์โค๊ดจาวา ที่มีชื่อเดียวกันแต่นามสกุลเป็น .class (first.class)

การสั่งให้โปรแกรมจาวาทำงาน (Run) ด้วยจาวาเวอร์ชัวน์แมทชีน ทำการอินเตอร์พรีเตอร์ ให้กับระบบปฏิบัติการด้วยการแสดงข้อความว่า Hello Word!!! ด้วยคำสั่ง java first โดย java เป็น คำสั่งให้ทำงาน (Run) ส่วน first เป็นไบท์โค็ดนามสกุล .class จึงไม่ต้องกำหนดนามสกุล แสดงในภาพที่ 1.15

การติดตั้งโปรแกรม Edit Plus

โปรแกรม Edit Plus สามารถดาวน์โหลดได้ที่ http://www.editplus.com/download.html สามารถนำมาใช้ได้ฟรี โดยจำกัดเวลาใช้งาน ภาพที่ 1.16



ภาพที่ 1.16 แสดงแหล่งดาวน์โหลด Edit Plus

เมื่อได้ตัวโปรแกรม Edit Plus ทำการติดตั้งโปรแกรมลงบนเครื่องคอมพิวเตอร์แล้วลำดับถัดมา ทำการสร้างเครื่องมือสำหรับการใช้งานของนักพัฒนาโปรแกรมบนโปรแกรม EditPlus โดยสร้างได้โดย การเลือกรายการ Tools เลือก Configure User Tools (ภาพที่ 1.17)



ภาพที่ 1.17 แสดงการกำหนดเครื่องมือ Edit Plus

ทำการสร้างเครื่องมืออำนวยความสะดวก ในการให้มีคำสั่งการ Compiler กับคำสั่ง Run ของ ตัวโปรแกรมภาษาจาวา ด้วย Group Name กำหนดชื่อเป็น JaVa และทำการเลือก ok (ภาพ 1.18)

Roname1 - Ed	Preferences	X	
	Categories	Groups and tool items:	
Directory Clipte	General Fonts Colors Print	Group 2 Group Name Group Name Add Tool >> Remove	+ <u>8</u> -
 Skul EdPlan BasicJav 2,2555 Inclass (III) InputTest.java 	File Settings & syntax Templates Tools User tools Keyboard Spell checker Toolbar Layout	Action: None	
Java (*.java)		Output Pattern Save: None ✓ OK X Cancel	h. I

ภาพที่ 1.18 แสดงการกำหนดชื่อกลุ่มเป็น JaVa

ทำการเพิ่มเครื่องมือในกลุ่ม JaVa ด้วยการเลือก Add Tool>> และเลือก Program ถัดมาทำ การกำหนดเมนูย่อยของกลุ่มเครื่องมือ JaVa ด้วยการพิมพ์ชื่อว่า CompilerJava ในช่อง Menu Text ส่วน Command: ให้ทำการเลือก javac.exe ที่อยู่ในโฟลเดอร์ของโปรแกรมภาษาจาวา ตามภาพที่ 1.19 ส่วน Argument: กำหนดให้เป็น File Name ด้วยเพราะในการสั่ง Compiler ต้องกำหนดชื่อ พร้อมนามสกุล (first.java) เพื่อทำการคอมไพล์ ส่วน Action ให้เลือก Capture Output

• A A • • • • • • • • • • • • • • • • •	↓↓ (ﷺ ::::::::::::::::::::::::::::::::::	สิงสิงbCcD สงสิงbCcD สงสิงbCcD File Path ชื่อเรื่อง ชื่อเรื่อง ตัวเช่น ตัวเช่น File Directory
Directory (D-1 Roname1 - Ed File Edit	Clipte General Fonts Preferences	Group 2 Group Name Add Tool >> File Name Without Extension File Extension File Path Without Drive Name File Dir Without Drive Name Project name
Directory Clipte D: D: Skul EdPlan Basiclav 2,2555 Inclass	Categories General Fonts Colors Print File Settings & syntax Templates Tools User tools	Groups and tool items: JaVa Group Name Group Name Group Name Add Tool >> Current Line Number Current Selection Up Down Up Down Up Down Current Vord Current Line Text Compiler Java
< <u>III</u> InputTest.java	Keyboard Spell checker Toolbar Layout	Menu text: Compile/Java Command: n Files (x86)/Java\jdk1.6.0_14\bin\java.c.exe Argument: Initial directory: Action: None Output Patterm Save:

ภาพที่ 1.19 แสดงการสร้างเครื่องมือ CompilerJava

กำหนดเครื่องมือให้กับกลุ่ม JaVa ด้วยเมนู RunJava ในช่อง Menu Text: ส่วน Command: ให้ทำการเปิดคำสั่ง java.exe ที่อยู่ในตัวโปรแกรมภาษาจาวา ในช่อง Argument: ให้ทำการเลือก Files Name Without Extension ในส่วนของ Action: ให้เลือก Capture output จากนั้นทำการเลือก Apply และ OK (ภาพที่ 1.20)

6	ย่อหน้า	โด ได้กษณะ	File Name
Roname1 - Ed		Action: None	File Name Without Extension File Extension File Extension File Path Without Drive Name
File Edit File Edit Directory Clipte [D:] D:\ D:\ Skul EdPlan Basiclav 2.2555 Inclass InputTest.java	Categories General Fonts Colors Print File Settings & syntax Templates Tools User tools User tools Keyboard Spell checker Toolbar Layout	Groups and tool items: JaVa Group Name Compitediate Rundava Add Tool >> Remove Up Down 2 ³ Icon Menu text: Rundava Command: Im Files (x86)/Java/jdk1.6.0,14/bin/java.exe Argument: Initial directory: \$(FileDir) Columnation of the second s	File Dir Without Drive Name Project name Project Base Directory Application Directory Current Line Number Current Column Number Current Selection Current Vord Current Line Text Copy Selection Window List Prompt For Arguments
Java (*.java) → Nonan For Help, press F1		Output Pattern Save: All open files VOR Cancel Apply ?	

ภาพที่ 1.20 แสดงการสร้างเมนูย่อย RunJava บนกลุ่มเครื่องมือ JaVa

หลักการเขียนโปรแกรมภาษาจาวา

การเขียนโปรแกรมจาวาตามที่กล่าวมาข้างต้น ตัวซอร์สโค็ดสามารถทำการเขียนด้วยอิดิเตอร์ใด ก็ได้ Notepad หรือ Edit Plus สิ่งสำคัญในการบันทึกชื่อของซอร์สโค๊ด ต้องเป็นชื่อเดียวกับชื่อ class ด้วยเพราะ เมื่อซอร์สโค๊ดถูกคอมไพล์ตัวคอมไพเลอร์จะทำการสร้างคลาสที่เป็นไบท์โค๊ดขึ้นมา โดยใช้ชื่อ ของแฟ้มข้อมูลที่บันทึกจากนามสกุล .java มาเป็นนามสกุล .class

การสร้างโค็ดภาษาจาวา มีความคล้ายกับการสร้างโค็ดให้กับภาษาซี หรือ ซีพลัสพลัส คือต้องมี เมธอด main() เป็นจุดเริ่มต้นการทำงาน และต้องถูกกำหนดให้อยู่ในคลาสใดคลาสหนึ่ง ของโค็ดจาวา เมธอด main() จะต้องปรากฏได้เพียงครั้งเดียวในหนึ่งโค็ดโปรแกรมภาษาจาวา ด้วยเพราะเป็นเมธอด หลัก และถ้าโค๊ดใดไม่มีส่วนของเมธอด main() โค๊ดนั้นจะไม่สมบูรณ์ หมายถึงไม่สามารทำงานได้ เมธอด main () ตามกฎของโปรแกรมภาษาจาวา ต้องมีส่วนประกอบประโยคดังนี้ public static void main(String args[]){ } การกำหนดให้เป็นเมธอด main() เป็น public เพื่อต้องการให้โค๊ดที่เขียน สามารถเข้าถึงหรือเรียกใช้งานได้ขณะเมื่อสังให้ทำการการรัน (interpreter) ส่วนการกำหนดให้คลาส เป็น static ด้วย มีผลทำให้เมธอด main() สามารถเรียกใช้ได้โดยไม่ต้องสร้าง instance ของคลาส

ดังนั้นทุกครั้งที่มีการสร้างโค๊ดโปรแกรมภาษาจาวา ต้องมีเมธอด main() ที่ประกอบไปด้วย public static void และส่วน Argument ต้องเป็น String Array เสมอ ซึ่งจะมีการกำหนดค่าในอาร์เรย์ อาร์กูเมนต์ หรือไม่กำหนดก็สามารถสร้างโค๊ดได้ ตัวอย่างโปรแกรมที่ 1.1 ชื่อ first.java ตัวอย่างโปรแกรมที่ 1.1 เป็นตัวอย่างแรก ซึ่งมีโครงสร้างพื้นฐานที่จำเป็นในการสร้างโปรแกรมครั้งแรก ดังนั้นการพิมพ์คำสั่งทุกคำสั่งต้องถูกต้อง โดยเฉพาะอย่างยิ่งตัวอักษรพิมพ์เล็ก พิมพ์ใหญ่ ต้องถูกต้อง ทั้งหมด บันทึกเป็นชื่อ first.java (โดยให้ทำการสำเนาชื่อแฟ้มข้อมูล ที่อยู่หลัง public class เป็นชื่อ ของแฟ้มข้อมูลที่ต้องการบันทึก เป็นการลดปัญหาการผิดพลาดจากการใช้ชื่อที่ประกอบด้วยตัวอักษร พิมพ์เล็กพิมพ์ใหญ่)

```
public class first{
public static void main(String args[]){
    System.out.println("Hello World!!!");
    }
}
```

การสั่งให้ทำการคอมไพล์โค๊ดโดย เลือกที่ Tools เลือกรายการย่อย CompilerJava (หรือ เลือกใช้ซอร์ตคีย์ ด้วยการกด Ctrl ค้างไว้พร้อมกดเลข 1) ภาพที่ 1.21 เมื่อไม่มีข้อผิดพลาดใดๆ จะ แสดงผลการคอมไพล์ด้วย Output completed



ภาพที่ 1.21 แสดงการสั่งคอมไพล์โค๊ดโปรแกรมภาษาจาวา

การสั่งให้โปรแกรมทำงานสามารถทำได้โดยการเลือกที่ Tools เลือกรายการย่อย RunJava (หรือเลือกใช้ซอร์ตคีย์ ด้วยการกด Ctrl ค้างไว้พร้อมกดเลข 2) ภาพที่ 1.22 เมื่อไม่มีข้อผิดพลาดใดๆ จะแสดงผลการทำงานด้วยข้อความ Hello Word!!!

🧠 D:\Java\first.java - EditF	Plus						- 0 <mark>- X -</mark>
🧪 File Edit View Se	earch Document Project	Тоо	ls Browser ZC	Window	Help	_	_ & ×
i 🗂 📂 🖬 🕼 🕒 🖶) 🍄 🖸 🔏 🖣 📋 🗙 🛉	1	Preferences			188 🗸 🗉 🖻	
Directory Cliptext	+1+		Set Directories			6	+7
[D:] •	1 public class first 2 public static void	ABC	Spell Check				
D:\	3 { 4 System out pr	\$	Sort				=
🔰 Java	5 }	Σ	Sum				
	6 }		Text to Speech		•		
			Record		+		
	RunJava		Configure User To	ools			
	Hello World!!!		User Tool Group		+		
	Output completed (0 s	13	CompilerJava	C	trl+1		
Java (.java) 🔻		2	RunJava 🧲	Ci	trl+2		
🔲 🔅 first.java 🖉		_					
Runs this user-defined tool	ln 1		col1 7	70	PC	ANSI	

ภาพที่ 1.22 แสดงผลการทำงานด้วย RunJava

ตัวอย่างโปรแกรมภาษาจาวา ที่มีการกำหนดรับค่าให้กับอาร์เรย์อาร์กูเมนต์ ผ่านทางการสั่ง Run ดังตัวอย่างโปรแกรมที่ 1.2 ชื่อ FirstArgs.java

```
public class FirstArgs{
public static void main(String args[])
{
    System.out.println("Created By>>>"+args[0]);
    System.out.print("Welcome to java, ");
    System.out.println("Thank you for looking. ");
    System.out.print("-..-");
  }
}
```

จากโปรแกรม 1.2 ส่วนของคำสั่ง System.out.println("Created By>>>"+args[0]); เป็นการเรียก เมธอด println อยู่ในคลาส out ในแพ็คเก็ต System มาทำการแสดงผลออกทางจอหน้า โดย เมธอด println เป็นการสั่งให้ข้อความหรือค่าของตัวแปรที่อยู่ในอาร์กูเมนต์ลิสต์ (Created By>>>) ออกสู่ หน้าจอ และในส่วนของอาร์กูเมนต์ลิสต์ ค่าที่สอง (args[0]) เป็นการรับค่าจากแป้นพิมพ์ขณะทำการสั่ง ให้ Run โปรแกรม ดังภาพที่ 1.22 ได้ยกตัวอย่างค่าที่ส่งเป็นข้อความ SKUL ซึ่งสามารถเปลี่ยนค่าได้ จากเมธอด println มีผลการแสดงค่าในอาร์กูเมนต์ลิสต์เสร็จแล้วระบบจะทำการขึ้นบรรทัดใหม่รอคำสั่ง ถัดไป ส่วนเมธอด print ของ System.out.print("Welcome to java, "); เป็นการสั่งให้แสดงค่า Welcome to java, โดยแสดงเสร็จแล้ว เคอร์เซอร์ยังคงอยู่ที่บรรทัดเดิม เมื่อมีคำสั่งต่อมาเป็นเรียกใช้ เมธอด println จะทำการแสดงข้อความหรือค่าของตัวแปรที่อยู่ในอาร์กูเมนต์ลิสต์ออกทางหน้าจอ แสดงภาพที่ 1.23



ภาพที่ 1.23 แสดงการสั่ง Run โดยการกำหนดอาร์กูเมนต์

สรุป

การเขียนโปรแกรมคอมพิวเตอร์สำหรับนักพัฒนาโปรแกรม สามารถทำการสร้างงานด้วยการ เขียนโปรแกรมได้ 2 แบบ คือ แบบการเขียนโปรแกรมแบบโครงสร้าง กับแบบการเขียนโปรแกรมเชิง วัตถุ ซึ่งการเขียนโปรแกรมเชิงวัตถุ สามารถสร้างงานได้ด้วยโปรแกรมแก้ไขใดๆ แล้วทำการบันทึกเป็น แฟ้มข้อมูลนามสกุล .java เพื่อนำมาสร้างคลาสได้ด้วยตัวแปลภาษาจาวา (javac.exe) ส่งผลทำการให้ ได้แฟ้มข้อมูลที่เป็นคลาสมาหนึ่งแฟ้มนามสกุล .class จากนั้นเมื่อต้องการนำงานที่พัฒนาขึ้นมาไปใช้งาน สามารถทำได้โดยการนำตัวจาวาเวอร์ชัวน์แมทซีนที่ทำหน้ารัน (java.exe) มาสั่งให้ทำงานตามที่ นักพัฒนาเขียนได้สร้างไว้

แบบฝึกหัด

- 1. การเขียนโปรแกรมภาษาจาวามีความแตกต่าง หรือสัมพันธ์กับภาษาแบบโครงสร้างอย่างไร
- 2. จงอธิบายระหว่าง jacac.exe กับ java.exe ว่าหน้าที่อะไร
- 3. จงอธิบายความแตกต่างระหว่างโปรแกรม Notepad กับ EditPlus
- 4. จงอธิบายความสำคัญของการแปลภาษา (Compiler) ว่าสำคัญอย่างไร
- 5. จงอธิบายความสำคัญของการสั่งทำงาน (Run) ว่าสำคัญอย่างไร
- 6. จงประโยชน์ของการกำหนดเส้นทาง(path) หลังจากทำการติดตั้งโปรแกรมแล้ว
- 7. การออกจากระบบปฏิบัติการวินโดว์สู่ระบบดอสทำได้อย่างไร
- 8. การเขียนคำอธิบายในโปรแกรมภาษาจาวาทำได้ด้วยเครื่องหมายใดได้บ้าง
- 9. คำสั่งภาษาจาวาสามารถเขียนด้วยตัวอักษรตัวใหญ่ได้หรือไม่เพาะเหตุใด
- 10. ในการสั่งจบคำสั่งของโปรแกรมภาษาจาวาในแต่ละบรรทัด ใช้เครื่องหมายใด

.....

เอกสารอ้างอิง

กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพฯ: สำนักพิมพ์ เคทีพี.

รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพา: ซัคเซค มีเดีย บจก.

วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

. (2545). Java Programming Volume II. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/

<u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2002). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Java download (2012). Retrieved August 28, 2012, from:

http://java.sun.com/javase/downloads/index.jsp

แผนบริหารการสอนประจำบทที่ 2

หัวข้อเนื้อหา

การเขียนโปรแกรมเชิงวัตถุ คลาส วัตถุ หรือ ออบเจกต์ การปิดบังข้อมูล การสืบทอดคุณสมบัติ นามธรรม โพลิมอฟิซึม พื้นฐานการเขียนโปรแกรมภาษาจาวา ส่วนประกอบของโปรแกรมภาษาจาวา อักษรหรือคำสั่งที่มองไม่เห็น คำอธิบายโค๊ด คำสงวน ชื่อเฉพาะ คำจัดกลุ่มหรือแยกกลุ่ม คำที่แสดงค่าคงที่ ตัวอย่างโปรแกรม สรุป

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 5. อธิบายการความคิดรวบยอดของแนวคิดการโปรแกรมเชิงวัตถุ
- 6. อธิบายขั้นตอนการเขียนโปรแกรมภาษาจาวา
- 7. มีทักษะการแปลโปรแกรมภาษาจาวาด้วยโปรแกรม Edit Plus
- 8. มีทักษะการเขียนโปรแกรมภาษาจาวา

วิธีการสอนและกิจกรรมการเรียนการสอน

- 1. วิธีสอน
 - 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อหลักการเขียนโปรแกรมเชิงวัตถุ

1.2 ใช้วิธีสอนแบบปฏิบัติการเรื่องพื้นฐานการเขียนโปรแกรมภาษาจาวา และส่วนประกอบของ โปรแกรมภาษาจาวา

1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์ และซอฟต์แวร์ที่ทำงานได้จริง

- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัดเกี่ยวกับเรื่องการเขียนโปรแกรมเชิงวัตถุ

สื่อการเรียนการสอน

- 15. แฟ้มข้อมูลการนำเสนอ ของบทที่ 2
- 16. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 17. เครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติที่สามารถเข้าถึงระบบปฏิบัติการดอส (DOS Prompt)
- 18. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 19. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 20. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 21. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- 2. การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรี้ยน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั่งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือร้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 2 การเขียนโปรแกรมเชิงวัตถุ (Object Oriented Programming: OOP)

การเขียนโปรแกรมเชิงวัตถุ เป็นการเขียนโปรแกรมโดยการกำหนดสิ่งที่เป็นปัจจัยในการเขียน โปรแกรม เป็นวัตถุ(object) ซึ่งวัตถุสามารถเป็นได้ทั้งรูปธรรม(สิ่งที่จำต้องได้) และนามธรรม(สิ่งที่จับ ต้องไม่ได้) ส่งผลทำให้การเขียนโปรแกรมเป็นเรื่องสามารถทำได้อย่างรวดเร็ว และการปรับปรุงแก้ไข โปรแกรมขนาดใหญ่ทำได้ง่ายและใช้เวลาน้อยลง ซึ่งการเขียนโปรแกรมเชิงวัตถุเป็นรูปแบบการเขียน โปรแกรมคอมพิวเตอร์ที่จำลองแนวคิด ว่าปัจจัยที่ป้อนเข้าตัวโปรแกรมเป็นวัตถุ ซึ่งสามารถนำมา ประกอบ ทำงานรวมกันได้ สามารถเพิ่มหน้าที่ของวัตถุได้ สามารถแลกเปลี่ยนข้อมูล แลกเปลี่ยนหน้าที่ และนำผลการประมวลผล ส่งต่อไปให้ วัตถุ อื่นๆ ได้

วัตถุ(อ็อบเจกต์)

วัตถุ หรือ อ็อบเจกต์ ที่ใช้ในการเขียนโปรแกรมเชิงวัตถุ จึงมีส่วนประกอบที่สำคัญ 2 ส่วน คือ ส่วนที่หนึ่งคุณลักษณะ (attribute) เป็นสิ่งที่บอกคุณลักษณะทั่วไปของวัตถุ หรือข้อมูลของวัตถุ (data) ส่วนที่สอง พฤติกรรม (behavior) เป็นสิ่งที่วัตถุสามารถกระทำได้หรือการกระทำของวัตถุ (method) การเขียนโปรแกรมเชิงวัตถุ ด้วยภาษาจาวามีสิ่งที่เป็นควรรู้พื้นฐานที่สำคัญในการเขียนโปรแกรม ดังนี้

คลาส (Class) คือการประกาศภาพรวมของกลุ่มวัตถุที่มีคุณลักษณะ (Data or Attribute) ที่เหมือนกัน มีพฤติกรรมที่ (Behavior or Method)เหมือนกัน ซึ่งคลาสเป็นโครงสร้างพื้นฐานของการ เขียนโปรแกรมเชิงวัตถุ เช่น คลาสรถยนต์เก๋ง มีคุณลักษณะประกอบด้วย ล้อ 4 ล้อ มีพวงมาลัย มีตัวรถ มีเครื่องยนต์ มีเครื่องปรับอากาศ ส่วนพฤติกรรม คือ สามารถวิ่งได้ เหลียวได้เดินหน้า ถอยหลัง หยุด เร่งความเร็วได้ เป็นต้น ดังนั้นคลาสเปรียบเสมือนพิมพ์เขียว (template) หรือต้นแบบของวัตถุนั่นเอง ดังนั้นหากจะต้องสร้างวัตถุ ลำดับแรกจำเป็นต้องสร้างคลาสขึ้นมาก่อน

อินสแตนซ์ (instance) ภาษาจาวามีการแบ่งตัวแปรเป็น 2 ชนิดคือ ชนิดไพรมิทีฟ (primitive type) คือตัวแปรที่เก็บข้อมูลพื้นฐานที่ไม่ใช่วัตถุ เช่น int, char, float, boolean เป็นต้น ส่วนตัวแปรอีกชนิดหนึ่งที่ใช้อ้างถึงข้อมูลในหน่วยความจำที่เป็นวัตถุ ในภาษาจาวาจะเรียกตัวแปรชนิดนี้ ว่าอ้างอิง (reference) ซึ่งการทำงานคล้ายๆกับพอยเตอร์ในภาษาซี คือเป็นตัวแปรที่อ้างไปยัง หน่วยความจำที่เก็บข้อมูลที่เป็นวัตถุนั้นๆ ตัวแปรพอยเตอร์จึงตัวแปรเป็นที่เก็บเฉพาะตำแหน่งของ ข้อมูล ถ้าจะสร้างข้อมูลใหม่ก็ต้องจัดการจองหน่วยความจำขึ้นมาแล้วให้พอยเตอร์นี้ไปซี้ เมื่อไม่ต้องการ ใช้ก็สามารถคืนหน่วยความจำได้ กล่าวคือตัวแปรชนิดอ้างอิงก็คือ อินสแตนซ์ในภาษาจาวานั่นเอง หมายความว่าอินสแตนซ์เป็นตัววัตถุที่ถูกสร้างขึ้นมาในหน่วยความจำด้วยคำสั่ง new เป็นการอ้างถึง ข้อมูลในหน่วยความจำที่เป็นวัตถุ ซึ่งทำงานคล้ายๆกับ พอยเตอร์ (pointer) ในภาษาซี คืออ้างไปยัง หน่วยความจำที่เก็บข้อมูลนั่นเอง ดังนั้นการสร้างวัตถุด้วยอินสแตนซ์จึงเป็นคำสั่งพื้นฐานของภาษาจาวา

การปิดบังข้อมูล (Encapsulation or Information hiding) เป็นวิธีการกำหนด สิทธิในการเข้าถึงข้อมูล หรือกำหนดการกระทำต่อกับวัตถุของ คลาสนั้นๆ ทำให้แน่ใจได้ว่าข้อมูลของ วัตถุนั้นจะถูกเปลี่ยนแปลงแก้ไขผ่านทางเมธอด (methods) หรือ คุณสมบัติ (properties) ที่ได้รับ อนุญาตเท่านั้น การปิดบังข้อมูลนั้นเป็นการป้องกันไม่ให้อ็อบเจกต์ภายนอกเข้าถึงข้อมูลได้อย่างอิสระ เป็นการสร้างความปลอดภัยให้กับข้อมูล เพราะข้อมูลจะถูกเข้าถึงได้จากได้รับอนุญาตให้เป็นผู้มีสิทธิ์ เท่านั้น

การสืบทอดคุณสมบัติ (Inheritance) เป็นการสืบทอดคุณสมบัติจากคลาสแม่ไปยัง คลาสลูก โดยที่คลาสลูกสามารถสืบทอดคุณสมบัติมาจากคลาสแม่ได้ ซึ่งเป็นคุณสมบัติของการนำอ็อบ เจกต์ กลับมาใช้ใหม่ (reuse) ประหยัดเวลาในการทำงาน ซึ่งเป็นจุดเด่นสำคัญ ของการเขียนโปรแกรม เชิงวัตถุดังนั้นในแต่ละแฟ้มข้อมูลหรือคลาส จะประกอบด้วยคลาสย่อย หรือซับคลาส (sub class) จะ ได้รับการถ่ายทอดคุณสมบัติต่างๆ มาจากคลาสหลัก (super class) หรือคลาสแม่

นามธรรม (Abstraction) เป็นการแสดงถึงคุณลักษณะและพฤติกรรมของอ็อบเจกต์ เท่าที่ จำเป็นต้องรับรู้และใช้งาน โดยซ่อนส่วนที่เหลือเอาไว้เพื่อไม่ให้เกิดความสับสน และสามารถนำมา เพิ่มเติมคุณสมบัติเพิ่มเต็มภายหลังได้

โพลิมอฟิซึม (Polymorphism) เป็นการกำหนดหน้าที่การทำงานที่มีหลายรูปแบบ หรือ กล่าวได้ว่า หนึ่งเมธอด สามารถทำงานได้หลายๆรูปแบบ โดยมีชื่อเมธอดเดียวกัน แต่สามารถทำงาน แตกต่างกันได้ ซึ่งโพลิมอฟิซึมสามารถแบ่งเป็น 2 แบบ คือ การโอเวอร์โหลดดิ้ง (Overloading) และ การโอเวอร์ไรดิ้ง (Overriding)

พื้นฐานการเขียนโปรแกรมภาษาจาวา

โครงสร้างของโปรแกรมภาษาจาวา ต้องมีประกอบด้วยคลาสอย่างน้อยหนึ่งคลาส โดยในคลาส จะประกอบด้วยสมาชิกที่เป็นข้อมูล(ตัวแปร) และหรือเมธอด (ในภาษาซีเรียกว่า ฟังก์ชัน: Function) จำนวนสมาชิกของคลาสจะอยู่ภายในเครื่องหมาย "{ }" ในการสร้างคลาสโดยภาษาจาวาสามารถสร้าง ขึ้นด้วยการพิมพ์คีย์เวิร์ด class ตามด้วยชื่อของคลาส ซึ่งชื่อคลาสที่ต้องเป็นชื่อเดียวกับแฟ้มข้อมูลของ โปรแกรมที่สร้างขึ้นโดยทำการบันทึกเป็นนามสกุล .java ด้วยเมื่อทำการคอมไพล์โปรแกรมภาษาจาวา แล้ว จะได้คลาสที่เป็นชื่อเดียวกับตัวโปรแกรมภาษาจาวาที่มีนามสกุล .class

ในการสร้างคลาสอย่างที่กล่าวข้างต้นว่า โปรแกรมภาษาจาวาในหนึ่งแฟ้มข้อมูลต้องมีคลาส อย่างน้อยหนึ่งคลาส แต่ไม่ได้หมายความว่ามีเพียงคลาสเดียวในหนึ่งแฟ้มข้อมูล ดังนั้นผู้เขียนโปรแกรม (Programmer) สามารถกำหนดให้หนึ่งคลาสของหนึ่งแฟ้มข้อมูลสามารถมีคลายย่อยได้ตามจำนวนที่ ต้องการได้ ตัวอย่างในภาพที่ 2.1

ดังนั้นเมื่อนำคลาสของโปรแกรมภาษาจาวามาใช้งาน ด้วยการอินเตอร์พรีเตอร์จาวา (Java Interpreter) นั้น ทำให้สามารถอ้างอิงคลาสให้มาทำงานได้อย่างอิสระ คือการเรียกใช้คลาสสามารถ เรียกได้ทั้งคลาสที่เป็นคลาสหลักและคลายย่อยได้ ส่งผลให้เวลาในการเรียกโปรแกรมภาษาจาวามาใช้ งาน คือเวลาการใช้งานขึ้นอยู่กับว่าในการอินเตอร์พรีเตอร์ของแต่ละครั้งจะเรียกคลาสทั้งหมดมาทำงาน หรือเรียกคลาสบางส่วนมาใช้งาน เพราะการอินเตอร์พรีเตอร์แต่ละครั้ง จะทำการโหลด (Load) คลาส มาใช้งานเท่าที่เรียกใช้เท่านั้น มีผลทำให้เวลาในการทำงานของแต่ละครั้งเป็นเวลาใช้งานจริงตามการ โหลดจำนวนเมธอดมาทำงานเท่านั้น ซึ่งแตกต่างจากโปรแกรมเชิงวัตถุภาษาซีพลัส ๆ ที่โค๊ดของ ภาษาซีพลัส ๆ จะถูกคอมไพล์เพียงครั้งเดียวทั้งตัวโค๊ดทำให้ตัวไฟล์นามสกุล .exe (executable code) มีขนาดใหญ่เพราะได้รวมฟังก์ชั่นทั้งโปรแกรมไว้ครั้งเดียว ดังนั้นเวลาในการทำงานจึงเท่ากันทุกครั้งที่ เรียกโปรแกรมมาใช้งาน



ภาพที่ 2.1 แสดงสร้างคลาสหลักและคลาสย่อย

จากภาพที่ 2.1 แฟ้มข้อมูลที่เป็นโค๊ดจาวาชื่อ Abc.java เป็นชื่อคลาสหลักชื่อ Abc.class ที่ถูก ทำการคอมไพล์แล้วพบว่ามีคลาสย่อยชื่อ a และ b ที่อยู่ภายในคลาสหลัก Abc เขียนแสดงผลการ คอมไพล์เป็นรูปคลาสเป็นดังนี้ Abc\$a.class และ Abc\$b.class ตามลำดับ

ในการกำหนดชื่อโค้ดจาวา ตามกล่าวมาข้างต้นว่าการที่ทำการบันทึกโค้ดควรเป็น ซื่อเดียวกัน กับชื่อคลาส แต่ก็มีการกำหนดชื่อแฟ้มข้อมูลกับคลาส ในบางกรณี คือ

 ถ้าคลาสใดไม่มีการกำหนดสิทธิ์การเข้าถึงด้วย public หรือ private หน้า class ชื่อคลาส จะเป็นชื่ออะไรก็ได้ ไม่จำเป็นต้องเป็นชื่อเดียวกับแฟ้มข้อมูล (ภาพที่ 2.2)



ภาพที่ 2.2 แสดงการบันทึกชื่อแฟ้มข้อมูล a.java ด้วยชื่อคลาสที่แตกแต่งกันด้วยชื่อ Abc

จากภาพที่ 2.2 โค็ดจาวาชื่อ a.java สามารถสร้างชื่อแฟ้มข้อมูลที่แตกต่างกับชื่อคลาส (class Abc) ซึ่งไม่ผิดกฎการตั้งชื่อ สามารถทำการสั่งคอมไพล์ได้ไม่แจ้งขอผิดพลาด (Output Completed) แต่โค็ดจาวาไม่สามารถสั่งให้ทำงาน (Run) ให้ทำงานเป็นลักษณะบนเครื่องส่วนบุคคล (Stand Alone) ได้ เพราะในการสั่งให้ทำงานด้วยการรันไทล์จาวาเป็นการเรียกคลาส a ด้วยคำสั่ง java a แต่ a เป็น เพียงชื่อโค็ดจาวา ส่วนคลาสที่ถูกสร้างขึ้นด้วยคำสั่ง javac a.java ทำการสร้างคลาสชื่อ Abc (class Abc) จึงไม่สามารถทำงานได้ แต่สามารถทำการเรียกคลาส Abc ที่ถูกสร้างได้ด้วยคำสั่ง java Abc (ภาพ ที่ 2.3) ได้ดังนี้

Administrator: Command Prompt	- • ×
C:\Program Files (x86)\Java\jdk1.6.0_14\bin>javac a.java C-\Program Files (x86)\Java\jdk1.6.0_14\bin>java Abc	
C:\Program Files (x86)\Java\jdk1.6.0_14\bin>_	-
	н. •

ภาพที่ 2.3 แสดงการเรียกใช้งานคลาสที่มีชื่อแตกต่างกับโค๊ดจาวา

จากภาพที่ 2.3 ในการบันทึกโค๊ดจาวา a.java ควรทำการเก็บไว้โฟล์เดอร์เดียวกับที่ตัวคอมไพล์ กับตัวรันไทล์จาวา (c:\Program Files (x86)\Java|jdk1.6.0_14\bin) ป้องกันการจะทำให้เกิดการ ผิดพลาด ในการเรียกใช้คำสั่งไม่พบไพล์ที่ต้องการเรียกใช้ (a.java)

การกำหนดชื่อแฟ้มข้อมูลที่มีการกำหนดสิทธิ์การเข้าถึงคลาสด้วย public ตามด้วย class ชื่อต้องเป็นชื่อเดียวกัน ซึ่งการกำหนดสิทธิ์การเข้าถึงคลาสด้วย public ในหนึ่งโค๊ดสามารถกำหนดได้ เพียงครั้งเดียวเท่านั้น ดังตัวอย่างโปรแกรม Applet ที่เป็นรูปแบบการทำงานของโปรแกรมภาษาจาวา อีกรูปแบบหนึ่งที่ทำงาน (รัน) โดยผ่านทางบราวเซอร์ (Browser) ซึ่งมีขั้นตอนดังนี้ 1.) สร้างโค๊ดจาวา (FirstApplet.java) 2.) ทำการคอมไพล์ด้วย javac FirstAplet.java ได้ผลเป็นคลาส FirstApplet.class
 นำคลาสมาเขียนเป็นสคริปต์ แทรกใน HTML 4.) ทำการเรียกใช้ได้ด้วยบราวเซอร์ (Internet Explorer, Google Chrome, Fire Foxe เป็นต้น)

2.1) สร้างโค๊ดจาวาสำหรับ Applet ทำการบันทึกแฟ้มชื่อ FirstApplet.java

```
import java.applet.*;
import java.awt.*;
public class FirstAppelt extends Applet {
    public void paint(Graphics g) {
        g.drawString("Hello World!", 25, 50);
    }
}
```



2.2 ทำการคอมไพล์ FirstApplet เป็นการสร้างคลาส FirstApplet.class (ภาพที่ 2.4)

ภาพที่ 2.4 แสดงการผลคอมไพล์ FirstApplet.java ได้คลาส FirstApplet.class

Normal

In 10

10

col 1

00 PC

REC INS READ

2.3 ทำการสร้างสคริปต์ Applet แทรกในเว็บเพ็จ HTML โดยทำการบันทึกด้วยแฟ้มข้อมูลชื่อ ใด ๆ ที่นามสกุล html ในที่นี้ตั้งชื่อเป็น JavaApplet.html ดังตัวอย่างโปรแกรมด้านล่าง (ภาพที่ 2.5)

Output completed (5 sec consumed)

FirstAppelt.class <

FirstArgs.class

```
<HTML><HEAD><TITLE> This is My First Applet </TITLE></HEAD>
<BODY> การแสดง Applet ที่สร้างด้วยโปรแกรมภาษาจาวา
<P>
<APPLET code = "FirstApplet.class" width = 200 height =100> </APPLET>
</BODY> </HTML>
```



ภาพที่ 2.5 แสดงการแทรกสคริปต์ Applet ลงใน HTML

2.4 ทำการเรียก Applet มาใช้งานสามารถทำการเรียกใช้ได้ 2 ลักษณะคือ 2.4.1 ใช้ appletviewer (ภาพที่ 2.6)

:\Program Files	(x86)\Java\jdk1.6.0_14\bin>ap	pletviewer D:\Java\JavaApplet.html
	Applet Viewer: FirstApplet.class	
	Applet	
	Hello World!	
	Applet started.	

ภาพที่ 2.6 แสดงการเรียกใช้ Applet ผ่านทาง appletviewer

2.4.2 การเรียกผ่าน Applet ผ่านทางบราวเซอร์ (ภาพที่ 2.7)



ภาพที่ 2.7 แสดงการเรียกใช้ Applet ผ่านทางบราวเซอร์ Chome

หมายเหตุ ในการเรียกใช้ผ่านบราวเซอร์ อาจมีการแสดงข้อผิดพลาด ให้ทำการปรับปรุง Java (TM) ถูกบล็อกเนื่องจากล้าสมัย และด้วยในด้านการกำหนดความปลอดภัยของตัวบราวเซอร์ ของแต่ละค่าย ดังนั้นต้องทำการเลือกทำการ **อัปเดตปลั๊กอิน และเลือก **เรียกใช้ตอนนี้** เป็นลำดับถัดมา (ภาพที่ 2.8)



ภาพที่ 2.8 แสดงการเรียกใช้อัปเดตปลั๊กอินและการอนุญาตเปิดการใช้งาน Java(TM)

ส่วนประกอบของโปรแกรมภาษาจาวา

ภาษาของโปรแกรมคอมพิวเตอร์ทุกภาษา ไม่เว้นกระทั้งโปรแกรมภาษาจาวา จะมีการกำหนด คำเฉพาะ และสัญลักษณ์ (การกำหนดดังกล่าวเรียกว่า token) ว่ามีความหมายอย่างไร รูปแบบของคำ (lexical grammar) เขียนอย่างไร รวมทั้งโครงสร้างระดับประโยค (syntax grammar) ว่าต้องเขียน อย่างไรให้ถูกต้อง โดยในการเขียนโปรแกรมภาษาจาในแต่ละโปรแกรม จะมีรูปแบบของซอร์สโค๊ด (source code) ที่ประกอบด้วย อักษรหรือคำสั่งที่มองไม่เห็น (white space) คำอธิบายโค๊ด (comment) คำสงวน (keywords) ชื่อเฉพาะ (identifiers) คำจัดกลุ่มหรือแยกกลุ่ม (separators) คำ ที่ใช้แสดงค่าคงที่ (literals)

อักษรหรือคำสั่งที่มองไม่เห็น (white space)

ในการเขียนโค๊ดทุกครั้ง โค๊ดทุกโค๊ดจะต้องมีส่วนประกอบของอักษรหรือคำสั่งที่มองไม่เห็นอยู่ เสมอ เช่น โค๊ดที่เขียนว่า public static void main(String args[]) พบว่าระหว่างคำว่า public static void main จำมีการเว้นวรรคระหว่างคำทุกคำ ซึ่งเป็นข้อกำหนดของการเขียนโค๊ดของภาษาจาวา ถ้า ผู้เขียนโปรแกรมไม่ปฏิบัติ คือการเขียนโดยไม่เว้นวรรค (publicstaticvoidmain) เมื่อสั่งคอมไพเลอร์ จะเกิดการอ่านคำสั่งไม่เข้าใจ (error) รวมทั้งคำสั่ง String args[] ถ้าไม่เว้นวรรค (Stringargs[]) ส่งผล ทำให้ตัวแปลภาษาหรือคอมไพเลอร์ ไม่เข้าใจคำสั่งที่เขียนได้

ในการเขียนโค็ดของโปรแกรมภาษาจาวา ผู้เขียนโปรแกรมสามารถใช้ การเว้นวรรคได้อย่าง อิสระหมายความว่า สามารถเว้นวรรค ระหว่างคำได้มากกว่า 1 เคาะ หรือ เว้นที่ละย่อหน้า หรือ เว้น เป็นที่ละบรรทัด เพื่อให้โค๊ดที่เขียนอ่านเข้าใจง่าย ก็สามารถทำได้โดยไม่มีผลกระทบต่อการแปลคำสั่ง หรือคอมไพล์ ของโปรแกรม

คำอธิบายโค๊ด (comment)

การเขียนโปรแกรมที่ดี ผู้เขียนควรเขียนคำอธิบายของแต่ละคำสั่งหรือแต่ละฟังก์ชันหรือเมธอด ว่าเขียนไว้เพื่อทำอะไร เป็นการเตือนความจำสำหรับผู้เขียนโปรแกรม และสำหรับผู้อ่านโปรแกรมทั่วไป ที่ต้องการศึกษาโค้ดที่ได้ เพื่อนำทำการพัฒนาต่อ ซึ่งการคำอธิบายโค้ด สามารถทำได้เหมือนกับภาษาซี ซึ่งแบ่งคำอธิบายโค้ดออกเป็น 3 ลักษณะคือ

- ลักษณะแรก ใช้เครื่องหมาย /* ตามด้วยข้อความทั้งภาษาไทยหรือภาษาอื่นๆ และทำการ ปิดคำอธิบายด้วยเครื่องหมาย */
- ลักษณะที่สอง ใช้เครื่องหมาย // ตามด้วยข้อความ ได้ตลอดหนึ่งบรรทัด โดยไม่ต้องมี เครื่องหมายใดสั่งปิด
- ลักษณะที่สาม เป็นการเขียนคำอธิบายมากว่าหนึ่งบรรทัดสามารถทำได้โดย ใช้เครื่องหมาย /** และทำการปิดด้วยเครื่องหมาย */ หรือเครื่องหมาย **/

คำสงวน (keywords)

คำสงวน เป็น คำหรือตัวอักษรพิเศษ ที่ภาษาโปรแกรมคอมพิวเตอร์แต่ละภาษา จำเป็นต้อง สงวนคำบางกลุ่มไว้ เพราะคำสงวน เป็นคำสั่งของโปรแกรมภาษานั้นห้ามไม่ให้ผู้เขียนโปรแกรมนำมาใช้ เป็นหน้าที่ใหม่ เป็นชื่อตัวแปร ชื่อฟังก์ชัน ชื่อเมธอด ชื่อคลาส เป็นต้น ซึ่งในโปรแกรมภาษาจาวา มีคำ สงวนจำนวน 50 คำคือ

abstract, boolean, break, byte, case, catch, char, class, const, continue, default, do, double, else, extends, false, final, finally, float, for, goto, if, implements, import, instanceof, int, interface, long, native, new, null, package, private, protected, public, return, short, static, super, switch, synchronized, this, throw, throws, transient, true, try, void, volatile, <code>uar</code> while

ชื่อเฉพาะ (identifiers)

ชื่อเฉพาะ เป็นการกำหนดชื่อให้กับโค๊ดในโปรแกรม ชื่อตัวแปร ชื่อคลาส รวมทั้งชื่อแฟ้มข้อมูล ห้ามนำคำสงวนมาใช้เด็ดขาด เว้นแต่การกำหนดให้มีความแตกต่างเล็กน้อย เช่น ชื่อของตัวแปร ต้องมี เครื่อง \$ นำหน้าชื่อตัวแปร ชื่อของคลาสควรขึ้นต้นชื่อคลาสด้วยตัวอักษรพิมพ์ใหญ่ เช่นด้วยกับชื่อ แฟ้มข้อมูลเป็นเหมือนชื่อคลาส ซึ่งในโค๊ดภาษาจาวา ชื่อเฉพาะที่ใช้มีลักษณะเป็น case-sensitive คือ คำของตัวอักษรใหญ่กับตัวอักษรเล็กเป็นคนละชื่อกัน เช่น คำว่า First กับ first เป็นชื่อที่เป็นคนละคำ กันสามารถนำมาใช้กำหนดเป็นตัวแปรได้ทั้งสองตัว เป็นต้น

้คำจัดกลุ่มหรือแยกกลุ่ม (separators)

ในการเขีย[ุ]นโปรแกรม การ[ู]่จัดกลุ่มหรือแยกกลุ่ม มีความสำคัญอย่างมากในการจัดให้เป็นกลุ่ม ของคำสั่งใด ฟังก์ชันหรือเมธอดใด ให้ทำงานเป็นชุดเดียวกัน หรือปฏิบัติตามเฉพาะกลุ่มเท่านั้น ในการ จัดกลุ่มสามารถทำได้ด้วยเครื่องหมายแต่ละชนิดดังนี้

- เครื่องหมายวงเล็บเปิด "(" วงเล็บปิด ")" (parentheses) ใช้สำหรับเป็นช่องทางเข้า ช่อง ทางออก ของอาร์กูเมนต์ (argument) ของฟังก์ชันหรือเมธอดทุกครั้ง ใช้สำหรับเป็นส่วนที่ กำหนดเงื่อนไขของ if-else, while, for, และ do รวมทั้งใช้สำหรับเป็นการกำหนดลำดับ การคำนวณทางคณิตศาสตร์ในส่วนประโยคทางคณิตศาสตร์ (expressions)
- เครื่องหมายปีกกาเปิด "{" ปีกกาปิด "}" (braces) ใช้หรับพื้นที่สมาชิกของ array และใช้ สำหรับกำหนดขอบเขตของ block, method และ class
- เครื่องหมายก้ามปูเปิด "[" ก้ามปูปิด "]" (brackets) ใช้สำหรับกำหนดจำนวนของ array และใช้สำหรับกำหนดตำแหน่งของ array ที่ต้องการติดต่อใช้งาน
- เครื่องหมายเซมโคลอน ";" (semicolon) ใช้สำหรับสั่งจบแต่ละคำสั่ง เพื่อให้ตัวแปลภาษา (compiler) นำคำสั่งก่อนเครื่องหมายเซมิโคลอนไปทำการแปลคำสั่ง
- เครื่องหมายคอมมา "," (comma) ใช้สำหรับการแยกประโยคของการประกาศ (declaration) ตัวแปร รวมทั้งในการแยกประโยคย่อยใน for(ค่าเริ่มต้น, เงื่อนไข, เพิ่มค่า หรือลดค่า)
- เครื่องหมายจุด "." (full stop) ใช้สำหรับแยกชื่อ package แยกชื่อ subpackage และใช้ สำหรับแยกชื่อ class หรือชื่อ instance กับชื่อสมาชิก
คำที่แสดงค่าคงที่ (literals)

ในการเขียนโค๊ดโปรแกรมภาษาจาวา ต้องมีการกำหนดค่าคงที่ให้กับตัวแปรชนิดต่างๆ สามารถแบ่งเป็น 5 ชนิดคือ

- ค่าคงที่จำนวนเต็ม (integer literal) ซึ่งค่าคงที่จำนวนเต็มสามารถกำหนดตามเลขฐานได้
 3 เลขฐาน ประกอบด้วย
 - เลขฐานแปด ในการกำหนดให้ใช้เลขศูนย์ (0) นำหน้าเลขที่ต้องการกำหนดค่า
 เช่น 0123 เป็นต้น
 - เลขฐานสิบ ในการกำหนดห้ามใช้เลขศูนย์ (0) นำหน้าเลขที่ต้องการกำหนดค่า
 เช่น 123 เป็นต้น
 - เลขฐานสิบหก ในการกำหนดให้ใช้เลขศูนย์เอ๊ก (0x) นำหน้าเลขที่ต้องการ กำหนดค่า เช่น 0x123 เป็นต้น
- ค่าคงที่เลขทศนิยม (floating point literal) สามารถกำหนดได้ 2 ลักษณะคือ 0.123 กับ กำหนดในรูปแบบวิทยาศาสตร์ 1.23e-1 หรือ 1.23E-1 เป็นต้น
- ค่าคงที่ค่าความจริง (boolean literal) เป็นการกำหนดค่าทางด้านโลจิก (logic) มี 2 ค่า คือ true กับ false
- ค่าคงที่ตัวอักษร (character literal) กำหนดค่าให้อยู่ภายใต้เครื่องหมาย single quotes เช่น '123' หรือ 'a' เป็นต้น
- ค่าคงที่สตริง (string literal) กำหนดค่าให้อยู่ภายใต้เครื่องหมาย double quotes เช่น "123" หรือ "abc" เป็นต้น

ตัวอย่างโปรแกรม

ให้นักศึกษาเขียนโปรแกรม ตามตัวอย่างที่กำหนดเพื่อให้ทราบถึงหลักการการเขียนโปรแกรม ภาษาจาวา ให้นักศึกษาฝึกทักษะในการเขียนโปรแกรมตามโครงสร้างของภาษาจาวาให้มีความเข้าใจ เป็นสำคัญ

ตัวอย่าง 2.1 โปรแกรมในการกำหนดค่าคงที่

```
public class MyfirstExpression
```

{

public static void main(String[] args)

int a = 10; //Assignment Variable a is integer type.

System.out.println("Display data area");

double b = 12345.56789; //Assignment Variable a is double type.

System.out.println("Display value of this expression b = " + b);

/** Show information for user by System.out.println ("information display on screen"), and show another data by "+" value in variable **/

}

{

จากตัวอย่างโปรแกรม ให้นักศึกษาอธิบายสาระสำคัญของโปรแกรม โดยทำการวิเคราะห์การ ทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่าทำงานอย่างไรโดยอ้างอิงจากเรื่อง อักษรหรือคำสั่งที่มองไม่เห็น คำอธิบายโค๊ด คำสงวน ชื่อเฉพาะ คำจัดกลุ่มหรือแยกกลุ่ม คำที่ใช้แสดงค่าคงที่ เป็นต้น

```
ตัวอย่าง 2.2 โปรแกรมในการแสดงค่าที่กำหนด
```

```
public class MySecondExpression
```

```
{ public static void main(String[] args)
{ int a = 21
    String name = "ชื่อนักศึกษา";
    Boolean rich = true;
    System.out.println("Display student name = "+ name );
    System.out.println("My old "+ a + "years" );
    System.out.println("My family is riches ? " + rich);
}
```

ให้นักศึกษาอธิบายการทำงานโปรแกรม ด้วยการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละ บรรทัดว่าทำงานอย่างไรโดย

ตัวอย่าง 2.3 โปรแกรมในการกำหนดค่าในขอบเขต

public class MyThirdExpression

{	publi	c static v	oid main(String[] args)	
	{	boolea	an rich = true;	
		lf(rich)		
		{	System.out.println(" I have a lot of money ");	}
		else		
		{	System.out.println("My family need some money");	
			System.out.println("Please me ");	}
	}			
}				

ให้นักศึกษาอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่าทำงาน อย่างไรโดยอ้างอิง ทั้งโปรแกรม ตัวอย่าง 2.4 การเขียนโปรแกรม ด้วยการจบคำสั่งของแต่ละบรรทัดด้วยคำสั่ง ; หรือเขียนโค๊ดใน บรรทัดเดียวกันมากกว่าหนึ่งคำสั่ง

```
public class ExpAndSemicolon
```

```
{ public static void main(String[] args)
{ int A = 100, B = 200; double C;
    System.out.println("Display variable A = "+ A ); System.out.println("
Display variable B = "+ B );
    System.out.println("The result of Rectangular area is " + A*B + " unit"
    + " . ");
}
```

ให้นักศึกษาสังเกตผลการทำงานของโปรแกรมว่าปกติ หรือมีข้อผิดพลาดเรื่องใดบ้าง

```
ตัวอย่าง 2.5 โปรแกรมที่ยังไม่สมบูรณ์
```

```
BugCode
        public static viod nain(String[] args)
{
        {
                lne a = 10:
                Boolean Grade = false;
                lf(grade)
                        {
                        Syftem.out.println(" Try learning again. ");
                        Sytem.ut.println(" Cheer up...");
                        }
                else
                        {
                        System.out.print("You are very good."):
                        Systen.out.println("Good luck!!!");
       }
}
```

ให้นักศึกษาช่วยกันในการทำการแก้ไขโค๊ดให้ถูกต้อง และเขียนโค๊ดใหม่ที่สมบูรณ์

การเขียนโปรแกรมเชิงวัตถุ ด้วยโปรแกรมภาษาจาวา เป็นการเขียนโปรแกรมคอมพิวเตอร์ที่ พัฒนามาจากการจำลองแนวคิดของมนุษย์ โดยการมองของต่างๆ รอบตัวที่เป็นปัจจัยที่ใช้ในการป้อน ข้อมูลที่เป็นก้อนๆ โดยในแต่ละก้อนนั้นๆ มีส่วนประกอบของหนึ่งก้อนวัตถุเป็นค่าตัวแปรต่างๆ รวมทั้ง ก้อนวัตถุหนึ่งก้อนมีหน้าที่ในการทำงานที่แตกต่างกัน ซึ่งอาจตีความหมายเมื่อเทียบกับการเขียน โปรแกรมแบบโครงสร้าง ภาษาชีว่า ก้อนวัตถุหนึ่งก้อนเทียบได้กับหนึ่งโครงสร้างของภาษาซี ในหนึ่ง ก้อนสามารถประกอบด้วยตัวแปรต่างๆ ประกอบด้วยฟังก์ชันที่หน้าที่ในการทำงาน เทียบได้กับเมธอด ของภาษาจาวา แต่ด้วยภาษาจาวาได้มีการพัฒนามาจากภาษาซีพลัสพลัส จึงทำให้ตัวภาษาจาวาได้ทำ การปรับปรุงการให้มีทำงานที่มีประสิทธิภาพมากขึ้น ภาษาจาวาสามารถส่งต่อหรือสร้างวัตถุให้จาก ต้นแบบของวัตถุเดิมได้เหมือนกับต้นฉบับ หรือสามารถสร้างวัตถุให้ให้มีหน้าที่ (เมธอด)ที่แตกต่างจาก ต้นฉบับได้(โพลิมอปริซึม) ภาษาจาวาสามารถนำวัตถุมาประกอบทำงานรวมกันได้ สามารถเพิ่มหน้าที่ ของวัตถุใหม่ได้ สามารถแลกเปลี่ยนข้อมูลระหว่างวัตถุได้ และนำผลการประมวลผลกับวัตถุ อื่นๆ ได้

แบบฝึกหัด

- 1. ยกตัวอย่าง white space พร้อมอธิบายความหมายของแต่ละคำ
- 2. เขียนคำ key word มาอย่างน้อย 10 คำ พร้อมอธิบายหน้าที่ของแต่ละคำมาพอสังเขป
- 3. เขียนคำอธิบายความแตกต่างระหว่างเครื่องหมาย "()" กับเครื่องหมาย "{}"
- เขียนตัวอย่างการกำหนดค่าให้กับตัวแปรทั้งห้าชนิด คือ 1) ค่าคงที่จำนวนเต็ม 2) ค่าคงที่เลข ทศนิยม 3) ค่าคงที่ค่าความจริง 4) ค่าคงที่ตัวอักษร 5) ค่าคงที่สตริง ชนิดละ 5 ค่า
- เขียนตัวอย่างการกำหนดค่าให้กับตัวแปรทั้งสามเลขฐาน คือ 1) เลขฐานสบหก 2) เลขฐานสิบ
 3) เลขฐานแปดมาอย่างน้อยเลขฐานละ 10 ค่า
- 6. จากโค๊ดด้านล่างจงเขียนแสดงผลการทำงาน

```
class IntergerType
{
    public static void main(String arg[])
    {
        System.out.println(10);
        System.out.println(010);
        System.out.println(0x10);
    }
    }
    iขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด
```

7. จากโค๊ดด้านล่างจงเขียนแสดงผลการทำงาน

```
class ConstantType
{
    public static void main(String arg[])
    {
        System.out.println(12345678901234);
        System.out.println(12345678901234L);
    }
}
```

เขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด

เขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด

```
9. จากโค็ดด้านล่างจงเขียนแสดงผลการทำงาน
class FloatDoubleType
```

```
{
    public static void main(String arg[])
    {
        float f = 3.145926535897896F;
        double d = 3.145926535897896D;
        System.out.println(f);
        System.out.println(d);
    }
}
```

เขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด

10. จากโค๊ดด้านล่างจงเขียนแสดงผลการทำงาน

```
class BooleanType
{
    System.out.println(true);
    System.out.println(false);
    System.out.println(1 < 10);
    System.out.println(10 <= 100);
    System.out.println(100 > 100);
    System.out.println(100 >= 100);
    System.out.println('A' == 'a');
    System.out.println('A' != 'a');
}
```

เขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด

เอกสารอ้างอิง

กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉ**บับพื้นฐาน.** กรุงเทพฯ: สำนักพิมพ์ เคทีพี.

รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.

วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

. (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/

<u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). The J2EE™ 1.4 Tutorial. Network Circle, CA: Sun Microsystem, Inc.

แผนบริหารการสอนประจำบทที่ 3

หัวข้อเนื้อหา

ตัวแปร ชนิดข้อมล

ตัวดำเนินการ

ตัวดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม

ตัวดำเนินการทางคณิตศาสตร์เลขทศนิยม

ตัวดำเนินการทางคณิตศาสตร์ร่วมกับการกำหนดค่า

ตัวดำเนินการทางคณิตศาสตร์การเพิ่มค่าและการลดค่า

สรุป

แบบฝึกหัด

เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกความสำคัญของการกำหนดตัวแปร
- 2. อธิบายชนิดของตัวแปรโปรแกรมภาษาจาวา
- 3. อธิบายขั้นตอนการใช้ตัวดำเนินการชนิดต่างๆ
- 4. มีทักษะการโปรแกรมภาษาจาวาด้วยการกำหนดตัวแปรชนิดต่างๆ
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตัวแปรและตัวดำเนินการได้เหมาะสม

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

- 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อตัวแปรและตัวดำเนินการ
- 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวา ด้วยตัวแปรและตัวดำเนินการ

1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์

2. กิจกรรมการเรียนการสอน

- 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
- 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
- 2.3 ทำแบบฝึกหัดเกี่ยวกับด้านตัวแปรและตัวดำเนินการ

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 3
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สังเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- 6. ตรวจแบบฝึกหัด

บทที่ 3 ตัวแปรและตัวดำเนินการ (Variable and Operator)

การเขียนโปรแกรมทุกภาษาคอมพิวเตอร์ รวมทั้งภาษาจาวา ในการนำตัวโปรแกรมมาช่วย แก้ปัญหางานหนึ่งๆ ตัวโปรแกรมจะทำงานต้องมีพื้นที่ในการจัดเก็บข้อมูล และนำข้อมูลมาในการ คำนวณค่าอย่างต่อเนื่อง ซึ่งในการติดต่อกับพื้นที่ของหน่วยความจำตัวโปรแกรมสามารถทำได้โดยผ่าน องค์ประกอบที่เรียกว่า **ตัวแปร** และในการทำงานของโปรแกรมต้องมีการนำค่าข้อมูลในตัวแปรมาทำ การคำนวณตามลักษณะต่างๆ ตามข้อกำหนดของผู้ใช้งานที่ต้องการให้โปรแกรมทำหน้าที่ใด การ คำนวณของโปรแกรมคอมพิวเตอร์สามารถทำการคำนวณตามลักษณะของมนุษย์ และโปรแกรม คอมพิวเตอร์ยังสามารถทำการคำนวณเฉพาะโปรแกรมคอมพิวเตอร์ที่มีรูปแบบการคำนวณที่แตกต่าง จากมนุษย์ได้ด้วย ซึ่งการคำนวณดังกล่าว ในภาษาการเขียนโปรแกรมคอมพิวเตอร์เรียกการกระทำนี้ว่า **ตัวดำเนินการ** ซึ่งมีความสำคัญอันดับต้นๆ ที่นักศึกษาต้องสามารถแยกแยะการใช้งานและรูปแบบการ สั่งงานโดยมีดังรายละเอียดต่อไปนี้

ตัวแปร (variable)

ตัวแปร (variable) เป็นการกำหนดภาชนะสมมุติขึ้นมาใช้ในการอ้างอิง (reference) ถึงตำแหน่ง (address) ของหน่วยความจำคอมพิวเตอร์ว่าตำแหน่งที่ใช้งานจริงอยู่ที่ตำแหน่งใดขึ้นอยู่ ระบบปฏิบัติการนั้น ว่าจะกำหนดให้ใช้พื้นที่ส่วนใด ที่ใช้ในการเก็บค่า (value) ให้กับตัวแปร ขนาดของ ค่าที่ต้องการเก็บไว้ในหน่วยความจำ จะมีผลให้ระบบปฏิบัติและตัวแปลภาษาโปรแกรมนำมากำหนดว่า ควรอยู่ที่ตำแหน่งใดถึงตำแหน่งใด ด้วยชนิดของตัวแปรแต่ละชนิดมีขนาดในการเก็บค่าข้อมูลไม่เท่ากัน ตัวแปรสามารถเก็บค่าข้อมูลได้ตั้งแต่ขนาด 8 บิท 16 บิท 32 บิท และ 64 บิท ส่งผลให้การจัด หน่วยความจำ (address mapping) ต้องดำเนินการตามการประกาศจองพื้นที่ของตัวแปรในโค๊ดที่ ผู้เขียนโปรแกรมกำหนดขึ้นมา

ในการเขียนโปรแกรม ต้องมีการกำหนดตัวแปรที่ใช้ในการเก็บค่าข้อมูลเสมอ จากที่กล่าวมา ด้านบนเพื่อให้ตัวแปลภาษาทราบว่า ผู้เขียนโปรแกรมต้องการใช้งานตัวแปร ดังนั้นก่อนใช้งานตัวแปรจึง จำเป็นต้องทำการประกาศตัวแปร (declaration) ตัวแปรก่อนใช้งานเสมอ โดยการประกาศตัวแปร ผู้เขียนโปรแกรมต้องกำหนดชนิดของตัวแปร ว่าต้องการนำมาเก็บค่าข้อมูลอะไร มีขนาดเท่าใด และ ต้องการใช้ชื่อใดเป็นการอ้างอิงการเข้าถึง (กำหนดค่าข้อมูลและการเรียกค่าข้อมูล) โดยการประกาศของ โปรแกรมภาษาจาวา มีโครงสร้างการประกาศหรือการจองพื้นที่หน่วยความจำดังนี้

<type> <identifier> [= <value>] [,<identifier> [= <value>] ...];

เมื่อ

	<type></type>	คือ	ชนิดตัวแปร (มีผลต่อขนาดของค่าข้อมูลที่ต้องการจัดเก็บ)
	<identifier></identifier>	คือ	ชื่อตัวแปร (ควรตั้งชื่อให้สื่อกับค่าของข้้อมูลที่ต้องการจัดเก็บ)
	[]	คือ	เป็นเครื่องหมายที่แสดงว่าผู้เขียนโปรแกรม จะกำหนดค่าใดๆ
			ในเครื่องหมาย[] หรือไม่ก็ได้ เช่น [= <value>] หมายถึง</value>
			จะกำหนดค่าข้อมูลให้กับตัวแปรหรือไม่ก็ได้ แต่ถ้ากำหนดค่า
			ข้อมูลให้ตัวแปรต้องกำหนดค่าข้อมูลหลังเครื่องหมาย = ด้วย
	<value></value>	คือ	ค่าข้อมูลของตัวแปร
ตัวอย่าง	การประกาศตัวแง	ไร	
	int a;	คือ	ประกาศตัวแปรชื่อ a กำหนดให้เป็นชนิดข้อมูล int (Integer: ค่า
			ข้อมูลเก็บได้เฉพาะจำนวนเต็มเท่านั้น) โดยไม่มีการกำหนดค่าข้อมูล
	int a = 10;	คือ	ประกาศตัวแปรชื่อ a เป็นชนิดข้อมูล int มีค่าข้อมูลเท่ากับ 10
	float b = 3.14;	คือ	ประกาศตัวแปรชื่อ b เป็นชนิดข้อมูล float (ค่าข้อมูลเก็บสามารถ
			เก็บเลขจำนวนจุดทศนิยมได้) ซึ่งในที่นี้ค่าข้อมูลของตัวแปรมี
			ค่าข้อมูลเท่ากับ 3.14
	int a = 10, x;	คือ	ประกาศตัวแปรชื่อ a เป็นชนิดข้อมูล int มีค่าข้อมูลเท่ากับ 10
			และมีการประกาศตัวแปรชื่อ x เป็นชนิดข้อมูล int เช่นเดียวกับ
			ตัวแปร a แต่ตัวแปร x ไม่มีการกำหนดค่าข้อมูล ซึ่งผู้เขียนสามารถ
			ประกาศเช่นนี้ได้ เป็นต้น

ดังแสดงการจองพื้นที่หน่วยความจำเสมือนดังภาพที่ 3.1 (ก.) แสดงโครงสร้างการประกาศตัว แปร และ (ข.) ตัวอย่างการประกาศตัวแปร a มีค่าข้อมูลเท่ากับ 10 (ค.) แสดงการประกาศตัวแปร x โดยไม่มีการกำหนดค่าข้อมูล พบว่าระบบจะทำการกำหนดตำแหน่ง (Address = 0x3363) เป็นการจอง พื้นที่ไว้ให้สำหรับค่าข้อมูลของตัวแปร x เท่านั้นแต่ค่าในหน่วยความจำยังไม่มีการกำหนดค่าให้



ภาพที่ 3.1 แสดงการการจองหน่วยความจำเสมือนของคอมพิวเตอร์

47

ชนิดข้อมูล (data type)

ชนิดข้อมูลของตัวแปรในโปรแกรมภาษาจาวาสามารถจัดการได้สองลักษณะคือ การจัดการบน รูปแบบคลาส (ซึ่งจะกล่าวภายหลัง) กับการจัดการรูปแบบตัวแปรพื้นฐาน ซึ่งมีความคล้ายกับตัวแปรใน โปรแกรมภาษาซีและซีพลัสพลัส แต่โปรแกรมภาษาจาวาได้ทำการปรับปรุงส่วนที่อาจก่อปัญหาสำหรับ นักพัฒนาโปรแกรมมือใหม่ ของตัวแปรภาษาซีและซีพลัสพลัส ที่ไม่มีการกำหนดค่าเริ่มต้นให้กับตัวแปร เมื่อมีการประกาศใช้ตัวแปร ส่งผลให้ค่าข้อมูลในตัวแปรภาษาซีและซีพลัสพลัส จะทำการสุ่มค่าข้อมูลมา ให้ตัวแปร ดังนั้นโปรแกรมภาษาจาวาจึงได้ทำการแก้ไข ในเรื่องการกำหนดค่าเริ่มต้นให้กับตัวแปร พื้นฐานทุกครั้งที่มีการประกาศตัวแปร โดยมีการกำหนดค่าเริ่มต้นให้กับตัวแปร (ตารางที่ 3.1)

ชนิดข้อมูล	ขนาดตัวแปร	ค่าเริ่มต้น
(Туре)	(Size)	(Default)
Boolean	1 บิท	false
byte	8 บิท	0
char	16 บิท	\u0000
short	16 บิท	0
int	32 บิท	0
float	32 บิท	0.0
long	64 บิท	0
double	64 บิท	0.0

ตารางที่ 3.1 แสดงชนิดข้อมูลของตัวแปรกับค่าเริ่มต้นของตัวแปร

จากตารางที่ 3.1 ชนิดข้อมูลที่เป็น boolean ในส่วนของโปรแกรมภาษาจาวามีขนาดในการ เก็บค่าเท่ากับ 1 บิท สามารถเก็บค่าได้ 2 สถานะ คือ อาจมีค่าเป็น true (ถูกต้อง) หรือมีค่าเป็น false (ไม่ถูกต้อง หรือ ผิด) ในการประกาศตัวแปร ในการจองพื้นที่ของหน่วยความจำ ถ้าผู้พัฒนาโปรแกรม ไม่ได้กำหนดค่าเริ่มต้นให้ ตัวแปลภาษาของโปรแกรมภาษาจาวาจะกำหนดค่าข้อมูลให้กับตัวแปรมีค่า เท่ากับ false ซึ่งแตกต่างจากโปรแกรมภาษาซีและซีพลัสพลัส ชนิดข้อมูลที่เป็น boolean มีจะแสดง เป็นเลขจำนวนเต็มสองสถานะคือ ค่าข้อมูลเป็น 1 (ถูกต้อง) หรือค่า 0 (ไม่ถูกต้อง)

ชนิดข้อมูล byte สามารถเก็บค่ำได้จำนวน 8 บิท คือค่าอยู่ระหว่าง 127 ถึง -128 รวมทั้งสิ้น จำนวน 256 จำนวน (ภาพที่ 3.2)



ภาพที่ 3.2 โปรแกรมแสดงตัวแปรชนิดไบท์

ชนิดข้อมูล char มีขนาด 16 บิท เป็นค่าคงที่ตัวอักษร ที่สามารถแสดงค่าได้ภายใต้เครื่องหมาย single quote ('a' หรือ '1') (ภาพที่ 3.3)

```
class VarCharSQ
   2
       {
   3
         public static void main(String[] args)
         {
   4
   5
           char a = 'a', b =',', c ='1', d ='Z';
           System.out.println("Char in single quotes => "+ "show var a = " + a );
  6
           System.out.println("Char in single quotes => "+ "show var b = " + b );
   7
           System.out.println("Char in single quotes => "+ "show var c = " + c );
  8
           System.out.println("Char in single quotes => "+ "show var d = " + d );
  9
        }
 10
11 }
  - JavaRun --
Char in single quotes => show var a
Char in single quotes => show var b =
Char in single quotes => show var c = 1
Char in single quotes => show var d = Z
```

ภาพที่ 3.3 แสดงค่าคงที่ตัวอักษรภายใต้เครื่องหมาย single quotes

ชนิดข้อมูล char มีขนาด 16 บิท เป็นค่าคงที่ตัวอักษร ที่เป็นตัวอักษรควบคุมการพิมพ์ (control character) ที่อยู่ภายใต้ตัวอักษร '\' เช่น backspace เขียนด้วย \b ขึ้นบรรทัดใหม่ เขียน ด้วย \n การ carriage return เขียนด้วย \r การ formfeed เขียนด้วย \f การย่อหน้า (tab) เขียนด้วย \t การแสดงเครื่องหมาย backslash (\) เขียนด้วย \\ การแสดงเครื่องหมาย single quote(') เขียน ด้วย \' การแสดงเครื่องหมาย double quote (") เขียนด้วย\" และการสั่งว่าง (null) เขียนด้วย \0 เป็นต้น (ภาพที่ 3.4)



ภาพที่ 3.4 แสดงการเขียนควบคุมด้วย back slash

กรณีแสดงค่าคงที่ตัวอักษร ด้วย escape sequences ด้วยเลขฐานแปด (octal) หรือ เลขฐานสิบหก (hexadecimal) ตามการแสดงค่าตัวอักษรตารางมาตรฐานของอเมริกา (ASCII: America Standard Code Interchange International) (ตารางที่ 3.2) และตัวอย่างโปรแกรม (ภาพที่ 3.5)

				50								
Dec	Hex	Name	char	Dec	Hex	Char	Dec	Hex	Char	Dec	Hex	Char
0	0	NULL	NUL	32	20	Space	64	40	0	96	60	`
1	1	Start of Header	SOH	33	21	!	65	41	А	97	61	а
2	2	Starr of Text	STX	34	22	"	66	42	В	98	62	b
3	3	End of Text	ETX	35	23	#	67	43	С	99	63	С
4	4	End of xmit	EOT	36	24	\$	68	44	D	100	64	d
5	5	Enquiry	ENQ	37	25	%	69	45	E	101	65	е
6	6	Acknowledge	ACK	38	26	&	70	46	F	102	66	f
7	7	Bell	BEL	39	27	'	71	47	G	103	67	g
8	8	Backspace	BS	40	28	(72	48	Н	104	68	h
9	9	Horizontal tab	HT	41	29)	73	49	I	105	69	i
10	0A	Line feed	LF	42	2A	*	74	4A	J	106	6A	j
11	OB	Vertical tab	VT	43	2B	+	75	4B	К	107	6B	k
12	0C	Form feed	FF	44	2C	,	76	4C	L	108	6C	ι
13	0D	Carriage feed	CR	45	2D	-	77	4D	М	109	6D	m
14	0E	Shift out	SO	46	2E		78	4E	Ν	110	6E	n
15	0F	Shift in	SI	47	2F	/	79	4F	0	111	6F	0
16	10	Data line escape	DLE	48	30	0	80	50	Р	112	70	р
17	11	Device control 1	DC1	49	31	1	81	51	Q	113	71	q
18	12	Device control 2	DC2	50	32	2	82	51	R	114	72	r
19	13	Device control 3	DC3	51	33	3	83	52	S	115	73	s
20	14	Device control 4	DC4	52	34	4	84	53	Т	116	74	t
21	15	Neg acknowledge	NAK	53	35	5	85	54	U	117	75	u
22	16	Synchronous idle	SYN	54	36	6	86	56	V	118	76	V
23	17	End of xmit block	ETB	55	37	7	87	57	W	119	77	w
24	18	Cancel	CAN	56	38	8	88	58	×	120	78	×
25	19	End of medium	EM	57	39	9	89	59	Y	121	79	У
26	1A	Substitute	SUB	58	3A	:	90	5A	Z	122	7A	z
27	1B	Escape	ESC	59	3B	;	91	5B	[123	7B	{
28	1C	File separator	FS	60	3C	<	92	5C	\	124	7C	
29	1D	Group separator	GS	61	3D	=	93	5D]	125	7D	}
30	1E	Record separator	RS	62	3E	>	94	5E	^	126	7E	2
31	1F	Unit separator	US	63	3F	?	95	5F		127	7F	DEL

ตารางที่ 3.2 แสดงค่าตัวอักษรตารางมาตรฐานของอเมริกา (ASCII)



ภาพที่ 3.5 แสดงค่าด้วยโค๊ด ASCII

ชนิดข้อมูล short มีหน่วยความจำขนาด 16 บิท (เป็นการจองพื้นที่หน่วยความจำ จำนวน 65,536 ตำแหน่ง) ในโปรแกรมภาษาจาวาไม่มีคุณสมบัติ unsigned (ไม่คิดเครื่องหมาย) เหมือนกับ โปรแกรมภาษาซี ดังนั้นค่าข้อมูลจึงสามารถเก็บได้ระหว่าง -32,768 ถึง +32,767 ดังแสดงภาพที่ 3.6 ดังนั้นถ้ามีการกำหนดค่าข้อมูลเกินค่าที่รับได้ จะเกิดการฟ้องการทำงานผิดพลาด (ภาพที่ 3.7)



ภาพที่ 3.6 แสดงค่าของชนิดข้อมูล short



ภาพที่ 3.7 แสดงการกำหนดค่าเกินชนิดตัวแปร short

ชนิดข้อมูล integer (int) มีหน่วยความจำขนาด 32 บิท ค่าข้อมูลสามารถเก็บค่าให้กับตัวแปร ได้ระหว่าง -2,147,483,648 ถึง +2,147,483,647 ดังนั้นในการกำหนดค่าให้กับตัวแปรชนิดข้อมูล int ถ้าเกินค่าด้านต่ำ (ติดลบ) จะเกิดการ Underflow คือค่าที่กำหนดจะทำการนำค่าด้านบวกตามจำนวนที่ เกินมากำหนดให้ และถ้าเกินค่าด้านบน (ด้านบวก) จะเกิดการ Overflow จะเป็นการกำหนดค่าโดยนำ ค่าด้านลบมากกำหนดค่าให้แทน (ภาพที่ 3.8)

```
class VarInteger
  2
      {
  3
        public static void main(String[] args)
  4
          int a = -2147483648;
  5
  6
          int b = 2147483647;
          System.out.println(Integer.MIN_VALUE);
  8
          System.out.println(Integer.MAX_VALUE);
           System.out.println("Value a = " + a +" => Underflow a -1 = " +(a-1));
  9
           System.out.println("Value b = " + b +" => Overflow b + 1 = " + (b+1));
 10
           }
 11
 12
       --- JavaRun --
2147483648
2147483647
Value a = -2147483648 => Underflow a -1 = 2147483647
Value b = 2147483647 => Overflow b + 1 = -2147483648
```

ภาพที่ 3.8 แสดงการกำหนดค่า underflow และ Overflow

โปรแกรมภาษาจาวา มีชนิดข้อมูลเลขทศนิยมจำนวนสองชนิดคือ ชนิดข้อมูล float มีขนาด 32 บิท เป็นชนิดข้อมูลที่สามารถมีเลขทศนิยมได้ถึง 7 หลัก ในการกำหนดค่าตัวแปรให้เป็นชนิดข้อมูลต้อง ใส่ตัวอักษร f หรือ F ต่อจากเลขทศนิยมที่กำหนดค่าให้กับตัวแปรชนิด float และชนิดข้อมูล double เป็นชนิดข้อมูลเลขทศนิยมขนาด 64 บิท ซึ่งแสดงเลขหลังจุดทศนิยมได้สูงสุด 16 หลัก ต้องเติมตัวอักษร d หรือ D ต่อท้ายค่าคงที่ ที่กำหนดกับตัวแปรชนิดข้อมูล double (ภาพที่ 3.9)

1	public class Float_Double
2	{
3	public static void main(String[] args)
4	{
5	int radius = 10;
6	float piF = $22/7$ f;
7	double $piD = 22/7d;$
8	System.out.println("Circle Area");
9	<pre>System.out.println("Radius = 10 ");</pre>
10	System.out.println("Circle Area (float) = "+ piF * radius * radius);
11	System.out.println("Circle Area (double) = "+ piD * radius * radius);
12	}
13	}
Circl Radiu	JavaRun e Area s = 10
Circl	= $=$ $=$ (float) = 314.2857
CIFCI	C AICA (UDUDIC) - 314.203/14203/143

ภาพที่ 3.9 แสดงชนิดข้อมูลเลขทศนิยมเปรียบเทียบระหว่าง float กัง double

ชนิดข้อมูล long เป็นชนิดข้อมูลเลขจำนวนเต็ม มีขนาดหน่วยความจำ 64 บิท ซึ่งเป็นตัวแปรที่ มีขนาดใหญ่ที่สุด (ภาพที่ 3.10)



ภาพที่ 3.10 แสดงการเปรียบเทียบค่าชนิดตัวแปร Long กับ Integer

ในการคำนวณของโปรแกรมภาษาจาวา มีส่วนสำคัญในการที่จะต้องกำหนดให้กับโปรแกรม นำไปคำนวณประกอบด้วย 2 ส่วนคือ เทอม (term) เป็นค่าหนึ่งๆ อาจเป็นค่าคงที่ ค่าที่อยู่ในตัวแปร ค่าที่ถูกส่งมาจากเมธอด (ฟังก์ชัน) และเทอม อาจอยู่เป็นค่าเดี่ยวก็ได้ แต่ส่วนใหญ่มักจะเป็นส่วนหนึ่ง ของ เอ็กเพรสชัน (expressions) ซึ่งเอ็กเพรสชันหมายถึง ประโยคทางคณิตศาสตร์ที่ประกอบด้วย เทอมกับตัวดำเนินการ (operator) แล้วนำผลกระทำมากำหนดค่า (assignment) ด้วยเครื่องหมาย = ให้กับตัวแปร โดยการแปลความหมายของคอมไพเลอร์ จะทำการนำค่าในเทอม หรือนำค่าในเอ็กเพรส ชัน ที่อยู่ทางขวาของเครื่องหมาย = นำมาเก็บไว้ในตัวแปรที่อยู่ทางซ้ายของเครื่องหมาย = นั้นๆ

ตัวดำเนินการ (operator)

ตัวดำเนินการ ในโปรแกรมภาษาจาวา สามารถทำการแบ่งตามลักษณะการดำเนินการ กับ ตัวถูกกระทำ (operand) ได้ 6 ลักษณะ ดังนี้

ตัวดำเนินการการกำหนดค่า (assignment operator) ใช้สัญญาลักษณ์ = (เครื่องหมายเท่ากับ) เป็นตัวดำเนินการในการทำหน้าที่ในการกำหนดค่า ต่างๆ (ค่าคงที่ เทอม เอ็กเพรส ชัน) เช่น การกำหนดค่าให้กับตัวแปร a เก็บค่าคงที่ 100 สามารถเขียนได้คือ a = 100 หรือ การ กำหนดค่า z เก็บผลลัพธ์การคำนวณของเอ็กเพรสชัน การบวกของตัวแปร a กับ 55 สามารถเขียนการ กระทำได้คือ z = a+55 เป็นต้น ตัวอย่างในภาพที่ 3.11

```
class Oassignment
 1
 2
    {
       public static void main(String[] args)
 3
 4
       ł
 5
          double a, b, c, d = 111.111;
          a = b = c = d:
 6
 7
          System.out.println(" ta = + a + + tb = + b + + tc = + c + + td = + d);
          a = ((c = (b=b+1)+1) + d);
 8
          System.out.println(" ta = + a + + tb = + b + + tc = + c + + td = + d);
 9
       }
10
11 }
•
     --- JavaRun ----
  a = 111.111 b = 111.111
a = 224.222 b = 112.111
                                             d = 111.111
d = 111.111
                               c = 111.111
                               c = 113.111
```

ภาพที่ 3.11 แสดงการกำหนดค่า (assignment) ด้วยสัญญาลักษณ์ = ให้กับตัวแปร

2. ตัวดำเนินการทางคณิตศาสตร์ (arithmetic operator) ในการโปรแกรม คอมพิวเตอร์ ถือเป็นเรื่องหลักของการนำเครื่องคอมพิวเตอร์มาช่วยมนุษย์ทำงาน ด้วยการนำโจทย์ที่ ต้องการแก้ปัญหา มาเขียนขั้นตอนวิธี (Algorithms) และนำมาเขียนโปรแกรม ดังนั้นการดำเนินทาง คณิตศาสตร์ จึงเป็นหัวใจหลักของการแก้ปัญหาโจทย์ ยิ่งโจทย์มีความสลับซับซ้อนยิ่งต้องส่งผลต่อการ คำนวณให้มีความสลับซับซ้อนตามไปด้วย ในโปรแกรมภาษาจาวาได้แบ่งลักษณะการดำเนินการทาง คณิตศาสตร์ออกเป็น 4 ประเภท ดังนี้

2.1 ตัวดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม (integer arithmetic operator) มี เครื่องหมาย + (บวก) เครื่องหมาย – (ลบ) เครื่องหมายคูณ (*) เครื่องหมายหาร (/) และเครื่องหมาย หารคิดเศษ (modulus: %) ในการดำเนินการของเครื่องหมายจำนวนเต็ม ที่กระทำกับเลขจำนวนเต็ม ส่งผลให้ผลลัพธ์การคำนวณได้เป็นเลขจำนวนเต็มด้วย แสดงตัวอย่างในภาพที่ 3.12

elass Oint	
[
public static void main(String[] args)	
{	
System.out.println(" $t10 - 20 = "+ (10-20)$;	
System.out.println(" $t10 + 20 = "+ (10+20)$;	
System.out.println(" \t10 * 20 = "+ (10*20));	
System.out.println(" $t10 / 20 = + (10/20)$;	
System.out.println(" \t10 % 20 = "+ (10%20));	
}	
JavaRun	
20 = 10 + 20 = 30	
20 = 200	
20 = 0 \$ 20 = 10	
<pre></pre>	<pre>class Oint { public static void main(String[] args) { System.out.println(" \t10 - 20 = "+ (10-20)); System.out.println(" \t10 + 20 = "+ (10+20)); System.out.println(" \t10 * 20 = "+ (10*20)); System.out.println(" \t10 / 20 = "+ (10/20)); System.out.println(" \t10 % 20 = "+ (10%20)); } } </pre>

ภาพที่ 3.12 แสดงการดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม

2.2 ตัวดำเนินการทางคณิตศาสตร์เลขทศนิยม (floating-point arithmetic operator) มีเครื่องหมาย + (บวก) เครื่องหมาย – (ลบ) เครื่องหมายคูณ (*) เครื่องหมายหาร (/) และเครื่องหมาย หารคิดเศษ (modulus: %) ในการดำเนินการของเครื่องหมายที่กระทำกับเลขทศนิยมส่งผลให้ผลลัพธ์ การคำนวณได้เป็นเลขทศนิยมด้วย (ข้อสังเกตถ้าเป็นโปรแกรมภาษาซี หรือ โปรแกรมซีพลัสพลัส เครื่องหมาย โมดูลัส (%) ไม่สามารถนำมากระทำกับเลขทศนิยมได้) แสดงตัวอย่างในภาพที่ 3.13

1 class Ofloat	
2 {	
3 public static void main(String[] args)	
4 {	
5 System.out.println(" $t100.5 - 20 = "+ (100.5-20)$));
6 System.out.println(" $t100.5 + 20 = "+ (100.5+20)$)));
7 System.out.println(" \t100.5 * 20 = "+ (100.5*20));
8 System.out.println(" $t100.5 / 20 = "+ (100.5/20)$));
9 System.out.println(" \t100.5 % 20 = "+ (100.5%2	20));
10 }	
11 }	
•	
100.5 + 20 = 120.5 100.5 * 20 = 2010.0 100.5 / 20 = 5.025 $100.5 \ 20 = 0.5$	



2.3 ตัวดำเนินการทางคณิตศาสตร์ร่วมกับการกำหนดค่า (assignment arithmetic operator) เป็นตัวดำเนินการที่นำผลการกระทำของคณิตศาสตร์มาเก็บยังตัวแปรเดิมที่ถูกกระทำ (ข้อสังเกต ตัวดำเนินการนี้ สามารถกระทำได้เฉพาะการคำนวณทางคอมพิวเตอร์เท่านั้น ทาง คณิตศาสตร์ทั่วไปไม่สามารถกระทำได้) เช่น a = a+1; หรืออาจเขียนลดรูปได้เป็น a +=1; ตัว ดำเนินการนี้สามารถเขียนได้ 5 รูปแบบคือ รูปแบบที่ 1: เขียนเป็น += รูปแบบที่ 2: เขียนเป็น -= รูปแบบที่ 3: เขียนเป็น *= รูปแบบที่ 4: เขียนเป็น /= และรูปแบบที่ 5: เขียนเป็น %= เป็นต้น แสดง ตัวอย่างภาพที่ 3.14

```
class OpAriAss
  1
  2
      {
  3
          public static void main(String[] args)
  4
          \{ int a = 10; \}
  5
             System.out.println(" Variable a = "+ a + " Assignment Arithmetic Operator "+" \ta -= 10 => "+ (a -= 10));
             System.out.println(" Variable a = "+ a + " Assignment Arithmetic Operator "+" \ta += 10 => "+ (a +=10));
  6
             System.out.println(" Variable a = "+ a + " Assignment Arithmetic Operator "+" \ta *= 10 => "+ (a *=10));
  8
             System.out.println(" Variable a = "+ a + " Assignment Arithmetic Operator "+" \ta /= 10 => "+ (a /=10));
  9
             System.out.println(" Variable a = "+ a + " Assignment Arithmetic Operator "+" \ta %= 10 => "+ (a %=10));
          }
 11
      }
.
Variable a = 10 Assignment Arithmetic Operator
Variable a = 0 Assignment Arithmetic Operator
                                                         a -= 10 => 0
a += 10 => 10
a *= 10 => 100
a /= 10 => 10
Variable a = 10 Assignment Arithmetic Operator
Variable a = 100 Assignment Arithmetic Operator
Variable a = 10 Assignment Arithmetic Operator
                                                          a %= 10 =>
```

ภาพที่ 3.14 แสดงการดำเนินการทางคณิตศาสตร์ร่วมกับการกำหนดค่า

2.4 ตัวดำเนินการทางคณิตศาสตร์การเพิ่มค่าและการลดค่า (increment and decrement arithmetic operator) เป็นตัวดำเนินการทางคณิตศาสตร์ ที่นิยมนำมาใช้ในการ กำหนดการเพิ่มค่าหรือลดค่า เป็นส่วนหนึ่ง ของกลุ่มคำสั่งวนรอบ (loop) หรือ นำมาเป็นประโยคการ คำนวณทั่วไปได้ ดัง เช่น a = a+1 สามารถเขียนลดรูปได้เป็น a++; หรือ ++a เช่นเดียว a = a-1 ซึ่ง สามารถเขียนลดรูปได้เป็น a--; หรือ --a แสดงตัวอย่างภาพที่ 3.15

```
class oIncDec1
 1
2
     {
       public static void main(String[] args)
3
        {
4
5
          int a = 1, b = 1;
           System.out.println("\ta befor a + + = " + a + "\tb befor b - = " + b);
6
7
          a++; --b;
8
          System.out.println("\ta after a++ = " + a +"\tb after --b = " +b);
       }
9
10 }
•
       - JavaRun -
 a befor a++ = 1 b befor b-- = 1
a after a++ = 2 b after --b = 0
```

ภาพที่ 3.15 แสดงการทำงานของการเพิ่มค่าและลดค่า

จากภาพที่ 3.15 พบว่าการกำหนดเครื่องหมายที่กระทำต่อตัวแปรตัวเดียวโดยไม่ผ่านตัวแปรอื่น การเขียนสามารถทำได้ทั้งแบบ postfix (a++) กับแบบ prefix (--x) ผลการกระทำต่อตัวแปรได้ผลลัพธ์ เท่ากัน แต่เมื่อนำการเพิ่มค่าหรือการลดค่านำมาเขียนเป็นประโยคร่วมกับตัวแปรอื่นการเขียนในแต่ละ แบบคือ postfix กับ prefix ให้ผลการคำนวณแตกต่างกัน แสดงตัวอย่างภาพที่ 3.16

```
class oIncDec2
 1
 2
   {
       public static void main(String[] args)
 3
 4
       {
          int a = 1, b = 1, x=0, y=0;
 5
          System.out.println("\tx befor x = a + + value x = " + x + "\t \t y befor <math>y = b - value y = " + y);
 6
         x = a + +; y = --b;
 7
         System.out.println("tx = a++ that value x = "+x + "t t y = -b that value y = "+y);
 8
       }
 9
10 }
•
       - JavaRun -
 x befor x = a++ value x= 0
                                 v befor v = b-- value v = 0
  x = a + + that value x = 1
                                 y = --b that value y = 0
```

ภาพที่ 3.16 แสดงการทำงานของการเพิ่มค่าและลดค่าโดยผ่านการประโยคคำนวณ

จากภาพที่ 3.16 พบว่าในการคำนวณการเพิ่มค่าหรือลดค่าของประโยคคำนวณทั้งสองประโยค x = a++; กับ y = --b; ให้ผลการคำนวณที่ส่งให้กับตัวแปร x กับ y ที่แตกต่างกัน ด้วยเพราะการ ประมวลผลของโปรแกรมคอมพิวเตอร์จะทำการคำนวณทางขวาของเครื่องหมายเท่ากับ (=) นำผลมา เก็บทางซ้ายมือ จาก x =a ++ (postfix) เป็นการบวกค่า a ด้วยหนึ่ง ทำให้ a มีค่าเท่ากับ 1 แล้วนำ 1 ไปเก็บไว้ที่ x ส่งผลให้ ค่า x เท่ากับ 1 ส่วนจาก y = --b (prefix) ค่าของ b ครั้งแรกยังไม่ได้ทำการลด ค่าหนึ่งค่า ทำให้ b มีค่าเท่ากับ 0 แล้วนำ 0 ไปเก็บไว้ที่ y ส่งผลให้ ค่า y เท่ากับ 0 ซึ่งถ้าทำรอบถัดไปค่า ของ y จะเท่ากับ -1 **3. ตัวดำเนินการระดับบิท (bitwise operator)** ในการกระทำระดับบิทสามารถ กระทำข้อมูลของตัวแปรที่เป็นชนิดตัวเลขขนาดต่างๆ ได้แก่ byte, short, integer, long และ char ตัว ดำเนินการที่นำมาดำเนินการระดับบิท มีดังนี้ อินเวอร์เตอร์ (NOT: ~) แอนด์ (AND : &) ออร์ (OR: |) เอ็กคูซีพออร์ (Excusive OR: ^) การเลื่อนขวา (shift right: >>) และเลื่อนซ้าย (shift left: <<) แสดง ตัวอย่างภาพที่ 3.17 แสดงการกระทำการด้วยตัวดำเนินการแอนด์

1	clas	ss oBoolean1
2	{	
3	I	public static void main(String[] args)
4	{	
5		int a = 130, b = 141;
6		int $aANDb = a\&b;$
7		System.out.print("\ta = "+a);System.out.print("\ta = ");System.out.println(Integer.toBinaryString((int)a));
8		System.out.print("\tb = "+b);System.out.print("\tb = ");System.out.println(Integer.toBinaryString(((int)b));
• 9		System.out.print("'ta&b = " + (Integer.toBinaryString((int)aANDb) + "'t = "+aANDb));
10	}	
11	}	
I C		
		JavaRun
a b		= 150 a = 1000000 = 141 b = 1000100
a	ab	= 10000000 = 128

ภาพที่ 3.17 แสดงการใช้ตัวดำเนินการ & แอนด์ (AND) ระดับบิท

จากภาพที่ 3.17 เป็นการนำค่า a เท่ากับ 130 ฐานสิบ มาทำการแปลงเป็นเลขฐานสองมีค่า เท่ากับ 10000010 ค่า b เท่ากับ 141 ฐานสิบเท่ากับ 10001101 ฐานสอง ด้วยคำสั่งบรรทัดที่ 7 และ 8 ตามลำดับ นำมาทำมาทำการแอนด์กันระหว่าง a กับ b และนำผลลัพธ์เก็บที่ตัวแปร aANDb ในบรรทัด ที่ 6 แสดงผลการการแอนด์ แสดงบรรทัดที่ 9 เป็นต้น

การดำเนินการเลื่อนด้วยการเลื่อนขวา (shift right: >>) และเลื่อนซ้าย (shift left: <<) แสดง ในภาพที่ 3.18

1 class oBoolean2 2 { public static void main(String[] args) 3 { 4 5 int a = 123;System.out.print("\ta = "+a);System.out.print("\ta = ");System.out.println(Integer.toBinaryString((int)a)); 6 7 System.out.println("\ta is shift right 4 bit = " + (Integer.toBinaryString((int) a>>4))); System.out.println("\ta is shift left 4 bit = " + (Integer.toBinaryString((int) a<<4))); 8 9 } JavaRun -----= 1111011 а = 123 a a is shift right 4 bit = 111 a is shift left 4 bit = 11110110000

ภาพที่ 3.18 แสดงการใช้ตัวดำเนินการเลือนบิท (shift) ระดับบิท

4. ตัวดำเนินการความสัมพันธ์เชิงเปรียบเทียบ (Relational Operators) ในการ เขียนโปรแกรมที่มีการเงื่อนไขในการกำหนดให้มีการตัดสินใจ ตัวดำเนินการความสัมพันธ์เชิง เปรียบเทียบ มีส่วนกำหนดความถูกต้องของเงื่อนไข ด้วยการเปรียบเทียบค่าในตัวแปรทั้งสองด้านของ ตัวดำเนินการ ได้แก่ ตัวดำเนินการเท่ากัน == (equal to) ตัวดำเนินการไม่เท่ากัน != (not equal to) ตัวดำเนินการมากกว่า > (greater than) ตัวดำเนินการน้อยกว่า > (lees than) ตัวดำเนินการมากกว่า และเท่ากับ >= (greater than or equal to) ตัวดำเนินการน้อยกว่าและเท่ากับ >= (lees than or equal to) สามารถนำมาใช้เปรียบเทียบ operand ชนิดข้อมูลในตัวแปรชนิด byte, char, short, int, long, float และ double แสดงในภาพที่ 3.19

1 class oRelation 2 { 3 public static void main(String[] args) { 4 int x = 0, y = 1;5 System.out.print("\tx = "+x);System.out.println("\ty = "+y); 6 System.out.println("\tx == y ? = "+(x == y));System.out.println("\tx != y ? = "+(x != y)); 7 System.out.println(" $tx \ge y$? = "+($x \ge y$));System.out.println(" $tx \le y$? = "+($x \le y$)); 8 } 9 10 } ∢ 🗌 ----- JavaRun -----x = 0 y = 1 x == y ? = false х != у ? = true = false х ≻= у ? х <= у ? = true

ภาพที่ 3.19 แสดงการใช้ตัวดำเนินการความสัมพันธ์เชิงเปรียบเทียบ (Relational Operators)

จากภาพที่ 3.19 เป็นตัวอย่างการนำค่าข้อมูลในตัวแปร x และในตัวแปร y ซึ่งตัวแปรทั้งคู่เป็น ตัวแปรชนิด Integer พบว่าการแสดงผลการทำงานของโปรแกรมจะแสดงในรูปของว่าถูกต้อง (true) หรือไม่ถูกต้อง (false) ซึ่งสามารถนำผลการทำงานไปใช้ในการทำงานร่วมกับกลุ่มคำสั่ง if/else ในเรื่อง ต่อไปได้

ตัวอย่างโปรแกรม

ให้นักศึกษาศึกษาโปรแกรม ตามตัวอย่างที่กำหนดเพื่อให้ทราบถึงหลักการการเขียนโปรแกรม ภาษาจาวา ให้นักศึกษาฝึกในการเขียนโปรแกรมตามโครงสร้างของภาษาจาวาให้มีความเข้าใจเป็น สำคัญ

ตัวอย่าง 3.1 โปรแกรมการกำหนดตัวแปร และการเปรียบเทียบตัวดำเนินการหารกับโมดูลัส class VarLab3 1

```
{
               public static void main(String[] args)
               {
                       short varA = 100;
                       int varB = 55;
                       float Result1, Result2, Result3, Result4;
                       Result1 = varA / 2;
                               System.out.println(Result1);
                       Result2 = varA \% 2;
                               System.out.println(Result2);
                       Result3 = varB / 2;
                               System.out.println(Result3);
                       Result4 = varB \% 2;
                               System.out.println(Result4);
                       }
       }
ตัวอย่าง 3.2
       public class MyOperator3 2
       {
               public static void main(String[] args)
               {
                       int a = 12, b = 20;
                       String Txt = "OK";
                       Boolean z1,z2;
               System.out.println("Variable a = "+ a +"\t Varible b = "+b );
                       z1 = (a == b);
               System.out.println("(a == b)" + "=>" + z1 );
                       z^2 = (a != b);
               System.out.println("(a != b)" + "=>" + z2 );
               }
       }
```

สรุป

ในการเขียนโปรแกรมเพื่อให้ทำการแก้ปัญหางานหนึ่งๆ ตัวแปร เป็นส่วนประกอบที่สำคัญอย่าง มากในการโปรแกรมทุกโปรแกรมต้องมีการเรียกใช้ตัวแปรทุกครั้งด้วยเพราะในการทำงานต้องมีการเก็บ ข้อมูล และนำข้อมูลมาใช้งานเสมอ ตัวแปรจึงต้องมีพื้นที่ในการติดต่อหน่วยความจำคอมพิวเตอร์ รวมทั้งข้อมูลที่เก็บในหน่วยความจำผ่านทางชื่อของตัวแปร ว่าเป็นตัวแปรเลขจำนวนเต็ม (int) ตัวแปร เลขทศนิยม (float) เป็นต้น นอกจากการนำข้อมูลที่เกี่ยวข้องกับตัวแปรแล้ว ในการทำงานของ โปรแกรมต้องมีการนำค่าข้อมูลในตัวแปรมาทำการคำนวณ ตามรูปแบบของตัวดำเนินการหรือ เครื่องหมายต่างๆ ของโปรแกรม ซึ่งตัวดำเนินการสามารถแบ่งตามลักษณะการคำนวณได้ดังนี้ 1) ตัว ดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม 2) ตัวดำเนินการทางคณิตศาสตร์เลขทศนิยม 3) ตัว ดำเนินการทางคณิตศาสตร์ร่วมกับการกำหนดค่า 4) ตัวดำเนินการทางคณิตศาสตร์การเพิ่มค่าและการ ลดค่า ในการโปรแกรมภาษาจาวาให้สามารถทำงานได้สมบูรณ์นักศึกษาต้องทำการศึกษารูปแบบการ สั่งงาน (Syntax) ต่างของทั้งตัวแปรและตัวดำเนินการ เป็นสำคัญ

แบบฝึกหัด

- 1. ให้นักศึกษาเขียนอธิบายถึงการเขียนโปรแกรมภาษาจาวาจำเป็นต้องใช้ตัวแปรหรือไม่เพาะเหตุใด
- 2. ในการเขียนโปรแกรมค่าข้อมูลที่ใช้ในการเก็บที่ตัวแปรสามาเก็บได้กี่ชนิดอะไรบ้าง
- ให้นักศึกษาเขียนค่าตัวอักษรตามตารางมาตรฐานของอเมริกา(ASCII) คำว่า Engineering โดยให้ทำ การเขียนค่าเป็นฐานสิบและเลขฐานสิบหก
- ในการคำนวณของโปรแกรมที่นำค่าที่เป็นเลขจำนวนเต็มกับเลขทศนิยมมาทำการคูณกัน อยาก ทราบว่าต้องประกาศตัวแปรเก็บผลลัพธ์เป็นตัวแปรชนิดใด
- 5. ในการกำหนดค่าคงที่ (120) ให้กับตัวแปรต้องใช้ ตัวดำเนินการใด
- ถ้าค่าข้อมูลของตัวแปร a = 100 ต่อมามีการเขียนโปรแกรม a =*10 อยากทราบว่า ค่าของ a จะ เป็นเท่าใด
- 7. เมื่อนำผลของค่า a มาทำการเขียนโปรแกรมต่อว่า a++ อยากทราบว่า ค่าของ a จะเป็นเท่าใด
- จากผลของ a ในข้อ 7 เมื่อนำมาทำการเขียนโปรแกรมต่อว่า a == 1001 อยากทราบว่า ผลจากทำ การทำงานของโปรแกรมจะเป็นเช่นใด
- เมื่อทำการกำหนดค่า x = 100 และ y = 200 แล้วทำการเขียนโปรแกรมในบรรทัดถัดว่า y >= x อยากทราบว่าผลการทำงานโปรแกรมจะตอบเป็น true หรือ false
- ให้นักศึกษาทำการนำค่าของตัวแปร x และ y ในข้อ 9 มาเขียนโปรแกรมโดยใช้ตัวดำเนินการ ความสัมพันธ์เชิงเปรียบเทียบ แล้วให้ผลการทำงานของโปรแกรมมีค่าเป็น true

เอกสารอ้างอิง

กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพฯ: สำนักพิมพ์ เคทีพี.

รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.

วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

. (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/

<u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). The J2EE™ 1.4 Tutorial. Network Circle, CA: Sun Microsystem, Inc.

แผนบริหารการสอนประจำบทที่ 4

หัวข้อเนื้อหา

ประโยคตามลำดับ ประโยคการตัดสินใจ ประโยค if-else การเขียนรูปแบบลัด if ประโยค switch-case ประโยคการวนลูป ประโยค for ประโยค while ประโยค do/while ประโยคการควบคุมอื่นๆเพิ่มเติม ประโยค break ประโยค continue ประโยค return สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกความสำคัญของการประโยคควบคุมการทำงาน
- 2. อธิบายความแตกต่างระหว่างการตัดสินใจกับการวนรอบ
- 3. อธิบายขั้นตอนการการทำงานของประโยคควบคุมการทำงาน
- 4. มีทักษะการโปรแกรมภาษาจาวาด้วยการกำหนดประโยคควบคุมการทำงาน
- 5. เขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานประโยคควบคุมการทำงาน

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อประโยคการตัดสินใจ ประโยคการวนลูป และ ประโยคการควบคุมอื่นๆ

 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวา หัวข้อประโยคการตัดสินใจ ประโยคการวนลูป และประโยคการควบคุมอื่นๆ

1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์

2. กิจกรรมการเรียนการสอน

- 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
- 2.2 ให้นักศึกษาอภิปรายผลการทดลอง

2.3 ทำแบบฝึกหัดเกี่ยวกับประโยคการตัดสินใจ ประโยคการวนลูป และประโยคการ ควบคุมอื่นๆ

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 4
- 2. ระบบอิน เทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั่งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 4 ประโยคควบคุมการทำงาน (Control statements)

ประโยคควบคุม เป็นประโยคที่ใช้ในการกำหนดการดำเนินการของโปรแกรมให้ทำงานตาม เงื่อนไขที่กำหนด ว่าให้ดำเนินการงานใด เมื่อใด จำนวนเท่าใด ซึ่งสามารถแบ่งรูปแบบการควบคุมการ ทำงานได้ 3 รูปแบบ คือ

ประโยคตามลำดับ (Sequence Statements) ประโยคการตัดสินใจ (Decision Statements) ประโยคการวนลูป (Loop Statements)

ประโยคตามลำดับ (Sequence Statements)

การเขียนโปรแกรมลักษณะใช้ประโยคให้มีการควบคุมการทำงานตามลำดับ ซึ่งเป็นพื้นฐานของ การเขียนโปรแกรมทั่วไป กล่าวคือ การเขียนโปรแกรมเป็นการนำคำสั่งมาเขียนเรียงต่อกัน เพื่อให้ทำการ แก้ปัญหาใดปัญหาหนึ่งให้สำเร็จ ดังนั้นการตีความหมายจึงนำคำสั่ง หรือประโยคในลำดับต้นไปทำงาน ก่อน ตามด้วยคำสั่งประโยคลำดับถัดมา ไปทำงานต่อจนเสร็จสิ้นงานที่กำหนด โดยคำสั่งหรือประโยค ที่มีอยู่ในโปรแกรมจะเริ่มต้นจากเมธอด main() ก่อนแล้วทำไปจนสิ้นสุดในเมธอด main() ตัวอย่าง ภาพที่ 4.1



ภาพที่ 4.1 แสดงการใช้ประโยคตามลำดับ

ประโยคการตัดสินใจ (Decision Statements)

ประโยคการตัดสินใจ เป็นการตัดสินใจภายใต้เงื่อนไขที่กำหนดไว้ว่าถ้าเงื่อนไขถูกต้อง ให้ตัดสินใจ ดำเนินการใด และเมื่อเงื่อนไขไม่ถูกต้องให้ดำเนินการใด ซึ่งประโยคการตัดสินใจมีอยู่ 2 รูปแบบ คือ ประโยค if-else ประโยค switch-case

ประโยค if-else

ประโยค if-else เป็นประโยคที่ใช้ในการควบคุมให้โปรแกรมเลือกดำเนินการไปในเส้นทางใด เส้นทางหนึ่ง โดยใช้ในการตรวจสอบเงื่อนไขว่าเป็นจริงหรือเท็จเป็นผลของ <boolean expression> เป็นตัวตัดสินใจ ซึ่งมีโครงสร้างของประโยค if-else ดังนี้

if (<boolean expression >) <statements>; [else <statements>;]

เมื่อ

<boolean expression:<="" th=""><th>>คือ ประ</th><th>ะโยคเงื่อนไขที่กำหนดตรวจสอบถ้</th><th><i>์</i> าถูกต้องให้ค่าบูลีนเป็น</th></boolean>	>คือ ประ	ะโยคเงื่อนไขที่กำหนดตรวจสอบถ้	<i>์</i> าถูกต้องให้ค่าบูลีนเป็น
	true	และให้ทำงานใน statements	ในบล็อกถัดไป และถ้า
	ตรวจสอ	วบเงื่อนไขว่าไม่ถูกต้องค่าบูลีนเป็น	false ให้ทำบล็อกถัดไป
	หรือทำด	กำเนินการหลัง else ถ้ามี	
<statements></statements>	คือ บล็อ	อกที่กำหนดงานให้ดำเนินงานในการเ	แก้ปัญหาของโปรแกรม
	ซึ่งอาจเห	ป็นคำสั่งหรือเมธอดเดียว หรือเป็นก	ลุ่มคำสั่งมากกว่าหนึ่ง
	คำสั่งต้อ	งเขียนภายใต้เครื่องหมาย { }	
[]	คือ เป็น	เครื่องหมายที่แสดงว่าผู้เขียนโปรแก	รม จะกำหนดค่าใดๆ
	ในเครื่อง	งหมาย[] หรือไม่ก็ได้ เช่น [else	<statements>;]</statements>
	หมายถึง	เ ถ้ามีการกำหนด else เป็นการกำห	านดบล็อกให้ดำเนินการ
	งานเมื่อเ	เงื่อนไขไม่ถูกต้อง หรือสามารถกำหา	นดเงื่อนไขที่สองเพิ่มมา
	ตรวจสอ	บบอีกเงื่อนไข สามารถทำได้ตามต้อง	การ แสดงภาพที่ 4.2



ภาพที่ 4.2 แสดง flow chart และ โครงสร้างการทำงานของ if-else

จากภาพที่ 4.2 เมื่อประโยคเงื่อนไขแบบที่ 1 เป็นการตรวจสอบ expression ถ้าถูกต้อง (true) ให้มาทำงาน statement 1 แล้วทำงาน statement 2 ต่อ และถ้าผลการตรวจสอบ expression ไม่ ถูกต้อง (false) ให้มาทำงาน statement 2 โดยไม่ทำงานใน statement 1 (ภาพที่ 4.3)

```
class Ex4_3 {
   1
         public static void main(String[] args) {
  2
           int a = 22;
  3
           int b = 0;
  4
           if (b==0)
  5
                        {
              System.out.println("Error: divide by zero!");
  6
              System.exit(0);
   7
            }
  8
            System.out.println("Result a/b = " +a/b);
  9
         }
 10
 11
     }
•
----- JavaRun ---
Error: divide by zero!
```

ภาพที่ 4.3 แสดงการใช้ if รูปแบบที่ 1

ส่วนแบบที่ 2 เป็นแบบที่มีการกำหนดงานให้ทำเมื่อ ผล expression ไม่ถูกต้องเพิ่มเติม คือ ประโยคเงื่อนไข expression ตรวจสอบถ้าถูกต้อง (true) ให้มาทำงาน statement 1 แล้วทำงาน statement 3 ต่อ และถ้าผลการตรวจสอบ expression ไม่ถูกต้อง (false) ให้มาทำงาน statement 2 แล้วทำงาน statement 3 ต่อ (ภาพที่ 4.4 และ ภาพที่ 4.5)

1	class Ex4_4 {
2	<pre>public static void main(String[] args) {</pre>
3	int $a = 22;$
4	int $b = 11;$
5	if (b==0) {
6	System.out.println("Error: divide by zero!");
7	System.exit(0);
8	} else
9	System.out.println("Result a/b = " +a/b);
10	}
11	}
Result	JavaRun a/b = 2

ภาพที่ 4.4 แสดงการใช้ if-else รูปแบบที่ 2



ภาพที่ 4.5 แสดงการใช้ if-else มากกว่าหนึ่งเงื่อนไข

การเขียนรูปแบบลัด if

ประโยค if สามารถเขียนให้สั้นลงได้ โดยมีข้อจำกัดคือ สามารถใช้ได้เฉพาะกับ if ที่เป็นแบบ 2 เงื่อนไข (if...else) เท่านั้น โดยมีรูปแบบดังนี้

```
variable = (<boolean expression >) ? <statement for true>:<statement for false>;
```

เมื่อ

```
variable คือ ตัวแปรในการเก็บผลการดำเนินการของเงื่อนไข
<boolean expression> คือ ประโยคเงื่อนไขที่กำหนดตรวจสอบถ้าถูกต้องให้ค่าบูลีน
เป็น true และให้ทำงานใน statements for true และถ้า
ตรวจสอบเงื่อนไขว่าไม่ถูกต้องค่าบูลีนเป็น false ให้ทำ
statement for false
```

เช่น

```
class Ex4_E {
```

```
public static void main(String[] args) {
    int score = 65;
    if (score >=50)
        System.out.println("Y");
    else System.out.println("N"); }
```

}

```
สามารถเขียนให้สั้นลงดัง (ภาพที่ 4.6)
```

```
class Ex4_6 {
      public static void main(String[] args) {
2
3
         int score = 65;
4
         char ans;
         ans = (score >=50)? 'Y' : 'N';
5
         System.out.println("Value ans = " + ans);
6
7
         }
8
•
      -- JavaRun
Value ans = Y
```

ภาพที่ 4.6 แสดงการใช้ if เขียนรูปแบบลัด
```
ตัวอย่างการใช้งาน if-else (ภาพที่ 4.7)
import javax.swing.JOptionPane;
public class Ex4 7
{
public static void main(String args[])
       {
       float sum = 0.0f, month;
       String input = JOptionPane.showInputDialog("Enter for month (1-12)");
       month =Float.parseFloat(input);
       String season;
       if (month ==1 || month ==2 || month==3)
               season = "Medium Hot";
               else if (month ==4 \parallel month ==5 \parallel month ==6)
                      season = "Very Hot";
               else if (month ==7 || month ==8 || month==9)
                      season = "Hard Rainy";
               else if (month ==10 || month ==11 || month==12)
                      season = "Min Hot";
               else
                      season = "ERROR";
       JOptionPane.showMessageDialog(null,"Season Wheath in Thailand is " + season);
       System.exit(0);
```

}

}

ผลการทำงานโปรแกรม 4.7 มีการป้อนค่า input (ภาพที่ 4_7) ให้กับตัวแปร month ผ่านทาง ตัวแปร input มาทำการตรวจเงื่อนไข ว่าอยู่ใน if-else ใดเพื่อนำค่าข้อความเก็บที่ตัวแปร season ใช้ ในการแสดงผลการทำงานโปรแกรม (ภาพที่ 4.8)

Input	X	
?	Enter for month (1-12) 11 OK Cancel	

ภาพที่ 4.7 แสดงการป้อนค่าให้กับ month ผ่านทาง input



ภาพที่ 4.8 แสดงผลการทำงานโปรแกรม

ประโยค switch-case

ประโยค switch-case เป็นการควบคุมการทำงานที่ทำการเลือกดำเนินการ ที่มีทางเลือกให้ ทำงานหลายงาน โดยการเลือกในการดำเนินการใดขึ้นอยู่กับ variable ที่มีโครงสร้างของประโยค ดังนี้

```
switch (<variable >)
{
    case <value> : <statements>; break;
    case <value> : <statements>; break;
    .
    .
```

```
default : <statements>;
```

เมื่อ

}

<variable></variable>	เป็น ค่า ที่ใช้กำหนดทางเลือก โดยการเทียบค่าของ variable ว่าตรง
	กับ value ที่ case ใด ตัวคอมไพเลอร์จะตัดสินใจเลือกงานนั้น ค่า
	ของ variable สามารถเป็นชนิดข้อมูล int, short, long, char, หรือ
	byte แต่ห้ามเป็น boolean, float, หรือ double
case <value></value>	เป็นตัวงานที่ถูกเลือกจาก ค่าของ variable ตรงกับ value ซึ่งค่าของ
	value เป็นตัวแยกแยะงาน ดังนั้นค่าของ value ห้ามมีค่าซ้ำกัน
<statements></statements>	เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม
	ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง
	คำสั่งต้องเขียนภายใต้เครื่องหมาย { }
default	เป็นบล็อกให้ดำเนินการก่อนออกจาก switch-case ในกรณีที่ค่าใน
	variable มีค่าไม่ตรงกับvalue ของ case ใดๆ
break	ทำหน้าที่ปิดงานของแต่ละ case ยกเว้นในส่วนของ default ไม่ต้อง
	ใช้ break

ตัวอย่างการใช้ switch-case (ภาพที่ 4.9)



ภาพที่ 4.9 แสดงการกำหนดทางเลือกด้วย switch-case

ตัวอย่างการใช้งาน switch-case (ภาพที่ 4.10)

```
import javax.swing.JOptionPane;
class Ex4_10
{
  public static void main (String [] args)
{
    int choice;
    do {
       System.out.println("Item For Help Syntax: ");
       System.out.println(" 1. If");
       System.out.println(" 1. If");
       System.out.println(" 2. Switch");
       System.out.println(" 3. While");
       System.out.println(" 4. Do .. While");
```

System.out.println(" 5. For n); System.out.println("Choose One"); String input = JOptionPane.showInputDialog("Enter Item (1-5)"); choice =Integer.parseInt(input); } while (choice $< 1 \parallel$ choice > 5); System.out.println ("\n"); switch (choice) { case 1: System.out.println ("The if:\n"); System.out.println ("if (condition) statement;"); System.out.println ("else statement;"); break: case 2: System.out.println ("The switch:\n"); System.out.println ("switch (expression){"); System.out.println ("statement sequence"); System.out.println ("break; "); System.out.println (" || ... "); System.out.println ("}"); break; case 3: System.out.println ("The while:\n"); System.out.println ("while(condition) statement;"); break; case 4: System.out.println ("The do ... while:\n"); System.out.println ("do { "); System.out.println (" statement; "); System.out.println ("} while conditions); "); break; case 5: System.out.println ("The for:\n"); System.out.println ("for(int; condition; iteration)"); System.out.println (" statement; "); break: } }

}

76

จากโปรแกรมตัวอย่างในภาพที่ 4.10 เมื่อทำการรันโปรแกรม ตัว InputDialog จะรอรับค่าที่ให้ทำงาน ตาม case ที่เลือก ซึ่งตัวอย่างนี้เป็นการกำหนดค่าของ variable เป็นเลขจำนวนเต็ม (integer) คือ 1-5 ในภาพที่ 4.10 เป็นการเลือกค่า value ของ case เท่ากับ 2 จะให้ผลการทำงานดังนี้

23	case 2:			
24	System.out.println ("The switch:\n");			
25	System.out.println ("switch (expression){");	Input		
26	System.out.println ("statement sequence");	P Enter Item (1-5)		
27	System.out.println ("break; ");			
28	System.out.println (" ");	OK Cancel		
29	System.out.println ("}");			
30	break;			
Image: Constraint of the system Jeam For Help Syntax: 1. If 2. Switch 3. While 4. Do While 5. For Choose One				

ภาพที่ 4.10 แสดงการใช้ switch-case ด้วยค่า variable ชนิดข้อมูล integer ได้ผลการทำงานของโปรแกรมของภาพที่ 4.11 ดังนี้

23	case 2:				
24	System.out.println ("The switch:\n");				
25	System.out.println ("switch (expression) {");				
26	System.out.println ("statement sequence");				
27	System.out.println ("break; ");				
28	System.out.println (" ");				
•29	System.out.println ("}");				
30	break;				
31	case 3:				
Choose	: One				
The sv	ritch:				
switch statem break; }	(expression) { ent sequence				



ประโยคการวนลูป (Loop Statements)

การเขียนโปรแกรมให้ทำการวนรอบหรือวนลูป เป็นวัตถุประสงค์หลักของการนำโปรแกรม คอมพิวเตอร์มาช่วยการทำงานของมนุษย์ ที่เป็นงานซ้ำๆ ซึ่งเป็นความสามารถที่จำเป็นมากต่อการ พัฒนาโปรแกรม การเขียนโปรแกรมด้วยการใช้ประโยคการวนลูป ผู้เขียนโปรแกรมสามารถเลือกใช้งาน 3 ลักษณะคือ

ประโยค for (for statement) ประโยค while (while statement) ประโยค do/while (do/while statement)

ประโยค for

การใช้ประโยค for ในการให้โปรแกรมทำงานซ้ำๆ เหมาะสำหรับผู้เขียนโปรแกรมต้องทราบ จำนวนรอบที่ต้องการให้ทำงานซ้ำจำนวนกี่รอบที่แน่นอน มีโครงสร้างประโยคดังนี้

for (<initial exp> ; <condition exp> ; <update exp>) <statements>;

เมื่อ

<initial exp=""></initial>	เป็น ค่าที่ใช้กำหนดค่าเริ่มต้นของรอบที่ต้องการให้ทำงานซ้ำ ซึ่ง สามารถกำหนดให้เป็นค่าผ่านตัวแปร หรือ กำหนดให้อยู่ในรูปของ สมการ สามารถกำหนดได้
<condition exp=""></condition>	เป็น เงื่อนไขในการตรวจสอบจำนวนรอบว่าครบตามจำนวนที่
	ดำเนินการใน statement
<statements></statements>	เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง คำสั่งต้องเขียนภายใต้เครื่องหมาย { }
<update exp=""></update>	จะถูกเรียกให้ประมวลผลต่อจากการทำงานของ statement และนำ ค่าตัวแปรเริ่มต้นที่ปรับปรุงค่า (update) แล้วไปทำการทดสอบ เงื่อนไขใน <condition exp=""> อีก และทำงานซ้ำเช่นนี้จนกว่าค่าใน <condition exp=""> มีค่าเป็น false โปรแกรมก็จะหลุดจากลูปไป ทำงาน ในประโยคที่ต่อจาก statement แสดงโพชาร์ตการทำงาน (ภาพที่ 4.12)</condition></condition>



ภาพที่ 4.12 แสดง flow chart ของ for loop

ตัวอย่างการใช้งาน for loop (ภาพที่ 4.13)

```
public class Ex4_13 {
 1
   public static void main(String args[]){
 2
 3 int i, f = 1;
       for (i=1; i<=10 ; i++ ) {
 4
         f *= i;
 5
         System.out.println( "! "+i +"=" +f); }
 6
 7
         }
 8
   }
•
   ----- JavaRun ------
! 1=1
! 2=2
! 3=6
 4=24
 5=120
 6=720
 7=5040
 8=40320
 9=362880
 10=3628800
```

ภาพที่ 4.13 แสดงการทำงานของประโยค for loop

ตัวอย่างการโปรแกรมด้วยการใช้งานของประโยคคำสั่ง for loop ด้วยการใช้ InputDialog ในการรับค่า และแสดงค่าด้วย MessageDialog พร้อมการคำนวณค่าเฉลี่ยปริมาณน้ำฝน จำนวน 12 เดือน ที่ผู้ใช้ โปรแกรมป้อนค่า ผ่านทาง InputDialog ตามความต้องการ (ภาพที่ 4.14)



ภาพที่ 4.14 แสดงประยุกต์ใช้งาน for loop

ประโยค while (while statement)

การใช้ประโยค while ใช้ในการควบคุมประโยคที่ทำให้โปรแกรมทำงานซ้ำๆ เหมาะสำหรับ ผู้เขียนโปรแกรม**ไม่**ต้องทราบจำนวนรอบที่ต้องการให้ทำงานซ้ำจำนวนกี่รอบที่แน่นอน ให้ทำงานซ้ำไป จนกว่าเงื่อนไขไม่ถูกต้อง มีโครงสร้างประโยคดังนี้

while (<boolean expression>) <statements>;

เมื่อ

- <boolean expression> เป็น เงื่อนไขในการตรวจสอบว่าถูกต้องหรือไม่ ซึ่งถ้าผลการ ตรวจสอบเป็น true โปรแกรมจะไปดำเนินการใน statement แต่ ถ้าผลการตรวจสอบเป็น false โปรแกรมจะไปดำเนินการใน นอก บล็อกของ statements
- <statements> เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง คำสั่งต้องเขียนภายใต้เครื่องหมาย { } แสดงโพชาร์ตการทำงานของ while (ภาพที่ 4.15)



ภาพ 4.15 แสดงโพชาร์ตการทำงานของ while

ตัวอย่างการใช้งาน while loop (ภาพที่ 4.16)

1	class Ex4_16 {
2	<pre>public static void main (String[] args) {</pre>
3	int x = 1;
4	int total = 0;
5	while (x <= 10) {
6	total $+= x;$
7	$\mathbf{x} = \mathbf{x} + 1;$
8	}
9	System.out.println ("The series from 1 to 10 is " + total);
10	}
11	}
•	
The	JavaRun series from 1 to 10 is 55

ภาพที่ 4.16 แสดงการใช้งาน while loop

ตัวอย่างการใช้งาน while loop (ภาพที่ 4.17)



ภาพที่ 4.17 แสดงการใช้ while loop ทำการนับรอบการทำงาน

ประโยค do...while (do...while statement)

ประโยคการวนรอบแบบนี้ จะเป็นการทำงานในลูปรอบแรก ก่อนแล้วจะตรวจสอบเงื่อนไขที่ คำสั่ง while หลังจากที่ทำงานเสร็จแล้ว ดังนั้นลูปจะวนอย่างน้อย หนึ่งรอบเสมอ ซึ่งแตกต่างกับ ประโยควนรอบด้วย while โดยประโยคการวนรอบ do...while จะทำการตรวจสอบเงื่อนไขเป็นจริง หรือไม่ถ้าเป็นจริงก็จะกลับขึ้นไปทำงานที่คำสั่งต่างๆ ใหม่ แต่ถ้าไม่เป็นจริงก็จะหลุดลูป ออกมาทำงานที่ คำสั่งต่อไป ซึ่งโครงสร้างประโยค do...while ดังนี้

do {

<statements>; } while (<boolean expression>); doเป็นการสั่งให้ทำงานเริ่มต้นก่อนหนึ่งรอบ ตามบล็อก statement<statements>เป็นบล็อกที่ดำเนินงานของลูปในการทำงานซ้ำ เป็นกลุ่มคำสั่งในการ
แก้ปัญหาของโจทย์ต้องเขียนภายใต้เครื่องหมาย { }while <boolean expression> เป็น เงื่อนไขในการตรวจสอบว่าถูกต้องหรือไม่ ซึ่งถ้าผลการ
ตรวจสอบเป็น trueโปรแกรมจะไปดำเนินการใน do
<statement> ไปเรื่อยๆ จนตรวจสอบแล้วผลเป็น false โปรแกรม
จะไปดำเนินการนอกบล็อกของ statements

ทำงานของ do...while (ภาพที่ 4.18)



ภาพที่ 4.18 แสดงโพชาร์ตการทำงานของ do...while

ตัวอย่างการใช้งาน do...while (ภาพที่ 4.19)



ภาพที่ 4.19 แสดงใช้งาน do...while ในการหาผลรวม

เมื่อ

หมายเหตุ ในการกำหนดให้การแสดงผลการทำงานเป็นภาษาไทย สามารถทำการกำหนดความสามารถ ของ edit plus (ภาพที่ 4.20 – 4.21)

🚰 EditPlus - [Ex4_19.java *]						
💋 File Edit	💋 File Edit View Search Document Project Tools Window Help					
🗂 🛩 🔛 🦉	$ \begin{tabular}{ c c c c } \hline \begin{tabular}{ c c c } \hline \begin{tabular}{ c c } \hline tabul$					
Directory Clipte		Screen Font	+	Default (AngsanaUPC, 18)		
[D:]		Printer Font	•	Custom 1 (Arial, 9)		
 □ D:\ □ Skul □ EdPlan □ BasicJavaF □ Book □ CodeJava 	✓	Tabs and Spaces Line Breaks URL Highlighting Syntax Highlighting	Alt+Shift+I Alt+Shift+L Alt+Shift+U Alt+Shift+S	Custom 2 (Verdana, 8) Custom 3 (Times New Roman, 9) Custom 4 (MS Sans Serif, 8) Custom 5 (Terminal, 10)		
	✓	Cursor Indicator	Alt+Shift+C	Set Font		
		Ruler	Alt+Shift+R			
Ex4_5.java Ex4_6.java	<	Line Number	Ctrl+Shift+L			
Ex4_7.java		Full Screen	Alt+Shift+0			
Ex4_MUL.java first java		View in Browser	Ctrl+B	เสรามพย : " + sum) ;		

ภาพที่ 4.20 แสดงการเลือกรายการคำสั่งในการกำหนด Screen Font

Categories General Fonts	Area: Output window	
Colors	Font	Font style Size
- Print	AngsanaUPC	Regular 16
Files Settings & syntax Templates Project Tools User tools Keyboard Spell checker Toolbar	Angsana UPC Aparagica Arabic Typesetting Arial Cypesetting Arial Black Arial Black Arial Narrow Arial Rounded MT Bold Arial Unicode MS Amo Pro Amo Pro Caption Amo Pro Display Amo Pro Septed	Regular 16 Italic 18 Bold 20 Bold Italic 22 24 26 Language : Thai
	Amo Pro Smbd Caption Amo Pro Smbd Display -	

ภาพที่ 4.21 แสดงการเลือกรายการคำสั่งในการกำหนด Screen Font

จากภาพที่ 4.21 เป็นการกำหนดให้การแสดงผลการทำงาน (JavaRun) ดังนั้นต้องทำการเลือก ใน Area ให้หัวเรื่อง Output window แล้วทำการเลือก Font ภาษาไทยที่ต้องการกำหนด แนะนำให้ เลือก font ที่มีมาตรฐานทั่วไป เช่น AngsanaUPC หรือ AngsanaNew เป็นต้น และสามารถกำหนด ขนาดของตัวอักษรได้ในหัวข้อ size ดังตัวอย่างกำหนดให้มีขนาดของตัวอักษร 16 point แล้วทำการ คลิด Apply

ตัวอย่างการใช้งาน do...while (ภาพที่ 4.22)

```
public class Ex4_22{
   1
      public static void main(String[] args){
  2
  3
         int counter = 5;
         int factorial = 1;
  4
  5
      do {
          factorial *= counter --; /* Multiply, then decrement. */
  6
            } while (counter > 0);
  7
         System.out.println("Factorial 5 is "+factorial);
  8
         }
  9
 10 \}
•
     -- JavaRun ---
Factorial 5 is 120
```

ภาพที่ 4.22 แสดงการใช้งาน do...while ในการประยุกต์การหาค่าแฟคโตเรียล 5

ประโยคการควบคุมอื่นๆเพิ่มเติม (other control statements)

ประโยคในการควบคุมทิศทางกำหนดให้โปรแกรมทำงานของผู้เขียนโปรแกรม นอกเหนือจาก วิธีในการกำหนดผ่านทางเงื่อนไข ผ่านทางการวนลูป เป็นหลักแล้วยังมีประโยคที่น่าสนใจเพิ่มเติม ที่ สามารถนำมาช่วยในการแก้ปัญหาในการเขียนโปรแกรมที่ไม่ปกติได้ เมื่อต้องการกระโดด (jump) ออก จากบล็อกของลูปการทำงานปกติ ไปยังจุดการทำงานจุดอื่นได้ ประโยคดังกล่าวประกอบด้วย ประกอบด้วย break, continue และ return มีรายละเอียดดังนี้

ประโยค break

ประโยค break เป็นการสั่งให้โปรแกรมหยุดทำงาน โดยประโยค break ต้องอยู่ภายในบล็อก ของลูปการวนรอบเท่านั้น และเมื่อโปรแกรมถูกหยุดทำงานแล้วโปรแกรมจะต้องสั่งให้ไปทำงานยังจุดที่มี ชื่อ label ของลูปที่ตรงกับชื่อ label ที่อยู่หลัง break ชื่อของ label ที่อยู่หน้าลูปจะต้องอยู่หลัง เครื่องหมาย : (โคลอน) โดยโครงสร้างประโยค break เป็นดังนี้

break [<label>];

เมื่อ

<label> เป็นชื่อของบล็อกของลูป ซึ่ง <label> อยู่ภายใต้เครื่องหมาย [] หมายความว่า หลัง break อาจมีชื่อของบล็อกหรือก็ได้ แต่ถ้าในบล็อกลูปนั้นมีประโยค break แนะนำต้องมีชื่อ label สำหรับให้การทำงานของ break เป็นไปอย่างสมบูรณ์ไม่เกิดปัญหาการวนลูป ไม่มีที่จบของโปรแกรม (ภาพที่ 4.23)

```
class Ex4_23 {
 1
       public static void main(String[] args) {
 2
         firstloop: for (int x = 0; x < 5; x++)
 3
              secondloop: for (int y=0;y<5; y++) {
 4
                 if(x == 1 \&\& y == 1)
 5
                   break firstloop;
                                                                           End Loop
 6
                      System.out.println("x = "+x + ", y = "+y);
 7
 8
              }
 9
              System.out.println("End Loop");
         }
10
       }
11
```

จากภาพที่ 4.23 ประโยค label แรกชื่อ firstloop ประโยค for ทำงานกำหนดค่าให้กับ x เริ่ม จาก x= 0 จนถึง x = 4 โดยรอบแรก x= 0 และมาทำงานที่ประโยค label ตัวที่สองชื่อ secondloop ประโยค for เริ่มทำงานโดยค่า y เริ่มมีค่า 0 และมาเข้าตรวจเงื่อนไข if ยังไม่เป็นจริง (ถ้าเป็นจริงค่าของ x=1 และ y=1) ดังนั้นจึงมาทำคำสั่งแสดงผลในบรรทัดที่ 7 ทำให้โปรแกรมแสดงค่าของ y=0 จนถึง y = 4 โดยค่าของ x ยังมีค่าเป็น 0

และเมื่อค่า y ถูกเพิ่มค่าเป็น 5 ทำให้เงื่อนไขใน secondloop เป็นเท็จ โปรแกรมจึงไปทำใน firstloop คือกำหนดค่า x = 1 และมาเริ่มต้นทำงานใน secondloop กำหนดค่า y = 0 และมาเข้า ตรวจเงื่อนไข if ยังไม่เป็นจริง (ถ้าเป็นจริงค่าของ x=1 และ y=1) จึงแสดงค่า x=1 , y =0 จากนั้น y ถูก เพิ่มค่าอีกหนึ่งทำให้ y = 1 และมาเข้าตรวจเงื่อนไข if ให้ผลเป็นจริง (ถ้าเป็นจริงค่าของ x=1 และ y=1) มีผลทำให้ประโยค break firstloop ทำงานคือสั่งให้ออกจาก firstloop กระโดดมาทำในคำสั่งบรรทัดที่ 9 คือแสดงค่าคำว่า End Loop ออกสู่หน้าจอภาพดังหัวข้อ JavaRun

ประโยค continue

ประโยค continue เป็นการสั่งให้โปรแกรมทำงานต่อไปยังอีกคำสั่ง หรืออีกประโยค ซึ่งอาจมี ผลทำงานหยุดการซ้ำของลูปได้ โดยประโยค continue มีโครงสร้างคล้ายกับประโยค break โดย โครงสร้างประโยค continue เป็นดังนี้

continue [<label>];

เมื่อ

<label> เป็นชื่อของบล็อกของลูป ซึ่ง <label> อยู่ภายใต้เครื่องหมาย [] หมายความว่า หลัง continue อาจมีชื่อของบล็อกหรือก็ได้ แบบเดียวกับประโยค break ตัวอย่างโปรแกรม(ภาพที่ 4.24)

```
class Ex4 24 {
1
                                                                      avaRun
      public static void main(String[] args) {
2
                                                                x = 0
         for (int x = 0; x < 5; x++) {
3
              if(x == 2)
4
                                                                End Loop
                   continue;
5
                      System.out.println(" x = "+x);
6
              }
7
              System.out.println(" End Loop");
8
         }
9
10
       }
```

จากภาพที่ 4.24 การทำงานของลูป for จะทำการกำหนดค่าเริ่มต้นที่ให้ x= 0 และเพิ่มค่าที่ หนึ่งไปเรื่อยๆ และก่อนทำการแสดงค่าของ x ออกสู่หน้าจอภาพ โปรแกรมมีประโยคการตรวจสอบค่า ของ x ว่ามีค่าเท่ากับ 2 หรือไม่ ถ้าไม่ก็แสดงค่าปกติ คือพิมพ์ค่า x เท่ากับ 0 พิมพ์ค่า x = 1 และเมื่อค่า ของ x =2 ทำให้ ประโยคการตรวจสอบค่าของ x ว่ามีค่าเท่ากับ 2 หรือไม่ เป็นจริง มีผลทำให้ โปรแกรม ทำงานในประโยค continue คือให้ทำการเพิ่มค่าของ x ต่อเนื่องไป โดยข้ามคำสั่งการแสดงผลของ x ออกสู่หน้าจอภาพ มีผลทำให้ ค่าที่ x= 2 ไม่มีการแสดงค่าออกมา แต่มีการเพิ่มค่าและแสดงค่าในลำดับ คำสั่งถัดไป กล่าวคือ ผลก็จะแสดงค่า x = 3 และ x = 4 ในลับถัดมา และออกจากลูป for มาทำคำสั่ง แสดงคำว่า End Loop เป็นการจบโปรแกรม

ประโยค return

ประโยค return เป็นการสั่งให้โปรแกรมทำการส่งค่าของการคำนวณในบล็อกที่มีคำสั่ง return อยู่ให้กับ method (ในภาษาโครงสร้างหรือภาษาซี หรือภาษาซีพลัสพลัส เรียกว่า ฟังก์ชัน) ก่อนหน้าที่ ได้เรียก method ที่มีคำสั่ง return เป็นส่วนหนึ่งก่อนจบของ Method โดยโครงสร้างประโยค return เป็นดังนี้

return [<expression>];

เมื่อ

<expression> เป็นค่าผลของการทำงานใน method ที่ถูกเรียก และถูกส่งไปให้กับ method หลัก ซึ่งโดยมาตรฐานในการสร้าง method ถ้าไม่มีการคืนค่า (return) ให้กับงานใด method นั้นๆ ต้องมีคำนำหน้าว่า void เป็นคำสงวน (keyword) ว่า method นี้ไม่มีการคืนค่า (ภาพที่ 4.25)

```
class Ex4 25 {
  1
       public static void main(String[] args) {
 2
          for (int x = 0; x < 5; x++) {
 3
               if(x == 2)
 4
 5
                  return;
                  System.out.println(" x = "+x);
 6
               }
 7
 8
               System.out.println(" End ");
             }
 9
          }
10
    -- JavaRun ------
x = 0
x = 1
```

ภาพที่ 4.25 แสดงโปรแกรมที่ไม่มีการคืนค่า

จากตัวอย่างโปรแกรมภาพที่ 4.25 พบว่า method void main ทำงานค่า x = 0 จนถึง x =1 และเมื่อค่า x = 2 โปรแกรมจะทำการคืนค่า ออกจาก method void main มีผลทำให้คำสั่งการ แสดงผลในบรรทัดที่ 8 ไม่มีการทำงาน และโปรแกรมทำการจบโปรแกรมเพราะออกจาก method void main เป็นที่เรียบร้อยแล้ว

ตัวอย่างโปรแกรม

}

ตัวอย่าง 4.1 เป็นตัวอย่างโปรแกรมการกำหนดแสดงข้อความ GoodMorning เมื่อรับค่า 0-11 แสดง ข้อความ GoodAfterNoon เมื่อรับค่า 12-18 และแสดงข้อความ GoodEverning เมื่อรับค่า 19-24 import javax.swing.JOptionPane; public class lab4 1

```
public static void main(String args[])
       {
       float CurTime:
       String input = JOptionPane.showInputDialog("Enter o'clock time(1-24)");
       CurTime =Float.parseFloat(input);
       String greet;
                      if (CurTime > 0 \& CurTime < 12)
                                      greet = "GoodMorning";
                      else if (CurTime >=12 && CurTime <= 18)
                                      greet = "GoodAfterNoon";
                      else
                                     greet = "GoodEverning";
       JOptionPane.showMessageDialog(null,"Greeting Time: " + greet);
       System.exit(0);
       }
}
ตัวอย่าง 4.2 โปรแกรมการใช้ for loop ให้นักศึกษาทำการวิเคราะห์โปรแกรม ว่าสามารถนำไปใช้งาน
้ได้หรือไม่ ให้นักศึกษาทำการปรับปรุงโค๊ดให้สามารถ run ได้ ตามที่นักศึกษาต้องการ
public class Lab4 2
public static void main(String args[]) {
       for (int count=0; count < 5; count++)
               System.out.println( "count = "+count--);
               }
```

ตัวอย่าง 4.3 โปรแกรมการใช้ do...while loop ให้นักศึกษาทำการวิเคราะห์โปรแกรม ว่าสามารถ นำไปใช้งานได้ด้านใด ให้นักศึกษาทำการปรับปรุงโค๊ดให้สามารถ run ได้ ตามที่นักศึกษาต้องการ

```
public class Lab4 3
{
public static void main(String[] args)
       {
         int n = 12345;
         int t,r = 0;
         System.out.println("The original number : " + n);
       do
       {
                 t = n % 10;
                 r = r * 10 + t;
                 n = n / 10;
       }
       while (n > 0);
               System.out.println("The reverse number : " + r);
        }
}
```

โจทย์ท้าทายนักศึกษา

ตัวอย่าง 4.4 จงทำการแก้ไขข้อผิดพลาดของโปรแกรมด้านล่างให้สมบูรณ์

if (sex = "M" or sex = "m") return "height – 110"; else return "height – 100";

ตัวอย่าง 4.5 จงทำการแก้ไขข้อผิดพลาดของโปรแกรมด้านล่าง พร้อมอธิบาย

if (score >= "50") System.out.println("Not Pass"; System.out.println("Pass";

ตัวอย่าง 4.6 จงทำการแก้ไขข้อผิดพลาดของโปรแกรมด้านล่าง พร้อมอธิบาย

char Size	
switch (Size)	
case S :	price = 100;
case M :	price = 200;
case L =	price = 300;
case XL =	price = 400

ตัวอย่าง 4.7 จงทำการแก้ไขข้อผิดพลาดของโปรแกรมด้านล่าง พร้อมอธิบาย

for (x =1 ; x <12; X ++) for (y =1 ; y < 12 ; Y++); Systen.out.println(x , y);

ตัวอย่าง 4.8 จงทำการแก้ไขข้อผิดพลาดของโปรแกรมด้านล่าง พร้อมอธิบาย

while (x : 10) int x += 1; System.out.println(N); ้ตัวอย่าง 4.9 จงทำการแก้ไขประโยค for ต่อไปนี้ให้เปลี่ยนเป็น do-whlie พร้อมอธิบาย

```
class For_DoWhile
{
    public static void main(String[] args)
        {
        for ( int i = 0 ;i < 12; i ++ )
            {
            for ( int j = 12 ;j > 1; j -- )
                System.out.println("i x j = "+ i*j );
                }
        }
}
```

ตัวอย่าง 4.10 จงเขียนโปรแกรมคำนวณภาษีเงินรายได้ส่วนบุคคลโดยมีเงื่อนไข ดังนี้ ถ้าบุคคลมีเงิน รายได้มากกว่า 1 ล้านบาท บุคคลนั้นต้องเสียภาษี 30 % ถ้าบุคคลมีเงินรายได้อยู่ระหว่าง 5 แสนถึง 1 ล้าน บุคคลนั้นต้องเสียภาษี 20 % และถ้าบุคคลมีเงินรายได้น้อยกว่า 5 แสน บุคคลนั้นต้องเสียภาษี 10 %

้ตัวอย่าง 4.11 จงเขียนโปรแกรมเพื่อแสดงจอภาพผลลัพธ์ดังนี้ โดยคำสั่ง loop ชนิดใดชนิดหนึ่ง

```
* * * * * * * * * * * * *
* * * * * * * * * *
* * * * * * * * *
* * * * * * *
* * * * *
* * * *
* * * *
* * *
```

ตัวอย่าง 4.12 จงเขียนโปรแกรมที่ให้ผู้ใช้โปรแกรมกำหนดคะแนนผ่านทางคีย์บอร์ดได้ เพื่อนำคะแนนไป ทำการคำนวณเกรดและแสดงผลลัพธ์ โดยมีเงื่อนไขอยู่ว่า

a, a,	U
ถ้าคะแนน ตั้งแต่ 80 ขึ้นไป	ได้เกรด A
ถ้าคะแนนอยู่ระหว่าง 70 ถึง 79	ได้เกรด B
ถ้าคะแนนอยู่ระหว่าง 60 ถึง 69	ได้เกรด B
ถ้าคะแนนอยู่ระหว่าง 50 ถึง 59	ได้เกรด B
ถ้าคะแนนน้อยกว่า 50	ได้เกรด F

้ตัวอย่าง 4.13 จงเขียนผลการทำงานของโปรแกรมด้านล่าง และอธิบายว่าได้ผลการทำงานมาได้อย่างไร

```
class Practice
{
        public static void main(String []args)
        {
                int i = 0, j = 10;
                BreakPoint: for (;;)
                {
                         i ++;
                        System.out.println("i = " + i + ", j = " + j);
                         for(::)
                                 if(i > j--)
                                         break BreakPoint;
                }
                System.out.println("i = " + i + ", j = " + j);
        }
}
```

สรุป

การสร้างโปรแกรมคอมพิวเตอร์นอกจากการกำหนดค่าต่างๆ มาทำการคำนวณแล้ว การให้ โปรแกรมมีความฉลาดมากขึ้น จำเป็นต้องโปรแกรมคอมพิวเตอร์สามารถช่วยในการตัดสินใจ ดังนั้น โปรแกรมคอมพิวเตอร์จึงต้องมีความสามารถในการควบคมการทำงานของประโยคย่อยหรืองานย่อยๆ ว่าให้ทำงานอะไร ทำนานแค่ไหน หรือ ทำงานกี่ครั้ง และต้องสามารถคำนวณได้ว่าทำงานครบรอบตามที่ กำหนดให้ทำงานหรือยัง ประโยคที่ใช้ในการกำหนดการดำเนินการของโปรแกรมให้ทำงานตามเงื่อนไข ที่กำหนด ว่าให้ดำเนินการงานใด เมื่อใด จำนวนเท่าใด สามารถแบ่งรูปแบบการควบคุมการทำงานได้ 3 รูปแบบ คือ 1) ประโยคตามลำดับ (Sequence Statements) 2) ประโยคการตัดสินใจ (Decision Statements) และ 3) ประโยคการวนลูป (Loop Statements) โดยประโยคการตัดสินใจ เป็นการ ตัดสินใจภายใต้เงื่อนไขที่กำหนดไว้ว่าถ้าเงื่อนไขถูกต้อง ประโยคการตัดสินใจมี 2 รูปแบบ คือ ประโยค if-else และประโยค switch-case ส่วนการวนรอบหรือวนลูป เป็นการทำงานซ้ำๆ ประโยคการวนลูป สามารถเลือกใช้งาน 3 ลักษณะ คือ 1) ประโยค for (for statement) 2) ประโยค while (while statement) และ 3) ประโยค do/while (do/while statement) นั่นเอง

แบบฝึกหัด

- 1. จงอธิบายความแตกต่างในการใช้ประโยคตัดสินใจระหว่าง if กับ if-else
- 2. จงเขียนประโยชน์ของประโยค if กับ if-else มาอย่างละประเด็น
- 3. จงอธิบายการทำงานของประโยคการตัดสินใจ

Rich = (Money > 1000000) ? "Y" : "N";

- 4. จงอธิบายความแตกต่างในการใช้ประโยคตัดสินใจระหว่าง if-else กับ switch-case
- 5. จงบอกข้อจำกัดของตัวแปรใน switch ที่ใช้ในการทำงานร่วมกับ case
- 6. break ทำหน้าที่ใดในประโยคของ switch-case
- 7. default ทำหน้าที่ใดในประโยคของ switch-case
- 8. จงอธิบายความแตกต่างระหว่างประโยค for กับ while
- 9. จงอธิบายความแตกต่างระหว่างประโยค while กับ do/while
- 10. จงอธิบายความแตกต่างระหว่างประโยค continue กับ return

เอกสารอ้างอิง

กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพฯ: สำนักพิมพ์ เคทีพี.

รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.

วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

. (2545). Java Programming Volume II. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/

<u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). The J2EE™ 1.4 Tutorial. Network Circle, CA: Sun Microsystem, Inc.

แผนบริหารการสอนประจำบทที่ 5

หัวข้อเนื้อหา

คลาส

class name การสร้างคลาสจากคลาสเดิม แพ็กเกจ อ็อบเจกต์ ประกาศ instance class ประกาศ instance สร้าง อ็อบเจกต์ การเรียกใช้งานแอตทริบิวต์ การส่งข้อมูลด้วยค่าของตัวแปร การส่งข้อมูลด้วยการอ้างอิงของวัตถุ คอนสตรัคเตอร์ สแตติคดาต้าเมมเบอร์ สแตติคเมธอดเมมเบอร์ สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อผู้เรียนได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกทำงานระหว่างคลาสกับแพ็กเกจ
- 2. อธิบายการกำหนดการสร้างอ็อบเจกต์
- 3. บอกการสร้างคอนสตรัคเตอร์
- 4. มีทักษะการโปรแกรมภาษาจาวาด้วยการกำหนดคลาสแลtอ็อบเจกต์
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนด

วิธีการสอนและกิจกรรมการเรียนการสอน

- 1. วิธีสอน
- 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อคลาส แพ็กเกจ อ็อบเจกต์ คอนสตรัคเตอร์ สแตติคดาต้าเมมเบอร์ และสแตติคเมธอดเมมเบอร์
- 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวา หัวข้อคลาส แพ็กเกจ อ็อบ เจกต์และส่วนเกี่ยวข้อง
- 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์
- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 5
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สังเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 5 คลาสและอ็อบเจกต์ (Class and Object)

ในการเขียนโปรแกรมภาษาเชิงวัตถุ เป็นการเขียนโปรแกรมโดยนำสิ่งแวดล้อมรอบตัวมาเป็น ต้นแบบในการคิดกำหนดข้อมูล กำหนดหน้าที่ของสิ่งๆนั้นว่ามีความสามารถทำอะไรได้บ้าง ซึ่งธรรมชาติ สิ่งรอบตัวของมนุษย์ ของหนึ่งสิ่งจะประกอบไปด้วยข้อมูลของสิ่งนั้นๆว่ามีความแตกต่างกับสิ่งอื่นอย่างไร ถือว่าเป็นคุณสมบัติของสิ่งนั้น รวมทั้งสิ่งต่างๆ ในชีวิตจริงจะมีปฏิสัมพันธ์ กันระหว่างวัตถุ รวมทั้งของใน แต่ละสิ่งจะมารวมกันเป็นกลุ่มหนึ่ง เพื่อกำหนดให้มีชื่อๆ หนึ่งที่มีความเป็นเฉพาะของสิ่งนั้นๆ ดังนั้นการ เขียนโปรแกรมเชิงวัตถุ จึงเป็นการเขียนโปรแกรมที่สามารถเข้าถึงในการแก้ปัญหา โดยมีแนวคิดจาก ธรรมชาติมากำหนดเป็นหลักในการเขียนโปรแกรม ซึ่งมีองค์ประกอบสำคัญคือ คลาส แพ็กเกจ และ อ็อบเจกต์ มีรายละเอียดดังนี้

คลาส (class)

คลาส เป็นเปรียบเสมือนกับภาชนะที่รวบรวมวัตถุ(object) โดยในแต่ละวัตถุ จะประกอบด้วย หนึ่งตัวแปร (variable) หรือข้อมูลของวัตถุ และสองเมธอด (method) ความสามารถหรือพฤติกรรม ของวัตถุ หรือวัตถุนี้มีหน้าที่ทำงานอะไรได้บ้าง ยกตัวอย่าง เช่น คลาสห้องเรียนวิศวกรรม มีวัตถุจำนวน 25 วัตถุ (คน) ในวัตถุ (นักศึกษาแต่ละคน) มีความสามารถหรือหน้าที่ (เมธอด) ที่แตกต่างกัน วัตถุ 1 (นักศึกษาคนที่ 1) มีผม(ตัวแปร)สีดำ มีความสามารถหรือหน้าที่ (เมธอด) ในการคำนวณได้อย่างดีเลิศ วัตถุ2 (นักศึกษาคนที่ 2) มีผม(ตัวแปร)สีน้ำตาล มีความสามารถหรือหน้าที่ (เมธอด) ในการคำนวณได้อย่างดีเลิศ วัตถุ2 (นักศึกษาคนที่ 2) มีผม(ตัวแปร)สีน้ำตาล มีความสามารถหรือหน้าที่ (เมธอด) ในการคำนวณได้อย่างดีเลิศ วัตถุ2 (นักศึกษาคนที่ 2) มีผม(ตัวแปร)สีน้ำตาล มีความสามารถหรือหน้าที่ (เมธอด) ในการพูด ภาษาอังกฤษได้ เป็นต้น กล่าวโดยรวม ความหมายของคลาส คือเป็นที่รวบรวมอ็อบเจกต์หนึ่งอ็อบเจกต์ หรือมากกว่าหนึ่งอีอบเจกต์ หรือเรียกคลาสว่า เป็นพิมพ์เขียว (blue print) ในการสร้างอ็อบเจกต์ โดย ในแต่อ**อบเจกต์**จะประกอบด้วย **ตัวแปร** และ **เมธอด** ซึ่งในแต่ละคลาสที่ทำการเก็บอ์อบเจกต์นั้นๆมี ความแตกต่างกัน ก็จะทำให้คลาสมีคุณสมบัติ หรือพร้อบเพอตี้ (properties) หรือโปรแกรมบางภาษา เรียกว่า แอตทริบิวต์ (attribute) มีความแตกต่างกันในแต่ละคลาสด้วย อย่างไรก็ดีข้อมูลที่อยู่ในคลาส ในแต่ละคลาส สามารถเรียก หรือนำมาใช้มาใช้งานภายนอกคลาสได้ทุกคุณสมบัติ โดยจะกล่าวใน ภายหลัง

โปรแกรมภาษาเชิงวัตถุ จึงเป็นการเขียนโปรแกรมที่ต้องนำคลาส มาเป็นกลไกในการโครงสร้าง และพฤติกรรมของอ็อบเจกต์ เพื่อใช้สร้างอินสแตนซ์ (instance) ของคลาสนั้นๆ เป็นการจอง หน่วยความจำสำหรับเก็บข้อมูล(variable) และพฤติกรรม(method) ของอ็อบเจกต์ ให้พร้อมใช้งานเมื่อ มีการสร้างคลาสขึ้นในครั้งแรก ดังนั้นในการสร้างคลาสขึ้นในแต่ละครั้งก็เป็นการกำหนดแต่ละ อินสแตนซ์ของคลาสว่ากำหนดสมาชิกเป็นสองประเภทคือ

ประเภทที่หนึ่ง เป็นสมาชิกที่เป็นข้อมูล (data members : filed) หรือแอตทริบิวต์ ซึ่งข้อมูล สามารถกำหนดเป็น ค่าคงที่ ตัวแปร ตัวแปรอาร์เรย์ และหรืออินสแตนซ์ของคลาส ประเภทที่สอง เป็นสมาชิกที่เป็นเมธอด (method members: function) เป็นการการทำงาน เหมือนฟังก์ชันที่ไม่มีการส่งค่าออกมาเมื่อทำงานในฟังก์ชันเสร็จ หรือเป็นฟังก์ชันที่มีการส่งค่าผลการ ทำงานออกมายังผู้เรียกใช้เมธอด (ภาพที่ 5.1)



ภาพที่ 5.1 แสดงความสัมพันธ์ระหว่าง แอตทริบิวต์ เมธอด คลาส อินสแตนซ์ และเมจเสจ

จากภาพที่ 5.1 แสดงความสัมพันธ์ของคลาสที่ประกอบด้วย แอ็ทติบิวต์ (data member) หรือ ตัวแปร กับเมธอดของคลาส ที่ได้ถูกซ่อนหรือห่อหุ้ม (encapsulation) ให้เป็นแพ็กเกจด้วยกัน เพื่อเป็น ต้นแบบในการนำไปสร้างอ็อบเจกต์ที่พร้อมใช้งาน(instance) และในการสื่อสารระหว่างอ็อบเจกต์ สามารถทำงานได้ด้วยการส่ง message ข้ามไปมาได้ด้วยการใช้คลาสมาตรฐานของจาวา เช่น System.out.println{''Hello World''}; ซึ่งอ็อบเจกต์ของ message อยู่ที่ System.out และเมธอดที่ ใช้ println โดยมีการส่งพารามิเตอร์ลงในเมธอด ด้วยข้อมูล Hello World!

การสร้างวัตถุนั้นจำเป็นต้องทำความเข้าใจกับการสร้างต้นแบบของวัตถุ หรือที่เรียกว่าคลาส โดยมีรูปแบบประกาศ class ดังนี้

```
[Modifier] class Class_name
{
    [data member;]
    [Method member;]
}
```

เมื่อ Modifier ของคลาสแบ่งเป็น 2 กลุ่มคือ

- Class Access Modifier

o Default Access หากไม่กำหนด modifier หน้าคีย์เวิร์ด class แสดงว่าคลาสนี้มี modifier เป็น default นั่นหมายความว่าคลาสใดก็ตามที่อยู่ใน package เดียวกันเท่านั้นที่ สามารถเรียกใช้งานคลาสนี้ได้

o Public Access หากกำหนดคีย์เวิร์ด public ไว้หน้าคีย์เวิร์ด class แสดงว่าคลาสนี้ สามารถนำไปสร้างอินสแตนซ์ ด้วยแพ็คเก็ตใดก็ได้เข้าถึงได้ทุกแพ็คเกจและไฟล์นามสกุล .java ต้องมีชื่อเดียวกันกับชื่อคลาส

o Private Access , protected access จะใช้ได้กับ inner class และ nested class

- Class Non Access Modifier

- o Final Class กำหนดให้คลาสนั้นไม่สามารถสืบทอดได้ เช่น String class
- o Abstract class หมายถึง คลาสที่ไม่สามารถถูกสร้างเป็นออบเจ็คได้
- o Strictfp class คือ คลาสที่มีใช้งาน floating point แบบ IEEE754

Class_name ชื่อของคลาสเป็นไปตามกฎการตั้งชื่อในภาษาจาวา นิยมขึ้นต้นด้วย ตัวอักษรตัว ใหญ่ของแต่ละคำ ถ้าชื่อคลาสมากกว่าหนึ่งคำให้เขียนติดกันโดยใช้ตัวใหญ่แยกระหว่างคำ ในหนึ่งแฟ้ม์ อาจมีการกำหนดคลาสได้มากกว่าหนึ่งคลาส เมื่อถูกคอมไพล์แล้วจะได้แฟ้มนามสกุล .class

ถ้าแฟ้มข้อมูล มีหลายคลาสเป็น default ทั้งหมด จะตั้งชื่อแฟ้มชื่อใดก็ได้ ถ้าแฟ้มนั้นมีหลาย คลาส แต่ต้องมีอยู่หนึ่งคลาสเป็น public และตั้งชื่อแฟ้มให้เหมือนกับคลาสที่ เป็น public นั้น หรือ ถ้า หากจำเป็นต้องสร้างแฟ้มโปรแกรมที่มีหลายคลาส ต้องมีคลาสหนึ่งที่มีเมธอด main() และต้องตั้งชื่อ แฟ้มให้เหมือนกับชื่อคลาสที่มีเมธอด main() อยู่

การสร้างคลาสย่อย (subclass) สามารถสร้างได้สองรูปแบบ รูปแบบที่หนึ่งเป็นการสร้างคลาส หลักและสร้างคลาสย่อยพร้อมกันเรียกว่า Nested class แสดงดังภาพ 5.2, 5.3 และรูปแบบที่สองเป็น การสร้างคลาสจากคลาสเดิม ดังหัวข้อถัดไป

ตัวอย่างการสร้าง Nested class โดยมีโครงสร้างคลาสหลักที่เป็นคลาสยานพาหนะ (Vehicle) ที่มีความสัมพันธ์กับคลาสย่อย รถสองล้อ (TwoWheel), รถสี่ล้อ (FourWheel) และรถสิบล้อ (TenWheel) (ภาพที่ 5.2)



ภาพที่ 5.2 แสดงโครงสร้างคลาสหลักกับคลาสย่อย

จากภาพที่ 5.2 เป็นโครงสร้างคลาสตามสภาพแวดล้อมความจริงของมนุษย์ ซึ่งเมื่อนำมาทำการ เขียนเป็นโปรแกรมให้อยู่ในรูปของโปรแกรมเชิงวัตถุภาษาจาวา โดยในแต่ละคลาสย่อยมีข้อมูลที่เป็น แอทติบิวต์ (สมาชิกของคลาส) ที่เป็นสีของรถ กับราคาของรถแต่ละชนิด (ภาพที่ 5.3)

1	class Vehicle {
2	static class TwoWheel extends Vehicle
3	{
4	String color = "Red";
5	$int cost = 65000;$ };
6	static class FourWheel extends Vehicle
7	{
8	String color = "Blue";
9	$int cost = 1200000; $ };
10	static class TenWheel extends Vehicle
11	{
12	String color = "Black";
13	$int cost = 5000000; $ };
14	}

ภาพที่ 5.3 แสดงโค๊ดจาวาจากโครงสร้างคลาส

จากภาพที่ 5.3 เมื่อทำการคอมไพล์โปรแกรม จาวาจะทำการสร้างคลาสหลัก 1 คลาสชื่อ Vehicle.class และคลาสย่อย จำนวน 3 คลาส มีชื่อดังนี้ Vehicle\$TwoWheel.class, Vehicle\$TenWheel.class และ Vehicle\$FourWheel.class โดยเครื่องหมาย \$ เป็นเครื่องหมาย ความสัมพันธ์ระหว่างคลาสหลักกับคลาสย่อย ตามลำดับ (ภาพที่ 5.4)

Coord P ► Searc	h Resul	ts in CodeJava 🕨 🔫		×
Organize Save s Searches might be slow i	search n non-i	ndexed locations: D:\Skul\EdPlan\BasicJavaProgramming\Book\Code	eJava. Click to add to index	() ×
★ Favorites ■ Desktop Downloads icloud Photos	* III	Vehicle\$FourWheel.class D:\Skul\EdPlan\BasicJavaProgramming\Book\CodeJava	Date modified: 16/2/2557 11:34 Size: 343 bytes Date modified: 16/2/2557 11:34	* III
 Recent Places Libraries Documents Music Pictures Videos 	Places ents	Vehicle\$TwoWheel.class D:\Skul\EdPlan\BasicJavaProgramming\Book\CodeJava Wehicle class D:\Skul\EdPlan\BasicJavaProgramming\Book\CodeJava	Date modified: 16/2/2557 11:34 Size: 340 bytes	
		Vehicle D:\Skul\EdPlan\BasicJavaProgramming\Book\CodeJava	Date modified: 10/2/253/11:34 Size: 336 bytes Date modified: 16/2/2557 11:34 Size: 330 bytes	_
30 items	÷			·

ภาพที่ 5.4 แสดง class ที่ถูกสร้างจากโค๊ดจาวา

จากภาพที่ 5.3 ในโค๊ดจาวาประกอบด้วยโครงสร้างหลักคลาส Vehicle และคลาสย่อย 3 คลาส ในแต่ละคลาสย่อยมีตัวแปรหรือ แอทติบิวต์ 2 ตัวคือ color กับ cost เช่น คลาสย่อย TwoWheel มีสมาชิกเป็นตัวแปรชื่อ color เท่ากับสีแดง และสมาชิกตัวแปรชื่อ cost เท่ากับ 65000 ส่วนคลาสย่อย FourWheel มีสมาชิกเป็นตัวแปรชื่อ color เท่ากับสีน้ำเงิน และสมาชิกตัวแปรชื่อ cost เท่ากับ 1200000 และ คลาสย่อย TenWheel มีสมาชิกเป็นตัวแปรชื่อ color เท่ากับสีดำ และสมาชิกตัวแปร ชื่อ cost เท่ากับ 500000

แต่คลาส Vehicle ยังต้องการความสามารถในการทำงานของคลาส หรือหน้าที่ของแต่ละคลาส นั่นคือ เมธอด(method) ของคลาสที่สามารถ สตาร์ท(start)ได้ วิ่ง(run)ได้ เลี้ยวซ้าย(left move)ได้ เลี้ยวขวา(right move)ได้ หยุด(stop)ได้ และจอด(park)ได้ เป็นต้น ซึ่งเมธอดดังกล่าวเป็นหน้าที่หลัก ของยานยนต์ ทุกคันดังนั้น เมธอดควรอยู่ที่คลาสหลัก ส่วนคลาสย่อยสามารถเรียก เมธอดมายังคลาส หลักได้ และเป็นหลักในการออกแบบอัลกอริธึมที่ดี ต้องทำการสร้างเมธอดหรือฟังก์ชันที่เรียกใช้บ่อยๆ ไว้เพียงชุดเดียวแล้วให้คลาสย่อย (ฟังก์ชันย่อย) ที่ต้องการใช้งานมาเรียกใช้เมื่อต้องการใช้ เมธอด ซึ่ง เป็นการออกแบบการเขียนโปรแกรมที่ดี เพราะเป็นการประหยัดพื้นที่ในการจัดเก็บ และใช้เวลาในการ ประมวลผลได้อย่างรวดเร็ว เป็นการเพิ่มประสิทธิภาพในการทำงานของเครื่องคอมพิวเตอร์อีกทางหนึ่ง ด้วย ซึ่งการสร้างเมธอดให้สามารถใช้งานร่วมกัน (ภาพที่ 5.5)



ภาพที่ 5.5 แสดงโครงสร้างคลาสหลักที่มีเมธอด

```
class VehicleCar {
 1
 2
          static class TwoWheel extends Vehicle
                                                     {
 3
            String color = "Red";
 4
            int cost = 65000;
                                  };
 5
          static class FourWheel extends Vehicle
                                                     {
            String color = "Blue";
 6
 7
            int cost = 1200000; };
         static class TenWheel extends Vehicle
 8
                                                       {
9
            String color = "Black";
            int cost = 5000000;
                                  };
11
         static void start()
                                   {
                                          System.out.println("สตาร์ทรถ" ) };
                                          System.out.println("รถกำลังวิ่ง"
12
         static void run()
                                                                                  )};
                                   {
                                          System.out.println("รถเลี้ยวซ้ำย"
13
         static void leftMove()
                                                                                 ) };
                                          System.out.println("รถเลี้ยวขวา"
14
         static void rightMove()
                                   {
                                                                              )};
15
         static void stop()
                                          System.out.println("หยุดรถ" ) };
                                   {
16
         static void park()
                                   {
                                          System.out.println("DDAID") };
17 }
```

จากภาพที่ 5.5 สามารถนำมาทำการสร้างเป็นโค๊ดภาษาจาวาได้ (ภาพที่ 5.6)

ภาพที่ 5.6 แสดงโค๊ดจาวามีการสร้างเมธอดหลัก

การสร้างคลาสจากคลาสเดิม

การสร้างคลาสจากคลาสเดิมเป็นการสร้างคลาสย่อยด้วยการอาศัยคลาสที่เป็นมาตรฐานมาเป็น ต้นแบบ เพื่อนำเมธอดของคลาสเดิมที่เป็นมาตรฐานและมีความสมบูรณ์มาใช้ประโยชน์ได้สูงสุด เป็นการ ทำให้ผู้พัฒนาโปรแกรมสามารถสร้างงาน มาแก้ปัญหาได้อย่างมาตรฐาน ถูกต้อง เช่นคลาสของ Applet เป็นคลาสที่นิยมสร้างคลาสด้วยวิธีนี้มากที่สุด การสร้างคลาสจะมีคีย์เวิร์ดคำว่า extends เป็นตัวเชื่อม ระหว่างคลาสที่สร้างใหม่กับคลาสเดิมที่มีอยู่ ซึ่งเป็นการรวมคลาสที่สร้างใหม่ไปรวมกับแพ็กเกจเดิมที่มี อยู่ตัวอย่างเช่น

public class subApplet extends java.applet.Applet

```
{
Body class;
```

}

จากตัวอย่างด้านบนเป็นโครงสร้างในการสร้างคลาสจากคลาสเดิม หรือคลาสหลักในที่เป็น คลาสที่มาจากซุปเบอร์คลาส java.applet.Applet นำมาทำการสร้างคลาสย่อยชื่อ subApplet ซึ่ง จะกล่าวรายละเอียดภายหลัง

แพ็กเกจ (package)

ในการเขียนโปรแกรมภาษาจาวา มีการพัฒนามาอย่างต่อเนื่อง มีคลาสที่อำนวยความสะดวกแก่ นักพัฒนาโปรแกรมอย่างมากมาย และได้ทำการจัดเป็นหมวดหมู่แบ่งตามการใช้งานของโปรแกรมเป็น ชุดๆ ของแต่คลาสเหมือนกับไลบรารี (library) ให้สามารถเลือกใช้งานได้ไม่ซ้ำกัน จึงเรียกการรวมของ คลาสว่า แพ็กเกจ (package) ทำให้ผู้พัฒนาโปรแกรมสามารถเรียกคลาสแต่ละแพ็กเกจมาใช้งานได้ อย่างสะดวกและรวดเร็ว ถูกต้อง โดยไม่ต้องมาทำการสร้างคลาสที่เป็นมาตรฐานที่เป็นพื้นฐานในการ พัฒนาโปรแกรม แต่สามารถนำ แพ็กเกจ มาพัฒนาหรือต่อยอดได้เพิ่มเติมได้ ทำให้สามารถสร้าง โปรแกรมได้อย่างมีประสิทธิภาพ กล่าวคือ การทำงานของโปรแกรมจะถูกต้อง รวดเร็ว และ ประหยัดเวลาในการพัฒนาโปรแกรมอย่างมากมาทำการแก้ปัญหาได้

การเขียนโปรแกรมภาษาเชิงวัตถุ ด้วยภาษาจาวามีการเรียกใช้งานคลาสที่เป็นแพ็กเกจอย่าง สม่ำเสมอ และตัวแพ็กเกจก็ได้มีการพัฒนาตามรุ่นของจาวาอย่างตลอดเวลา เพื่ออำนวยความสะดวกแก่ ผู้พัฒนาโปรแกรมได้อย่างต่อเนื่อง ดังนั้นจึงขอกล่าวแพ็กเกจที่เป็นส่วนสำคัญหรือพื้นฐานที่ผู้เริ่มต้นใน การพัฒนาโปรแกรมภาษาจาวา ควรรู้และนำมาใช้งานบางส่วนดังนี้

java.applet	เป็นแพ็กเกจที่เก็บคลาสด้าน Applet	
java.awt	เป็นแพ็กเกจที่เก็บคลาสด้าน GUI(Graphic User Interface) แยกจาก swing	
java.beans	เป็นแพ็กเกจที่เก็บคลาสด้าน beans	
java.io	เป็นแพ็กเกจที่เก็บคลาสด้าน Streams	
java.lang	เป็นแพ็กเกจที่เก็บคลาสด้านภาษา	
java.math	เป็นแพ็กเกจที่เก็บคลาสด้านการคำนวณทางคณิตศาสตร์	
java.net	เป็นแพ็กเกจที่เก็บคลาสด้านเครือข่าย	
java.rmi	เป็นแพ็กเกจที่เก็บคลาสด้าน Remote Method Invocation	
java.secuirity	เป็นแพ็กเกจที่เก็บคลาสด้าน secuirity	
java.sql	เป็นแพ็กเกจที่เก็บคลาสด้านการติดต่อสื่อสารกับฐานข้อมูล	
java.util	เป็นแพ็กเกจที่เก็บคลาสด้านเก็บ utility ให้โปรแกรมทั่วไป	
javax.crypto	เป็นแพ็กเกจที่เก็บคลาสที่ใช้ในการเข้าและถอดรหัส	
javax.media	เป็นแพ็กเกจที่เก็บคลาสด้าน multi media	
javax.servlet	เป็นแพ็กเกจที่เก็บคลาสด้าน servlet	
javax.swing	เป็นแพ็กเกจที่เก็บคลาสด้าน GUI แตกต่างกับ .awt	

ซึ่งแพ็กเกจ javax เป็นแพ็กเกจที่เพิ่มเติมมาจากแพ็กเกจมาตรฐาน java ซึ่งในปัจจุบันมีการ พัฒนาอย่างต่อเนื่องจนทำให้ภาษาจาวาเป็นที่ยอมรับจากมาตรฐานอื่นๆ ด้วยเช่น org.omg.CORBA และ org.omg.CosNaming เป็นต้น ในการเรียกใช้แพ็กเกจ ผู้ใช้งานต้องทำการอ้างถึงหรือทำการเชื่อมต่อด้วยประโยค import และตามด้วยชื่อแพ็กเกจ แต่ตัวที่ต้องการเรียกใช้ ซึ่งสามารถเรียกเป็นรายโดยระบุคลาสที่ต้องการ เช่น import java.lang.String หมายความว่าผู้ใช้ต้องการเรียกคลาส String ที่อยู่ในแพ็กเกจของ java.lang เพียงคลาสเดียวเท่านั้น แต่ถ้าผู้ใช้งานต้องการระบุแพ็กเกจทั้งชุดคือทุกคลาสที่อยู่ใน java.lang สามารถ ทำได้ด้วยการใช้ประโยค import java.lang.* คือ * เป็นการแทนชื่อคลาสทุกคลาสที่อยู่ในแพ็กเกจ java.lang ตัวอย่างโปรแกรมในภาพที่ 5.7

1	import javax.swing.JOptionPane;	
2	public class HelloWorldGUI1 {	
3 public static void main(String[] args) {		
4 JOptionPane.showMessageDialog(null, " Hel lo World ! "); }		
5	}	

ภาพที่ 5.7 แสดงการเรียกใช้ javax.swing.JOptionPane

จากภาพที่ 5.7 เป็นการเรียกใช้คลาส showMessageDialog() ที่อยู่ในแพ็กเกจของ javax.swing.JOptionPane โดยการทำการอ้างอิงก่อนเข้าส่วนของหัวโปรแกรมหลัก ด้วยประโยค import. javax.swing.JOptionPane และเมื่อทำการรัน โปรแกรมจึงแสดงคำว่า Hello Word! ตามที่ ผู้เขียนโปรแกรมกำหนดข้อความในอาร์กูเมนต์ ของเมธอด showMessageDialog()

ตัวอย่างการเรียกใช้แพ็กเกจที่มีมากกว่า 1 แพ็กเกจในหนึ่งโปรแกรม ที่สามารถทำงานร่วมกัน ได้อย่างสมบูรณ์ ดังโปรแกรม import java.awt.*;

import java.awt.event.*; import javax.swing.*; public class HelloWorldGUI2 { private static class HelloWorldDisplay extends JPanel { public void paintComponent(Graphics g) { super.paintComponent(g); g.drawString(" Hel lo World ! ", 20, 30);

} } private static class ButtonHandler implements ActionListener { public void actionPerformed(ActionEvent e) { System.exit(0); } } public static void main(String[] args) { HelloWorldDisplay displayPanel = new HelloWorldDisplay(); JButton okButton = new JButton("OK"); ButtonHandler listener = new ButtonHandler(); okButton.addActionListener(listener); JPanel content = new JPanel(); content.setLayout(new BorderLayout()); content.add(displayPanel, BorderLayout.CENTER); content.add(okButton, BorderLayout.SOUTH); JFrame window = new JFrame("GUI Tes t "); window.setContentPane(content); window.setSize(250,100); window.setLocation(100,100); window.setVisible(true); } }

ผลการทำงาน (ภาพที่ 5.8) ถ้าต้องการออกจากโปรแกรมการทำงานได้ด้วยการคลิก OK

import java.awt.*;	
2 import java.awt.event.*;	
<pre>3 import javax.swing.*;</pre>	GUI Tes t
public class HelloWorldGUI2 { private static class HelloWorldDisplay extends JPanel {	Hel lo World !
7 super.paintComponent(g);	
<pre>g.drawString("Hel lo World ! ", 20, 30);</pre>	
9 }	
0 }	

ภาพที่ 5.8 แสดงผลการทำงานการอ้างอิง java.awt.*

อ็อบเจกต์ (object)

การสร้างวัตถุจากคลาส (Object Declaration) อ็อบเจกต์ที่สร้างขึ้นจากคลาส จึงเรียกว่า สร้างอ็อบเจกต์ที่พร้อมใช้งาน (instance of class) ดังได้กล่าวมาข้างต้น มีรูปแบบในการสร้าง ประกอบด้วย 3 แบบคือ

1. ประกาศ instance class ด้วย

```
class_name Object_ identifier ;
และทำการเตรียมพื้นที่บนหน่วยความจำสำหรับ instance class ที่สร้างขึ้นด้วย
Object_ identifier = new class_name ([class_parameter]);
```

ดังตัวอย่าง

```
คลาส InsClass ประกอบด้วย
class InsClass {
int id ;
String name;
void setId(int i) {
}
int getId() {
return id;
}
}
```

ประกาศ instance ด้วยชื่อ ic1 ดังนี้

InsClass ic1;

และทำการเตรียมพื้นที่บนหน่วยความจำสำหรับ instance class ที่สร้างขึ้นด้วย lc1= new InsClass();

เป็นการทำการสร้างอ็อบเจกต์ด้วยการให้ตัวแปร ic1 ที่ประกาศอ้างอิงมายัง อ็อบเจกต์ InsClass() มีผลทำให้ ic1 สามารถเรียกใช้ตัวแปร id ตัวแปร name และเมธอด setId() เมธอด getId() ที่อยู่ในคลาส InsClass ได้ทุกประการ ดังนั้นจึงสามารถสร้างคลาสเพิ่มเติมได้ที่ละหลายๆ อ็อบเจกต์ ดังนี้

```
InsClass ic2, ic3, ic4; // ประกาศ instance ที่ละหลายๆ ตัว ชื่อ ic2, ic3 และ ic4
ic2 = new InsClass () ;
ic3 = new InsClass();
ic4 = new InsClass();
```
2. ประกาศ instance พร้อมเตรียมพื้นที่บนหน่วยความจำสำหรับ instance ที่สร้างใหม่

class name Object_ identifier = new class_name();

ตัวอย่างเช่น

InsClass ic = new InsClass ();

สร้างอ็อบเจกต์โดยไม่ประกาศตัวแปรอ็อบเจกต์เพื่อใช้อ้างอิงอ็อบเจกต์ได้

new class_name();

ตัวอย่างเช่น

new InsClass();

ดังนั้นจึงกล่าวโดยรวมได้ว่าในหนึ่งคลาสสามารถสร้างอ็อบเจกต์ใช้งานได้หลายครั้ง ด้วยการใช้ คีย์เวิร์ด new ทุกครั้งจะเป็นการประกาศขอใช้พื้นที่หน่วยความจำใหม่ และหาก instance ใดไม่ถูกใช้ งานในภาษาจาวาก็ทำให้เกิดมีการสะสมขยะ (garbage collector) ขึ้นทุกครั้งที่มีการสร้างอ็อบเจกต์จึง จำเป็นอย่างยิ่งที่ต้องมีการจัดการคืนหน่วยความจำให้กับระบบ จึงจะทำให้ภาษาจาวาเป็นภาษาที่มี คุณสมบัติของการทนทาน (robust) เนื่องจากมีตัวกำจัดขยะ(garbage collection) นั่นเอง

การเรียกใช้งานแอตทริบิวต์

แอตทริบิวต์ คือส่วนประกอบที่เป็นสมาชิกของคลาส ดังนั้นการที่จะทำงานกับแอตทริบิวต์ได้ นั้นจึงต้องกระทำผ่านตัวแปรอ็อบเจกต์ที่อ้างอิงอ็อบเจกต์ของคลาสนั้นๆ อยู่ดังนี้

Object_ identifier.attribut_name

ตัวอย่าง

ic1.id = 1001; ic1.name = "ICE"; ic2.id = 1002; ic2.name = "TRU"; ตัวอย่าง โปรแกรมการเรียกใช้งานแอตทริบิวต์ ภาพที่ 5.9 เป็นการสร้างอ็อบเจกต์ ic1 จาก InsClass โดย InsClass มีสมาชิก ที่เป็นตัวแปรคือ ic กับ name ซึ่งเมื่อ ic1 เป็นอ็อบเจกต์ที่สร้างมา จาก InsCalss จึงทำให้สามารถกำหนดตัวแปรผ่านทาง แอตทริบิวต์ได้ด้วย ic1.id = 1001 และ ic1.name = "ICE" ดังนั้นเมื่อทำการให้แสดงค่าที่ถูกกำหนดผ่านเมธอดมาตรฐาน println จึงสามารถ แสดงค่าได้ดังผลการทำงานในภาพที่ 5.9

1	class InsClass {		
2	int id ;	JavaRun	
3	String name;	Identification = 1001	
4	void setId(int i){	Name = ICE	
5	id = i; }		
6	int getId() {	Output completed (0 sec consumed) - Normal Termination	
7	return id; }		
8	}		
9	public class CallAttribute {		
10	<pre>public static void main(String args[]){</pre>		
11	InsClass ic1 = new InsClass();		
12	ic1.id = 1001;		
13	ic1.name = "ICE";		
14	14 System.out.println("Identification = " + ic1.id);		
15	<pre>15 System.out.println("Name = " + ic1.name);</pre>		
16	}		
17	}		

ภาพที่ 5.9 แสดงการสร้างอ็อบเจกต์ และการเรียกใช้แอตทริบิวต์ที่อยู่ในคลาสมาใช้งาน

จากตัวอย่างโปรแกรมภาพที่ 5.9 เป็นการที่กำหนดค่าให้กับแอทตริบิวต์ที่อยู่ในคลาสได้โดยตรง ได้โดยการตามรูปแบบของ modifier จะต้องเป็น public , protected หรือ default เท่านั้น ส่วนการ modifier ที่เป็น private นั้นจะสามารถเรียกใช้งานของแอทตริบิวต์ ของอ็อบเจกต์ในคลาสต้อง ดำเนินการผ่านเมธอดเท่านั้น ดังตัวอย่างโปรแกรมในภาพที 5.10 เป็นการกำหนดค่าให้แอทตริบิวต์ เมธอด setId() ให้กับค่าตัวแปร id มีค่าเป็น 1002 และการเรียกใช้งานของเมธอด getId() เป็นการรับ ค่าที่อยู่ในอ็อบเจกต์ คือค่า id = 1002 มาแสดงค่าในเมธอดนี้ (getId()) รวมทั้งการรับค่าของเมธอด run() มาแสดงผ่านที่เมธอดมาตรฐาน println() ดังแสดงผลการทำงานในภาพที่ 5.10

class InsClass { private int id; 2 --- JavaRun -----public void setId(int i){ 3 Identification = 1002 id = i;} 4 Print from method run() int getId() { 5 return id; } 6 Output completed (0 sec consumed) - Normal Termination void run(){ System.out.println("Print from method run() "); } 8 } 9 10 public class CallAttribute2{ 11 public static void main(String args[]){ InsClass ic2 = new InsClass(); 12 13 ic2.setId(1002); 14 System.out.println("Identification = " + ic2.getId()); 15 ic2.run(); 16 } 17 }

ภาพที่ 5.10 แสดงการเรียกเมธอดผ่านทางตัวแปรเป็น private modifier

การส่งข้อมูลระหว่างเมธอด เป็นสิ่งจำเป็นอย่างมากในการ เขียนโปรแกรมเชิงวัตถุ ด้วย โครงสร้างในการเขียนโปรแกรมที่มีการแบ่งงานทำตามหน้าที่ของแต่ละเมธอด ดังนั้นในการเรียกใช้เมธ อดจึงต้องมีการส่งค่าข้อมูลให้เมธอดที่เรียกนำค่าไปประมวลผล และเมื่อเมธอดทำหน้าที่สำเร็จก็อาจต้อง ส่งผลการทำงานมายังอ็อบเจกต์หรือคลาสหลัก เพื่อนำค่ามาทำงานต่อไป การส่งค่าข้อมูลสามารถทำได้ สองลักษณะคือ

การส่งข้อมูลด้วยค่าของตัวแปร (Pass by value)

ในการเรียกใช้งานเมธอด มีความจำเป็นบ่อยครั้งที่ต้องส่งค่าข้อมูลไปยัง เมธอดที่เรียกใช้งาน โดยผ่านทางตัวแปร ซึ่งค่าที่อยู่ในตัวแปรจะถูกส่งไปยัง ตัวแปรในพารามิเตอร์ (parameter) ของเมธอด ที่ถูกเรียกใช้งาน ตัวอย่างโปรแกรมภาพที่ 5.11

```
public class ValuePass{
   1
  2
     static void setData(int a, int b){
  3
        int z = a+b;
           System.out.println("After Pass Value A+B = "+ z);
  4
  5
         }
      public static void main (String [] args) {
  6
  7
        int x = 555, y = 111;
        System.out.println("Before Pass Value X = "+x +" Y = "+y);
  8
        setData(x, y);
  9
           }
 10
 11 }
•
 ----- JavaRun ------
Before Pass Value X = 555 Y = 111
After Pass Value A+B = 666
```

ภาพที่ 5.11 แสดงการส่งค่าข้อมูลผ่านทางตัวแปร

จากภาพที่ 5.11 เป็นการส่งค่าตัวแปร x, y ผ่านทางการเรียกเมธอด setData(x, y) มายังเมธ อด setData(int a, int b) และภายในเมธอดได้นำค่าที่ส่งเข้ามาทำการบวกกัน คือ 555+111 ได้เป็น 666 มาเก็บไว้ยังตัวแปร z จากนั้นทำการแสดงค่าของการบวกมายังหน้าจอด้วยประโยคมาตรฐาน println

การส่งข้อมูลด้วยการอ้างอิงของวัตถุ(Pass by reference)

การส่งข้อมูลด้วยการอ้างอิงของวัตถุ เป็นกรณีที่ทำการส่งค่าที่เป็นพารามิเตอร์ด้วยเมธอดที่เป็น ตัวแปรของอ็อบเจกต์ ส่งค่าไปยังอ็อบเจกต์ที่ต้องเรียกใช้ จะพบว่าค่าที่ส่งไปจะทำการเปลี่ยนค่าข้อมูล ผ่านทางเมธอด ตัวอย่างภาพที่ 5.12

1	public class PassValueObject {		
2	int data;		
3	static void setData (PassValueObject a){		
4	a.data = 555;		
5	}		
6	public static void main (String [] args) {		
7	PassValueObject a = new PassValueObject();		
8	a.data = 111;		
9	System.out.println("Before pass value Object a = "+a.data);		
10	setData(a);		
11	System.out.println("After pass value Object a= "+a.data);		
12	}		
13	}		
JavaRun			
Before pass value Object a = 111			
After	After pass value Object a= 555		

ภาพที่ 5.12 แสดงการส่งค่าข้อมูลผ่านทางเมธอด

จากภาพที่ 5.12 คลาสที่เป็นหลัก (PassValueObject) มีแอทตริบิวต์ที่เป็นตัวแปรค่า data และมีเมธอดชื่อ setData (PassValueObject a) ซึ่งมีการรับค่าพารามิเตอร์ด้วยอ็อบเจกต์ a ที่เป็น อ็อบ - เจกต์ชนิด PassValueObject โดยค่าของตัวแปรผ่านทางอ็อบเจกต์ a ด้วยประโยคเมธอด a.data = 555 ดังนั้นใน เมธอดหลัก (main) มีการเรียกใช้เมธอด setData(a) ผ่านทางอ็อบเจกต์ a ซึ่ง ค่าข้อมูลที่อยู่ในเมธอด setData (PassValueObject a) มีค่าเท่ากับ a.data = 555 มีผลทำให้การ แสดงค่าของข้อมูลที่ถูกอ้างอิงด้วยเมธอดมายังหน้าจอด้วยประโยคมาตรฐาน println ได้ผลเป็น After pass value Object a= 555 (ภาพที่ 5.12)

คอนสตรัคเตอร์ (Constructor)

คอนสตรัคเตอร์ หมายถึง เป็นการสร้างอ็อบเจกต์พร้อมกับการกำหนดค่าเริ่มต้นให้กับแอทตริ บิวต์ต่าง ๆ ของอ็อบเจกต์ที่ได้สร้างขึ้นด้วยคีย์เวร์ด new โดยคอนสตรัคเตอร์จะถูกเรียกใช้ให้ทำงาน อัตโนมัติเมื่อเราสร้างออบเจ็คขึ้น ในขณะที่สร้างแต่ละอินสแตนซ์ขึ้นมาที่มีชื่อเดียวกันกับคลาส และจะมี เมธอดที่เป็นส่วนของ modifier ของคอนสตรัคเตอร์จะมี default เป็น public ดังนั้นจึงไม่จำเป็นต้อง ใส่ modifier ไว้หน้า constructor หากจำเป็นต้องใส่ก็สามารถกำหนดเป็น public ได้เท่านั้น

ในการสร้าง อินสแตนซ์ ขึ้นด้วยการ new ที่คอนสตรัคเตอร์ของคลาสนั้น จะถูกเรียกใช้งาน คล้ายกับเมธอดทั่วไป ซึ่งอาจมีการส่งพารามิเตอร์ให้แก่คอนสตรัคเตอร์ นำไปใช้ในการกำหนดค่าเริ่มต้น ให้แก่ ค่าแอทตริบิวต์ในอินสแตนซ์นั้นได้ด้วยผลของการ new ทำให้อินสแตนซ์ที่ถูกสร้างขึ้นมี อ้างอิง (reference) ถูกส่งออกมาด้วยเสมอ มีผลทำให้คอนสตรัคเตอร์ไม่สามารถส่งค่าอย่างอื่นออกมา ดังนั้น จะมีประโยคที่ return ในคอนสตรัคเตอร์ ดังตัวอย่างโครงสร้างด้านล่าง และโปรแกรม (ภาพที่ 5.13)

```
public class A {
    public A() {
        ...... // constructor }
    public A(int i) {
        ...... // copy constructor }
```

2
ι
٢
^

1	class ProgContructor {			
2	//constructor method			
3	ProgContructor() {			
4	System.out.println("Constructor method called.");			
5	}			
6	<pre>public static void main(String[] args) {</pre>			
7	ProgContructor object = new ProgContructor(); //creating object			
8	}			
9	}			
•				
JavaRun				
Cons	Constructor method called.			

ภาพที่ 5.13 แสดงการสร้างอ็อบเจกต์คอนสตรัคเตอร์

คอนสตรัคเตอร์ โอเวอร์โหลด (constructor overloading)

เมธอดของจาวาก็เหมือนกับ เมธอดอื่นทั่วๆ ไป ที่สามารถทำการสร้างคอนสตัคเตอร์ได้ มากกว่าหนึ่งคอนสตัคเตอร์ ในคลาสเดียวกัน แต่มีการกำหนดแอทตริบิวต์ ที่แตกต่างกันได้ในคอน สตัคเตอร์ที่สร้างขึ้นมาใหม่ ดังนั้นการสร้างคอนสตรัคเตอร์หลายคอนสตรัคจึงถูกเรียกว่า คอนสตรัค เตอร์ โอเวอร์โหลด โปรแกรมตัวอย่างในภาพที่ 5.14



ภาพที่ 5.14 แสดงคอนสตรัคเตอร์ โอเวอร์โหลด

จากภาพที่ 5.14 อ็อบเจกต์ cpp ถูกสร้างเป็น default constructor คือจะถูกสร้างพร้อมกับ การสร้างสร้างคลาส ส่วนอ็อบเจกต์ java ถูกสร้างขึ้นด้วยการส่งพร้อมกับการใส่อาร์กูเมนต์ "Java" ส่วนเมธอด (ฟังก์ชัน) setName() ถูกเรียกใช้พร้อมกับกำหนดค่าแอทตริบิวส์ 'name' ของคลาส ConOverLoad และส่วนเมธอด (ฟังก์ชัน) getName() เป็นเมธอดที่ทำหน้าที่ในการแสดงผลเมธอด ออกสู่หน้าจอภาพ ซึ่งจากบรรทัดที่ 11 โค๊ด java.getName(); ทำงานก่อนบรรทัดที่ 12 โค๊ด cpp.getName(); มีผลทำให้มีการแสดงที่หน้าจอด้วยข้อความว่า "เขียนโปรแกรมด้วยภาษา: จาวา" ก่อน"เขียนโปรแกรมด้วยภาษา: C++" เพราะความว่า "จาวา" ถูกกำหนดค่าให้กับ getName() ก่อน คำว่า C++ นั้นเอง

สแตติคดาต้าเมมเบอร์ (Static data member)

การกำหนดคุณสมบัติข้อมูลของสมาชิกของอ็อบเจกต์ให้เป็นชนิดสแตติค หรือสแตติคดาต้าเมม เบอร์ เป็นการขอจองพื้นที่หน่วยความจำเดียวกันให้กับค่าของตัวแปรนั้น มีผลทำให้ค่าของตัวแปรจะ เป็นค่านั้นทุกครั้งที่เริ่มต้น และเมื่อมีการปรับปรุงค่าของตัวแปร จะมีผลทำให้ค่าของตัวแปรปรับตามไป ด้วยเสมอ ดังนั้นเมื่อมีการสร้าง instance ใหม่อีกกี่ครั้ง ค่าก็ปรับตามตลอดไป ตัวอย่างโปรแกรมในภาพ ที่ 5.15



ภาพที่ 5.15 แสดงการสร้างอ็อบเจกต์ด้วย static data member

จากภาพที่ 5.15 พบว่าตัวแปร x เป็นข้อมูลชนิด Integer แบบ static มีค่าข้อมูลเท่ากับ 111 เป็นสมาชิกของอ็อบเจกต์ StaticData จากนั้นทำการสร้างอ็อบเจกต์ใหม่เป็น t1 และ t2 จากอ็อบเจกต์ StaticData มีผลทำให้ค่าตัวแปร x เป็นสมาชิกของอ็อบเจกต์ t1 และ t2 ตามไปด้วย ด้วยค่าของ สมาชิกที่เป็นตัวแปรเป็นสแตติค ดังนั้นเมื่อมีการเพิ่มค่าที่ละหนึ่งของตัวแปรต้นแบบ (StaticData.x++) มีผลทำให้ค่าของตัวแปร x ของอ็อบเจกต์ t1 และ t2 ถูกเพิ่มค่าที่หนึ่งเป็นค่า 112 ด้วย

สแตติคเมธอดเมมเบอร์ (Static method member)

การกำหนดเมธอดเป็น static method จะมีผลทำให้สามารถเรียกใช้ได้ภายในคลาสเดียวกันได้ เช่นเดียวกันกับเมธอด main() ที่สามารถเรียกใช้นอกคลาสได้โดยผ่านชื่อคลาสได้ ซึ่งต่างจากเมธอด ทั่วไปที่ต้องสร้าง instance ขึ้นมาก่อน ดังนั้นคลาสที่มีสมาชิกเป็นสแตติค ไม่ว่าจะสมาชิกนั้นเป็นแอท ตริบิวต์หรือเป็นเมธอด เมื่อมีการปรับปรุงค่าก็จะมีผลถึงจุดอ้างอิงเดียวกัน นั่นคือทำให้ค่าของสมาชิก นั้นๆ เปลี่ยนค่าตามด้วย ตัวอย่างโปรแกรมภาพที่ 5.16



ภาพที่ 5.16 แสดงการใช้งานสแตติคเมธอด

จากภาพ 5.16 คลาส Ex5_16 เป็นคลาสที่มีสมาชิกที่เป็นเมธอด 3 เมธอดคือ เมธอด print(String str) เมธอด printChar(char ch) และเมธอด printStar(int stp) ซึ่งในเมธอดหลัก public static void main(String[] args) สามารถเรียกใช้เมธอดได้โดยตรงด้วยประโยค printStar(10) หรือ ประโยค print("ICE Department") นอกจากนี้ในส่วนของ main() ยังสามารถเรียกใช้เมธอดด้วยการ เรียกผ่านคลาสได้โดยตรง ด้วยประโยค Ex5_16.print("ข้อความที่ต้องการให้แสดงผลหน้าจอภาพ") ได้ผลการทำงาน (ภาพ 5.16)

ตัวอย่างโปรแกรม

ให้นักศึกษาฝึกทักษะในการทำความเข้าใจการเขียนโปรแกรมตามโครงสร้างของภาษาจาวาให้มี ความเข้าใจเป็นสำคัญ

ตัวอย่าง 5.1 เป็นตัวอย่างโปรแกรมการสร้างคลาสลักษณะคอนสตรัคเตอร์ และการส่งค่าเมื่อมีการ สร้างอ็อบเจกต์ ทำการบันทึกคลาสเป็น Lab5_1 ทำการทดสอบโปรแกรม เขียนผลการทำงานของ โปรแกรม

```
class Lab5_1
{
               static int x;
               int k;
               // constructor with 2 arguments
               public Lab5 1( int n, int m )
                       {
                               x = n;
                               k = m;
                               }
public static void main(String[] args)
       {
       Lab5 1 t1 = new Lab5 1(10, 20);
       Lab5 1 t2 = new Lab5 1(30, 40);
       System.out.print(t1.x + " ");
       System.out.print(t1.k + " ");
       System.out.print(t2.x + " ");
       System.out.println(t2.k);
                       }
}
```

ตัวอย่าง 5.2 โปรแกรมการใช้คลาส ให้นักศึกษาทำการวิเคราะห์โปรแกรม ว่าสามารถนำไปใช้งานได้ หรือไม่ ให้นักศึกษาทำการปรับปรุงโค๊ดให้ได้ผลตามที่นักศึกษาต้องการ

```
class Lab5_2
{
  static void Swap(Integer j, Integer k)
    {
      int tmp = k.intValue();
      k = new Integer(j.intValue());
      j = new Integer(tmp);
    }
  public static void main(String[] args)
    {
      Integer n = new Integer(5), m = new Integer(6);
      Swap(n, m);
      System.out.println("n = " + n + "; m = " + m);
      }
}
```

ด้วยแพ็กเกจของโปรแกรมภาษาจาวา เป็นส่วนสำคัญที่ผู้เริ่มต้นในการพัฒนาโปรแกรมควรรู้และ นำมาใช้งานดังตัวอย่างโปรแกรม 5.3

```
import javax.swing.JOptionPane; // import class JOptionPane
public class Lab5_3
{
    public static void main( String args[] )
    {
      JOptionPane.showMessageDialog(null, "Welcome to First inport packet
      \nJavaProgramming!" );
      System.exit( 0 ); // terminate application
      }// end method main
}// end class Program4
```

ให้นักศึกษาทำการเขียนโปรแกรมตามตัวอย่างโปรแกรมที่ 5.4 ให้สามารถทำงานรับจากผู้ใช้งาน จำนวน 2 ค่า แล้วนำมาคำนวณ พร้อมแสดงผลการคำนวณด้วย Dialog ของ Packet

```
import javax.swing.JOptionPane;
public class Lab5 4
public static void main( String args[ ] )
      {
       String FirstData, SecondData;
       int FirstNum, SecondNum, Sum;
       FirstData = JOptionPane.showInputDialog( "Enter first data" );
       SecondData = JOptionPane.showInputDialog( "Enter second data" );
       FirstNum = Integer.parseInt( FirstData );
       SecondNum = Integer.parseInt( SecondData );
       Sum =FirstNum +SecondNum:
       JOptionPane.showMessageDialog(null, "The summary = " + Sum +
          JOptionPane.INFORMATION MESSAGE ); // display the results
      System.exit( 0 );
              }
}
```

สรุป

การเขียนโปรแกรมภาษาจาวา เป็นการนำสิ่งแวดล้อมรอบตัวมาเป็นต้นแบบในการคิดกำหนดข้อมูลที่ เรียกว่า **อ็อบเจกต์** กำหนดหน้าที่ของสิ่งๆนั้นที่เรียกว่า **เมธอด** และเมื่อนำอ็อบเจกต์กับเมธอดมาเก็บ รวมกันกลายเป็น คลาส หนึ่งขึ้นมาเป็นคลาสต้นแบบ การสร้างคลาสจากคลาสเดิม(ต้นแบบ) หรือคลาส ที่เป็นมาตรฐาน(library class) คลาสมาตรฐานเป็นที่อำนวยความสะดวกแก่นักพัฒนาโปรแกรมอย่าง มากมาย และได้ทำการจัดเป็นหมวดหมู่แบ่งตามการใช้งานของโปรแกรมเป็นชุดๆ ของแต่คลาส ทำให้ สามารถเลือกใช้งานได้ไม่ซ้ำกัน จึงเรียกการรวมกันของคลาสว่า แพ็กเกจ (package) การสร้างวัตถุจาก คลาสอ็อบเจกต์ที่สร้างขึ้นจากคลาส จึงเรียกว่า สร้างอ็อบเจกต์พร้อมใช้งาน (instance of class) ซึ่งมี รูปแบบในการสร้าง 3 แบบ คือ 1) ประกาศ instance class ด้วย class_name Object_ identifier ; และทำการจองหน่วยความจำที่สร้างขึ้นด้วย Object_ identifier = new class_name ([class_parameter]); 2) ประกาศ instance พร้อมเตรียมพื้นที่บนหน่วยความจำสำหรับ instance ที่ สร้างใหม่ด้วย class name Object_ identifier = new class_name(); 3. สร้างอ็อบเจกต์โดยไม่ ประกาศตัวแปรอ็อบเจกต์เพื่อใช้อ้างอิงอ็อบเจกต์ใด้ new class name(); เป็นต้น

แบบฝึกหัด

- 1. จงอธิบายถึงความแตกต่างและความสัมพันธ์ระหว่างคลาสกับอ็อบเจกต์
- 2. จงอธิบายถึงความแตกต่างและความสัมพันธ์ระหว่างตัวแปรกับเมธอด
- 3. สมาชิกของคลาสสามารถประกอบด้วยอะไรบ้าง
- 4. จงยกตัวอย่างการตั้งชื่อคลาสที่เหมาะสมอย่างน้อย 5 คลาส
- 5. จงเขียนการกำหนดค่าให้กับตัวแปรของคลาสมาอย่างน้อย 5 ตัวอย่าง
- 6. จงบอกความจำเป็นในการสร้างอ็อบเจกต์พร้อมยกตัวอย่างในการอธิบาย
- 7. จงบอกประโยชน์ของการใช้แพ็กเกจ มาอย่างน้อย 5 ด้าน
- 8. จงบอกความแตกต่างการส่งผ่านข้อมูลด้วยค่าข้อมูลกับการอ้างอิงตำแหน่งของข้อมูล
- 9. จงบอกประโยชน์ของคอนสตรัคเตอร์พร้อมยกตัวอย่างอธิบาย
- 10. ให้นักศึกษาค้นคว้าว่า ชื่อสัญลักษณ์ภายในกล่อง Dialog box มีอะไรบ้าง

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพา: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพา: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก. - . (2545). Java Programming Volume II. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/

<u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

About the classes (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/java/javaOO/classes.html

About the java-Classes. (2012). Retrieved August 28, 2012, from:

http://pages.cs.wisc.edu/~hasti/cs368/JavaTutorial/NOTES/Java-Classes.html

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.
- Armstrong, E. (2004). The J2EE™ 1.4 Tutorial. Network Circle, CA: Sun Microsystem, Inc.

แผนบริหารการสอนประจำบทที่ 6

หัวข้อเนื้อหา

วงจรชีวิตแอพเพ็ต เมธอด init() เมธอด start() เมธอด stop() เมธอด destroy() เมธอด paint() การเขียนโค๊ดแอพเพ็ตแทรกในภาษาไฮเปอร์เท็กต์มาร์คอัพ (HTML) แอพเพ็ตกับกราฟิก แอพเพ็ตกับเสียง แอพเพ็ตกับภาพ แอพเพ็ตกับเทรดสำหรับภาพเคลื่อนไหว การสร้างเทรด การสร้างเทรดสำหรับภาพเคลื่อนไหว สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกทำงานวงจรชีวิตแอพเพ็ต
- 2. อธิบายประโยชน์การทำงานการโปรแกรมด้วยแอพเพ็ต
- 3. บอกการโปรแกรมตามวงจรชีวิตแอพเพ็ต
- 4. มีทักษะการโปรแกรมภาษาจาวาด้วยการเรียกใช้แอพเพ็ต
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนด

วิธีการสอนและกิจกรรมการเรียนการสอน

- 1. วิธีสอน
 - 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อแอพเพ็ตกับการใช้งานในรูปแบบต่างๆ
 - 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวา การน้ำแอพเพ็ตประยุกต์ไปใช้
 - งาน

1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์

- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 6
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือร้นในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั่งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 6 การใช้งานแอพเพ็ต (Applet Application)

การเขียนโปรแกรมด้วยภาษาจาวาที่ผ่านมามีการเรียกใช้แอพเพ็ตมาบางแล้ว ในบทนี้จะเป็น การทำความรู้จักแอพเพ็ตเริ่มจากโครงสร้างการทำงานอย่างเป็นระบบ และการใช้งานอย่างหลากหลาย ด้วยแอพเพ็ตเป็นที่นิยมใช้งานบนระบบอินเทอร์เน็ตในปัจจุบัน ด้วยตัวแอพเพ็ตเป็นโค๊ดขนาดเล็กที่ได้ แทรก(ฝัง)ตัวอยู่ในใน HTML ด้วยเป็นลักษณะสคริปต์จาวา ทำให้เมื่อทำการรันหรือเรียกใช้งานเว็บเพ็จ ที่เป็นลักษณะที่มีการโต้ตอบ (Active web) กันระหว่างผู้ใช้งานกับตัวแม่ข่าย เป็นไปอย่างราบรื่น ทำให้ สคริปต์จาวาที่เป็นโค๊ดแอพเพ็ตมีการใช้หน่วยความจำที่น้อย จึงเป็นที่นิยมใช้งานจนถึงปัจจุบัน

จากที่ได้กล่าวมาแล้วว่าการโปรแกรมด้วยแอพเพ็ต เป็นรูปแบบการทำงานของโปรแกรมภาษา จาวาอีกรูปแบบหนึ่ง ที่ทำงานโดยผ่านทางบราวเซอร์ ซึ่งมีขั้นตอนในการโปรแกรมคือ สร้างโค๊ดจาวา แล้วทำการคอมไพล์ด้วย javac ได้ผลเป็นคลาสหนึ่งตามชื่อแฟ้มข้อมูลที่กำหนด จากนั้นนำคลาสมา เขียนสคริปต์แทรกใน HTML ส่วนการเรียกใช้แอพเพ็ตต้องเรียกผ่านทางบราวเซอร์ หรือใช้ appletviewer ที่เป็นเทคโนโลยีของจาวา

วงจรชีวิตแอพเพ็ต (Applet Life Cycle)

ในการทำงานแอพเพ็ตจะมีวงจรชีวิต ด้วยการเรียกจาวาเวอร์ชวลแมทชีน (Java Virtual Machine) ทำในการสร้างจังหวะแรกของวงจรชีวิตและจบวงจรชีวิตด้วยระยะเวลาสั้นตัวโค๊ดของ Applet เป็นการสร้างเมธอดให้แอพเพ็ตทำงานได้ครบกระบวนการทำงาน ซึ่งประกอบด้วยเมธอดต่างๆ ดังนี้ init(), start(), pain(), stop(), และ destroy() (ภาพที่ 6.1)



ภาพที่ 6.1 แสดงกระบวนการทำงานวงจรชีวิตแอพเพ็ต ที่มา (Applet Life Cycle, 2014)

จากภาพที่ 6.1 เมื่อเริ่มต้น applet ถูกสร้างขึ้นโดยการ new() เหมือน instances ทั่วไป ก็จะ เริ่มทำงานทันที ด้วยเข้าสู่สถานะ init() แล้วผ่านเข้าสู่สถานะ start() จากนั้นเข้าสู่สถานะ paint() และ เข้าสู่สถานะรอ (wait) ซึ่งจะหยุดรอเหตุการณ์หรือสัญญาณจากผู้ใช้ที่จะเข้ามาให้เกิดการทำงานของ เมธอดที่กำหนดไว้สำหรับจัดการกับเหตุการณ์นั้นอยู่ หากผู้ใช้เปลี่ยนแปลงหน้าเว็บเพ็จ (Web page) จนทำให้พื้นที่ของแสดงผลของแอพเพ็ตนั้นต้องมีการถูกวาดใหม่ เช่นเมื่อ แอพเพ็ตถูกย้ายที่ (move) เปลี่ยนขนาด (resize) ขยายขนาด (maximize) แอพเพ็ตจะออกคำสั่งให้ทำการวาด (paint) อีกครั้ง โดยเรียก paint() ซึ่งจะส่งผลให้ แอพเพ็ตเข้าสู่สถานะ paint() และทำการวาดพื้นที่แสดงผลของ แอพเพ็ต อีกครั้ง และเมื่อผู้ใช้ทำการเลื่อนเว็บเพ็จไปดูหน้าอื่น หรือกดปุ่ม minimize จะทำให้ browser ออกคำสั่งให้ แอพเพ็ตผ่านเข้าสู่สถานะ stop() และแอพเพ็ตจะหยุดรออยู่จนกระทั่งหากผู้ใช้ กดปุ่ม close ของ browser เพื่อหยุดทำงานจะทำให้แอพเพ็ตเข้าสู่สถานะ stop() และไปสู่สถานะ destroy() แล้วสิ้นสุดการทำงาน ดังนั้นช่วงชีวิตของแอพเพ็ตสามารถสรุปเมธอดต่างๆ ได้ดังนี้

• เมธอด init() เมธอดนี้จะถูกเรียกเพียงครั้งเดียวเมื่อมีการโหลดโปรแกรมจาวาแอพเพ็ตหรือ เมื่อมีการรีโหลด (reload) โปรแกรมเว็บบราวเซอร์

 เมธอด start() เมธอดนี้จะถูกเรียกหลังจากทำคำสั่งในเมธอด init() และถูกเรียกใช้ทุกครั้งที่ ผู้ใช้กลับมายังเว็บเพ็จที่มีโปรแกรมจาวาแอพเพ็ตหน้านี้อยู่ หรือทุกครั้งที่มีการแสดงเฟรมที่มีโปรแกรม จาวาแอพเพ็ตหน้านี้อยู่

 เมธอด stop() เมธอดนี้จะถูกเรียกใช้เมื่อผู้ใช้ย้ายไปยังเว็บเพ็จหน้าอื่นหรือเมื่อมีการยกเลิก การแสดงเฟรมที่มีโปรแกรมจาวาแอพเพ็ตหน้านี้อยู่หรือหน้าเว็บเพ็จนั้นถูก minimized เมื่อใดที่ แอพเพ็ตถูก stop() จะทำให้ไม่มีการแสดงผลและไม่สามารถโต้ตอบกับผู้ใช้ได้

• เมธอด destroy() เมธอดนี้จะถูกเรียกครั้งเดียวเมื่อมีการปิดเว็บบราวเซอร์หรือสิ้นสุดการ ทำงาน เช่น กดปุ่มปิดอยู่ด้านบนขวาของ browser ก็จะทำการเรียก stop() และเรียก destroy() เมื่อ destroy() พื้นที่หน่วยความจำจะถูกเรียกกลับคืนไป

 เมธอด paint() โดยปกติ เมธอด paint() จะถูกเรียกหลังจากทำงานของ start() และถูกเรียก ให้ทำการการวาดเพื่อในแสดงผลของ แอพเพ็ตนั้นใหม่ ซึ่งอาจเกิดขึ้นได้หลายกรณี เช่น พื้นที่แสดงผล ของแอพเพ็ตนั้นถูกวินโดว์อื่นมาทับหรือในกรณีที่เว็บเพ็จนั้นถูก minimized (หรือ maximized) และ เมื่อถูกนำกลับมาอีก paint() ก็จะถูกเรียกเช่นกัน paint() จะมี Graphic g เป็นพารามิเตอร์ ซึ่งเป็น ข้อมูลเกี่ยวข้องกับสภาวะแวดล้อมทาง graphic ของแอพเพ็ตที่กำลังทำงานอยู่ แต่โปรแกรมของ แอพเพ็ต ไม่มีข้อมูล Graphic g นี้จึงไม่สามารถเรียก paint() ได้โดยตรง หากเราต้องการให้มีการวาด ใหม่เกิดขึ้นก็จะต้องเรียก repaint() ใหม่ทุกครั้ง คำสั่งที่เกี่ยวกับการวาดรูปกราฟิกมีรูปแบบดังนี้

```
public void paint (Graphics g)
{
[statements]
}
```

}

ในการสร้างโค๊ดจาวาแอพเพ็ตเตรียมพร้อมให้สามารถไปใช้งานได้นั้น เป็นการสร้างอ็อบเจกต์ที่ ได้ทำการขยายมาจากคลาสต้นแบบชื่อ java.applet.Applet ด้วยคีย์เวิร์ด extends หรือสามารถเรียก ได้ว่าในการสร้างโค๊ดจาวาแอพเพ็ตนั้นเป็นการสร้างคลาสย่อยจากคลาสของ java.applet.Applet นั่นเอง ดังตัวอย่างการสร้างอ็อบเจกต์ชื่อ FirstApplet สามารถทำได้ดังนี้

```
public class FirstApplet extends java.applet.Applet
```

```
ตัวโค็ดของ Applet
paint( ) เป็นเมธอดที่ทำหน้าที่สั่งให้ข้อมูล (ข้อความ รูปภาพ หรือสื่ออื่นๆ (เส้น กล่อง))
ของแอตเพ็ตแสดงสู่หน้าจอภาพด้วนรูปแบบการสั่งดังนี้
public void paint (Graphics g)
{
ตัวโค๊ดของสิ่งที่ต้องการให้แสดงสู่หน้าจอภาพ
}
```

หลักการทำงานของเมธอด paint() มีอาร์กูเมนต์ (Graphics g) หมายความว่าเมธอดpaint() ต้องการส่งพารามิเตอร์ g ที่มีชนิดของคลาส เป็น Graphics ออกสู่หน้าจอภาพ (ภาพที่ 6.2)



ภาพที่ 6.2 แสดงการสร้าง applet อยู่ในไฟล์ FirstApplet.java

เมื่อ FirstApplet.java ถูกคอมไพล์แล้วจะได้ FirstApplet.java.class จากนั้นจะต้องมี HTML page ที่มี APPLET tag ที่เรียก FirstApplet.java.class ให้ทำงานอยู่ใน ไฟล์ JavaApplet.html และทำการเรียก Applet มาใช้งานด้วย appletviewer (ภาพที่ 6.3)



ภาพที่ 6.3 แสดงการ APPLET tag อยู่ในไฟล์ JavaApplet.html

จากภาพที่ 6.3 พบว่าเมื่อแอพเพ็ตด้วยการเมธอด paint() ซึ่งเมธอด paint() มีส่งตัวแปร msg ที่เป็นชนิด String และทำการเก็บข้อความ "ขณะนี้แอพเพ็ตกำลังทำงานอยู่....." ทำการแสดง หน้าจอภาพ ซึ่งพบว่าเมื่อแอพเพ็ตทำงาน จะมีการแสดงข้อมูลเพิ่มเติม คือ init()start() ร่วมแสดงบน หน้าจอภาพด้วยเพราะขณะที่เมธอดpaint() ทำงานนั้น จากหลักการทำงานของวงจรชีวิตแอพเพ็ต ทำงานอยู่เมธอดของ init() กับเมธอด start() ยังคงทำงานอยู่รวมกับการทำงานของ เมธอดpaint() ตลอดเวลานั่นเอง

การเขียนโค๊ดแอพเพ็ตแทรกในภาษาไฮเปอร์เท็กต์มาร์คอัพ (HTML)

ในการให้ภาษาจาวาสามารถทำงานบนเว็บเพจได้สมบูรณ์นั้น จำเป็นอย่างยิ่งที่ต้องศึกษา การ เขียนโค๊ดในแท็กของแอพเพ็ตที่จำเป็นต่อการควบคุมการแสดงผลจอภาพได้อย่างหลากหลาย ดังเช่น การกำหนดไดเร็คทอรึให้กับไฟล์ของนามสกุล .class กับ html จำเป็นต้องทำการเก็บไฟล์ให้ อยู่ที่ไดเร็กทอรีที่เดียวกัน ซึ่งในการกำหนดชื่อไฟล์ .class ให้กับโค๊ดของแอพเพ็ตไม่สามารถกำหนด เส้นทางไดเร็กทอรีร่วมกับการกำหนดใน code ได้ ดังนั้นในไฟล์ .html ต้องทำการสั่งให้ทำการเรียกโค๊ด .class ที่ต่างไดเร็คทอรึได้ด้วยคำสั่ง CODEBASE และตามด้วย "./ชื่อไดเร็คทอรี" ดังตัวอย่าง เช่น ถ้าไฟล์อยู่ที่ไดเร็คทอรี classes ดังนั้น โค๊ดสามารถกำหนดได้ดังนี้

<APPLET CODE = "FirstApplet.class" CODEBASE= "./classes" width = 300 height =100>

</APPLET>

อีกความสามารถของการทำงานร่วมกันระหว่างแท็กแอพเพ็ต กับ HTML ก็คือ การส่ง พารามิเตอร์ที่อยู่ภายใต้แท็ก <APPLET> กับ </APPLET> โดยมีรูปแบบ

ซึ่ง ชื่อของพารามิเตอร์ เป็นลักษณะ Case-Sensitive หมายความว่า ชื่อพารามิเตอร์ที่กำหนด ขึ้นที่เป็นตัวอักษรภาษาอังกฤษที่เป็นตัวอักษรพิมพ์ใหญ่กับตัวอักษรพิมพ์เล็กเป็นคนละชื่อกัน ดังนั้นถ้า เป็นชื่อเดียวกันแต่คนละลักษณะตัวพิมพ์ ถือว่าเป็นชื่อพารามิเตอร์ คนละตัวกันจึงสามารถใช้ ในการเก็บ ค่าข้อมูล คนละตัวได้ (ภาพที่ 6.4 และ 6.5)

```
import java.applet.*;
 1
  import java.awt.*;
2
   public class ParamApplet extends Applet
3
   {
4
      public void paint(Graphics g)
 5
      {
6
         String Txt = getParameter("msg");
7
         g.drawString(Txt, 100, 100);
8
      }
9
10 }
```

ภาพที่ 6.4 แสดง ParamApplet.java รับพารามิเตอร์ msg จาก ParamApplet.HTML มาเก็บ Txt



ภาพที่ 6.5 แสดง ParamApplet.HTML การกำหนดพารามิเตอร์ msg ให้กับ ParamApplet.java

จากภาพที่ 6.4 และ 6.5 เป็นการทำงานร่วมกันระหว่างไฟล์ ParamApplet.HTML กับ paramApplet.java ในการส่งค่าพารามิเตอร์ด้วยการกำหนดค่า msg ให้มีค่าข้อมูลเท่ากับ Test senting parameter in HTML ด้วยแท๊ก *<param name = "msg" value = "Test senting parameter in HTML">* ของ ParamApplet.HTML มายัง ParamApplet.java ทำการรับค่าด้วย เมธอด getParameter จากพารามิเตอร์ msg มาเก็บไว้ใน Txt ด้วยโค๊ด *String Txt = getParameter("msg");*

แอพเพ็ตกับกราฟิก (Applet with Graphic)

การสร้างกราฟิกแสดงผลสู่หน้าจอภาพเป็นสิ่งที่จำเป็นอย่างมากในการนำเสนอข้อมูลให้มีความ น่าสนใจ ในโปรแกรมจาวาแอพเพ็ตได้สร้างคลาสอำนวยความสะดวกมาให้ผู้ใช้งานสามารถเรียกมา ใช้งานได้อย่างมากมาย โดยได้จัดเก็บเมธอดไว้ในแพ็คเก็ต java.awt (ต้องทำการ import java.awt เสมอ) เมธอดที่ใช้ในการจัดการคุณลักษณะของกราฟิก ได้แก่ setColor() และ setFont() ส่วนเมธอด ที่ทำการสร้างลักษณะรูปร่างต่างๆ ได้แก่เมธอด drawString(), drawLine(), drawRect(), fillRect(), drawRoundRect(), fillRoundRect(), drawOval(), fillOval(), drawArc(), fillArc(), drawPolygon(), fillPolygon() และเมธอด drawPolyline() ดังตัวอย่างในการนำมาใช้งานดังนี้

การกำหนดส์ให้ไปใช้กับอ็อบเจกต์ที่ต้องการแสดงกราฟิก มีรูปแบบการโค๊ด

setColor(Object สี);

ตัวอย่างในการใช้เมธอดการสร้างกราฟิกด้วยเมธอด setColor()และเมธอด drawString() (ภาพที่ 6.6 และ 6.7)

import java.applet.*; 1 2 import java.awt.*; public class ColorApplet extends Applet 3 4 { public void paint(Graphics g) { 5 g.setColor(Color.red); g.drawString ("Red", 50,10); 6 7 g.setColor(Color.green); g.drawString ("Green", 50,20); g.setColor(Color.blue); g.drawString ("Blue", 50,30); 8 } 9 $10 \}$

ภาพที่ 6.6 แสดงการกำหนด setColor() กับ drawString()



ภาพที่ 6.7 แสดงผลการเรียกใช้เมธอด setColor() กับ drawString()

จากภาพที่ 6.6 และ 6.7 พบว่ามีการกำหนดสีด้วย setColor(Color.สีที่ต้องการแสดง) การ กำหนดข้อมูลที่เป็นตัวอักษรด้วยเมธอด drawString("ข้อความที่ต้องการแสดง", ตำแหน่งข้องแกน x หรือตำแหน่งของตัวอักษรที่, ตำแหน่งข้องแกน y หรือตำแหน่งของบรรทัดที่)

ตัวอย่างในการใช้เมธอดการสร้างกราฟิกด้วยเมธอด setColor()และเมธอด drawLine() ซึ่งมีรูปแบบการโค๊ด คือ drawLine(x1, y1, x2, y2) โดย x1, y1 เป็นจุดเริ่มต้นของเส้นที่ต้องลาก ส่วน x2, y2 เป็นจุดสุดท้ายของเส้นที่ต้องลาก (ภาพที่ 6.8 และ 6.9)

```
import java.applet.*;
1
2
   import java.awt.*;
   public class drawLineApplet extends Applet
3
4
   {
5
     public void paint(Graphics g) {
        g.setColor(Color.red); g.drawLine (20,10,20,100);
6
       g.setColor(Color.green); g.drawLine (50,10,50,100);
7
8
     }
9 }
```

ภาพที่ 6.8 แสดงการกำหนด setColor() กับ drawLine()



ภาพที่ 6.9 แสดงผลการเรียกใช้เมธอด setColor() กับ drawString()

จากภาพที่ 6.8 และ 6.9 พบว่ามีการกำหนดสีด้วย setColor(Color.สีที่ต้องการแสดง) สามารถทำการกำหนดจุดการลากเส้นด้วยเมธอด drawLine()

แอพเพ็ตกับเสียง (Applet with Sound)

การกำหนดเสียงของแอพเพ็ตได้ทำการสร้างเมธอดสำหรับใช้งานได้อย่างสะดวก เมธอดแรก เป็นการนำเสียงมาเก็บไว้ที่ตัวแปรได้ด้วยเมธอด getAudioClip() และเมธอดสั่งการควบคุมเสียง เมธอด สั่งให้เปิดเสียง AudioClip.play() เมธอดสั่งวนรอบเสียง AudioClip.loop() และเมธอดปิดเสียง AudioClip.stop() เป็นต้น ซึ่งไฟล์เสียงที่นำมาใช้งานควรอยู่ที่โฟลเดอร์เดียวกับโค๊ดจาวา และชนิดของ ไฟล์เสียงที่สนับสนุนได้แก่ .wav .au และ .midi เป็นต้น (ภาพที่ 6.10 และ 6.11)

```
import java.awt.*;
 1
2 import java.applet.*;
    public class SoundApplet extends java.applet.Applet
3
    { AudioClip bkg;
4
      public void init() {
5
        bkg = getAudioClip(getCodeBase(), "complete.wav"); }
 6
      public void start() {
7
        bkg.loop();
8
                     - }
      public void paint(Graphics g) {
9
        g.setColor(Color.red);
                                g.drawString("Music time",50,100); }
10
      public void stop() {
11
12
        bkg.stop();
                         }
13
   }
```

ภาพที่ 6.10 แสดงการเรียกคลิปเสียงมาใช้งาน

จากโค็ดภาพที่ 6.10 พบว่าได้ทำการใช้เมธอด getAudioClip() รับเสียง complete.wav มา ทำการให้กับอ็อบเจกต์ bkg ดังนั้นในการนำเสียงมาใช้งานสามารถเรียกเมธอดผ่านทางอ็อบเจกต์ bkg ได้ด้วยเมธอด bkg.loop() ได้ทันที เช่นเดียวกับเมธอด bkg.stop() ซึ่งเมธอด bkg.loop() เป็นการสั่ง เปิดเสียงวนรอบตลอดเวลาที่วงจรชีวิตแอพเพ็ตทำงาน ดังนั้นถ้าต้องการให้เปิดเสียงเพียงรอบเดียว สามารถทำได้ด้วยการเขียนเมธอด public void start() {bkg.play(); } แทนได้



ภาพที่ 6.11 แสดงผลการทำงานของข้อความและเปิดเสียงจาก SoundApplet.class

แอพเพ็ตกับภาพ (Applet with Image)

การกำหนดภาพของแอพเพ็ตสามารถเรียกใช้งานได้อย่างสะดวก ได้ด้วยถ้าต้องการนำภาพมา เก็บไว้ที่ตัวแปรได้ด้วยเมธอด getImage() และเมธอดสั่งแสดงภาพได้ด้วยเมธอด g.drawImage("ไฟล์ ภาพ", 0, 0, null) ซึ่งไฟล์ภาพที่นำมาใช้งานควรอยู่ที่โฟลเดอร์เดียวกับโค๊ดจาวา และชนิดของไฟล์ภาพ ที่แอพเพ็ตสนับสนุนได้แก่ .gif และ .jpg เป็นต้น (ภาพที่ 6.12 และ 6.13)

```
import java.awt.*;
1
2
  import java.applet.*;
   public class ImageApplet extends java.applet.Applet
3
   { Image img;
4
5
     public void init() {
       img = getImage(getCodeBase(), "cool.gif");
6
                                                      }
     public void paint(Graphics g) {
7
        g.drawImage(img, 0, 0, null); }
8
9 }
```

ภาพที่ 6.12 แสดงการเรียกภาพเคลื่อนไหว cool.gif มาใช้งาน



ภาพที่ 6.13 แสดงภาพเคลื่อนไหวตามลักษณะของไฟล์ภาพ

จากภาพที่ 6.13 ในการแสดงภาพเคลื่อนไหวผู้เขียนโปรแกรมต้องพิจารณาเลือกภาพที่มีการ เคลื่อนไหวตามที่ต้องการมาแสดงได้ด้วย g.drawImage() ดังนั้นในส่วนของแอพเพ็ตมีหน้าที่เพียงนำ ภาพมาแสดงยังตำแหน่งที่ต้องการของวินโดว์เท่านั้น

แอพเพ็ตกับเทรดสำหรับภาพเคลื่อนไหว (Applet with Thread of Animation)

ภาพเคลื่อนไหวสามารถสร้างขึ้นได้ โดยการนำภาพนิ่งหลายๆ ภาพมาฉายต่อเนื่องกันด้วย ความเร็วสูง ทำให้เกิดภาพลวงตาของการเคลื่อนไหว โดยปกติความเร็วของภาพเคลื่อนไหวจะฉายด้วย ความเร็วที่ต่างกัน ขึ้นอยู่กับว่านำภาพเคลื่อนไหวมาใช้กับการแสดงผลชนิดใด เช่น ระบบภาพยนตร์จะ ใช้ 24 เฟรมต่อวินาที ในระบบ PAL ใช้ 25 เฟรมต่อวินาที และในระบบ NTSC ใช้ 29.97 หรือ 30 เฟรมต่อวินาที ส่วนในด้านภาพเคลื่อนไหวของคอมพิวเตอร์ เป็นภาพเคลื่อนไหวแบบ 2 มิติ คือเป็นภาพ ที่แสดงด้านความสูงและความกว้าง ซึ่งจะมีความเหมือนจริงพอสมควรและการสร้างจะไม่สลับซับซ้อน มากนัก เช่น ภาพเคลื่อนไหวที่ปรากฏตามเว็บ และภาพที่เป็นตระกูล .gif สามารถสร้างภาพเคลื่อนไหว โดยใช้ความเร็วเริ่มต้นตั้งแต่ 10 เฟรมต่อวินาที จนถึง 30 เฟรมต่อวินาที ก็สามารถทำได้ขึ้นอยู่ความ ต้องการของผู้สร้างต้องการให้ภาพเคลื่อนไหวโดยมีความรื่นไหลของภาพ (ภาพไม่สะดุด) อย่างต่อเนื่อง เพียงใด

เทรด (Thread) คือ ส่วนประกอบย่อยของการโปรเซส (ประมวลผล) นั่นคือเทรด จึงเป็นงาน ย่อยๆ ของการประมวลผลในขบวนการทำงานของโปรเซสเซอร์ (CPU) ดังนั้นถ้าในการทำงานของซีพียู 1 โปรเซส (ครั้ง) ที่มีการทำงานหลายๆ เทรด (งานย่อย) จึงถูกเรียกว่า Lightweight process (LWP) แต่ถ้าโปรเซสเซอร์มีการทำงาน 1 เทรด ก็แสดงว่ามีการทำงานได้เพียง 1 งาน จะถูกเรียกว่า Heavyweight process (HWP) โดยปกติโปรเซสที่มี 1 เทรด จะเรียกว่า Single thread แต่ถ้า 1 โปรเซสมีหลายเทรด จะเรียกว่า Multithread ด้วยโปรเซสหนึ่งอาจมีได้หลายเทรด เช่น การทำงานของ เว็บบราวเซอร์ 1 หน้า อาจมีทั้งเทรดการดาวน์โหลดข้อมูล เทรดการแสดงข้อความแจ้งสถานะภาพ ดาวน์โหลด พร้อมทั้งต้องมีเทรดที่ทำหน้าที่แสดงรูปภาพการทำงานของการดาวน์โหลด

การสร้างเทรด (Created Thread)

ในแพ็กเกจของแอพเพ็ต เป็นระบบการทำงานในรูปแบบของเทรดเป็นพื้นฐานอยู่ กล่าวคือเมื่อ ทำการเรียกเมธอด run() ซึ่งระบบก็ต้องมีส่วนประกอบการทำงานเทรดของเมธอด paint() ประกอบทุก ครั้งอย่างน้อย 1 เทรด ตัวอย่าง การสร้างคลาสที่มีการอินเตอร์เฟส (interface) เทรดเป็นส่วนย่อย นั่นเอง เช่น public class ClassName implements Runnable {...} หรือสร้างคลาสด้วย public class ClassName extends Thread {...} ดังตัวอย่างโค๊ด และการแสดงผลทำงาน (ภาพที่ 6.15)

```
import java.applet.*;
import java.awt.*;
public class AnimationApplet extends Applet implements Runnable
{
    Thread anima;
```

```
int delay;
       int frame;
public void init()
       {
       int frameNo =Integer.parseInt(getParameter("frameNo"));
       delay = 100 / frameNo;
       }
public void start()
       {
       anima = new Thread(this);
       anima.start();
       }
public void stop()
       {
       anima = null;
       }
public void run()
{
       while (Thread.currentThread() == anima )
               {
               repaint();
                       try
                       {
                              Thread.sleep(delay);
                       }
                       catch (InterruptedException e)
                       {
                               break;
                       }
               frame++;
               }
}
public void paint(Graphics g)
       {
        g.drawString("Animation Display", frame +20, 20);
        g.drawString("Frame = " + frame ,10,10);
       }
```

```
}
```



ดังภาพที่ 6.14 แสดงผลการทำงาน Animation ของข้อความ

การสร้างเทรดสำหรับภาพเคลื่อนไหว

ในการสร้างภาพเคลื่อนไหว ก่อนทำการสร้างต้องเตรียมภาพที่ต้องการนำมาแสดง และพื้นหลัง ที่ต้องการให้ภาพนิ่ง จำนวนตามที่ต้องการสร้าง ในตัวอย่างนี้ต้องการสร้างภาพเคลื่อนไหว 2 ภาพ (bravo.gif) (cool.gif) และ 1 พื้นหลัง (lighthouse.gif) โดยเขียนโค๊ด และตรวจสอบผลการทำงาน (ภาพที่ 6.15)

import java.applet.*;
import java.awt.*;
public class AnimationMixApplet extends Applet implements Runnable
{
 Thread anima;
 int delay;

int frame; Image bravo; Image cool; Image Lighthouse;

```
Dimension stopDimension;
       Image stopImage;
       Graphics stopGraphic;
public void init()
{
       int frameNo =Integer.parseInt(getParameter("frameNo"));
       delay = 1000 / frameNo;
       bravo = getImage(getCodeBase(), "bravo.gif");
       cool = getImage(getCodeBase(), "cool.gif");
       Lighthouse = getImage(getCodeBase(), "Lighthouse.gif");
       }
public void start()
{
       anima = new Thread(this);
       anima.start();
       }
public void stop()
{
       anima = null;
       }
public void run()
{
       long temp = System.currentTimeMillis();
       while (Thread.currentThread() == anima )
              {
               repaint();
               try
                      {
                              temp += delay;
                              Thread.sleep(delay);
              }
               catch (InterruptedException e)
                      {
                              break;
              }
               frame = (frame +1) % 100;
              }
       }
```

141

```
public void paint(Graphics g)
{
        update(g);
       }
public void update(Graphics g)
{
       Dimension d = new Dimension(200,200);
       if ( (stopGraphic == null) || (d.width != stopDimension.width) || (d.height
           != stopDimension.height))
       {
               stopDimension = d;
               stopImage = createImage(d.width, d.height);
               stopGraphic = stopImage.getGraphics();
       }
       // Remove the priviouse image
       stopGraphic.setColor(getBackground());
       stopGraphic.fillRect(0,0,d.width, d.height);
       stopGraphic.setColor(Color.black);
       //Paint the frame in the image
       paintFrame(stopGraphic);
       //Paint the image onto the screen
       g.drawImage(stopImage,0,0,null);
       }
public void paintFrame(Graphics g)
{
       dimension d = new Dimension (200, 200);
       int w = Lighthouse.getWidth(this);
       int h = Lighthouse.getHeight(this);
       if ( (w>0) && (h>0) )
       {
               g.drawlmage(Lighthouse, (d.width - w) /2, (d.height -h )/2, this);
               }
               w = bravo.getWidth(this);
               h = cool.getHeight(this);
```

1	<html><head><title> This is Animation Applet </title></head></html>	
2	<body><center></center></body>	
3	<applet code="AnimationMixApplet.class</th><th>s" height="500" width="500"></applet>	
4	<pre>>param name = "frameNo" value = 10></pre>	🛃 Applet Viewer: AnimationMixApplet.class
5		Applet
6		56
		Applet started.

ภาพที่ 6.15 แสดงผลการทำงานภาพเคลื่อนไหว

จากโค็ดเมธอด paint() ถูกแทนด้วย update() ทำหน้าที่ในการแสดงภาพเคลื่อนไหว โดย สามารถกำหนดตำแหน่งการเคลื่อนไหวของแต่ละภาพได้ ภาพไฟล์ bravo.gif กำหนดสูตรด้วย g.drawImage(bravo, d.width - ((frame * 5) % w), (d.height - h)/2, this); และภาพไฟล์ cool.gif กำหนดด้วย g.drawImage(cool, d.width - ((frame * 7) % w), d.height/2 - h*2, this); ส่วนการ สัมพันธ์ระหว่างภาพด้วยโค็ด g.drawString((((d.height /2) - Math.abs((int)(h * Math.sin(frame / 4)))) % 200) + "", 20, 30); ดังนั้นผู้เขียนโปรแกรมสามารถปรับปรุงสูตรได้ตามต้องการ

ตัวอย่างโปรแกรม

ตัวอย่างที่ในการใช้เมธอดการสร้างกราฟิกด้วยเมธอด setColor()และเมธอด drawRect() ซึ่ง มีรูปแบบการโค๊ด คือ drawRect(x, y, w, h) โดย x, y เป็นจุดเริ่มต้นของรูปสี่เหลี่ยม ส่วน w เป็น ความกว้างของรูปสี่เหลี่ยม และ h เป็นความสูงของรูปสี่เหลี่ยม และการเติมสีลงในรูปเหลี่ยม

```
ตัวอย่าง 6.1 โปรแกรมในการกำหนดสี สร้างรูปสี่เหลี่ยม และ เติมส์ในรูปเหลี่ยม
import java.applet.*;
import java.awt.*;
public class drawRectApplet extends Applet
{
       public void paint(Graphics g)
       {
              g.setColor(Color.red); g.drawRect (20,10,20,100);
              g.setColor(Color.green); g.fillRect (50,10,50,100);
       }
}
และโค๊ด drawRectApplet.HTML
<html><HEAD><TITLE> This is applied Applet </TITLE></HEAD>
<BODY>
<APPLET code = "drawRectApplet.class" width = 300 height =100>
</APPLET>
</BODY>
</HTML>
```

```
ผลการทดลอง
```



ภาพที่ 6.16 แสดงผลการทำงานภาพรูปสี่เหลี่ยม และ เติมสึในรูปเหลี่ยม
ให้นักศึกษาว่าจะเขียนโปรแกรม ให้ได้ได้ดังภาพ 6.17 ได้อย่างไร ซึ่งมีการใช้เมธอดการสร้าง กราฟิกด้วยเมธอด setColor()และเมธอด drawRoundRect(20,10,100,100,50,50) ซึ่งมีรูปแบบการ โค๊ด คือ drawRoundRect(x, y, w1, h1, w2, h2) โดย x, y เป็นจุดเริ่มต้นของรูปสี่เหลี่ยม ส่วน w1 เป็นความกว้างของรูปสี่เหลี่ยม และ h1 เป็นความสูงของรูปสี่เหลี่ยม ส่วน w2 เป็นความกว้างของรูป สี่เหลี่ยมของที่ใช้กำหนดความโค้งของมุมสี่เหลี่ยม และ h2 เป็นความสูงของรูปสี่เหลี่ยมของที่ใช้กำหนด ความโค้งของมุมสี่เหลี่ยม (ภาพที่ 6.14) และการเติมสีลงในรูปเหลี่ยมเมธอด fillRoundRect()



ภาพที่ 6.17 แสดงพิกัดของเมธอด drawRoundRect(x, y, w1, h1, w2, h2)

ตัวอย่าง 6.2 โปรแกรมในการแสดงค่าของเมธอด drawRoundRect() กับเมธอด fillRoundRect()

```
import java.applet.*;
import java.awt.*;
public class drawRoundRectApplet extends Applet
{
    public void paint(Graphics g)
    {
        g.setColor(Color.black); g.drawRoundRect(20,10,100,100,50,50);
        g.setColor(Color.blue); g.fillRoundRect(50,50,100,100,50,50);
    }
}
```

และโค๊ด drawRoundRectApplet.HTML

```
<HTML><HEAD><TITLE> This is applied Applet </TITLE></HEAD>
<BODY>
```

<APPLET code = "drawRoundRectApplet.class" width = 300 height =100> </APPLET>

</BODY> </HTML>

ผลการทดลอง



ภาพที่ 6.18 แสดงการใช้เมธอด drawRoundRect() กับเมธอด fillRoundRect()

ตัวอย่าง 6.3 โปรแกรมในการแสดงค่าของเมธอด drawOval() กับเมธอด fillOval()

```
import java.applet.*;
import java.awt.*;
public class drawOvalApplet extends Applet
{
    public void paint(Graphics g)
    {
        g.setColor(Color.pink); g.drawOval(20,10,100,100);
        g.setColor(Color.orange); g.fillOval(50,50,100,100);
    }
}
```

และโค๊ด drawOvalApplet.html

<HTML><HEAD><TITLE> This is applied Applet </TITLE></HEAD> <BODY>

```
<APPLET code = "drawOvalApplet.class" width = 300 height =100>
</APPLET>
```

</BODY> </HTML> ให้นักศึกษาทำการศึกษาค้นคว้าเพิ่มเติมในการเขียนโค๊ด โดยใช้ g .setColor(), drawRect(), fillRect(), fillOval() ให้สามารถแสดงดังผลการทดลอง (ภาพที่ 6.19)



ภาพที่ 6.19 แสดงการใช้เมธอด ใช้ g .setColor(), drawRect(), fillRect(), fillOval()

สรุป

การเขียนโปรแกรมด้วยภาษาจาวาให้สามารถใช้งานบนระบบอินเทอร์เน็ตได้ ต้องอาศัย เทคโนโลยีแอพเพ็ต เป็นเครื่องมือช่วยในการพัฒนาโค๊ดที่มีขนาดเล็กที่ได้ฝังตัวอยู่ในหน้าของเว็บเพ็จ ใน ลักษณะสคริปต์จาวา ทำให้เมื่อมีการรันหรือเรียกใช้งานเว็บเพ็จ จะทำให้เว็บเพ็จนั้นๆ สามารถโต้ตอบ กันระหว่างผู้ใช้งานกับระบบอินเทอร์เน็ตได้ การสร้างกราฟิก แสดงผลสู่หน้าจอภาพเป็นสิ่งที่จำเป็น อย่างมากในการนำเสนอข้อมูลให้มีความน่าสนใจ ในแอพเพ็ตได้สร้างคลาสอำนวยความสะดวกเก็บไว้ที่ แพ็คเก็ต java.awt ที่ใช้ในการจัดการคุณลักษณะของกราฟิก เช่น setColor() และ setFont() drawString(), drawLine(), drawRect(), fillRect(), drawRoundRect(), fillRoundRect(), drawOval(), fillOval(), drawArc(), fillArc(), drawPolygon(), fillPolygon() และเมธอด drawPolyline() รวมทั้งเมธอดที่อำนวยความสะดวกด้านเสียง เช่น getAudioClip() AudioClip.play() AudioClip.loop() AudioClip.stop() และด้านการจัดการด้านภาพเคลื่อนไหว เป็นต้น

แบบฝึกหัด

```
จงเขียนความแตกต่างระหว่าง stop() กับ destroy() ของวงจรชีวิตแอพเพ็ต
1.
   ้จงเขียนขบวนการพัฒนาแอพเพ็ตว่าต้องใช้องค์ประกอบใดบ้างในการสร้างผลการทดลอง
2.
   ้จงเขียนรูปแบบการโค๊ด drawArc(), พร้อมอธิบายความหมายของแต่ละอาร์กูเมนต์
3.
   จงเขียนรูปแบบการโค๊ด fillArc() พร้อมอธิบายความหมายของแต่ละอาร์กูเมนต์
4.
   ้จงเขียนรูปแบบการโค๊ด drawPolygon() พร้อมอธิบายความหมายของแต่ละอาร์กูเมนต์
5.
6. จงเขียนรูปแบบการโค๊ด fillPolygon() พร้อมอธิบายความหมายของแต่ละอาร์กูเมนต์
7. จงเขียนรูปแบบการโค้ด drawPolyline() พร้อมอธิบายความหมายของแต่ละอาร์กูเมนต์
8. จากโค๊ดตัวอย่างด้านล่างมีจุดที่เขียนไม่ถูกต้องให้ทำการปรับปรุงให้ถูกต้อง
   import java.applet.*;
   import java.awt.*;
   public class ParamExApplet extends Applet
   {
       public void paint(Graphics g)
       ł
               Sting Txt = getParaMeter("msg");
               g.drawString(Txt, 20, 100);
       }
   }
9. จากโค๊ดด้านล่างจงเขียนแสดงผลการทำงาน
       import java.applet.*;
       import java.awt.*;
       public class drawLineExApplet extends Applet
       {
               public void paint(Graphics g) {
                       g.setColor(Color.red);
                                                      g.drawLine (40,50,100,50);
                        g.setColor(Color.blue);
                                                     g.drawLine (40,60,100,60);
               }
       }
   เขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด
```

จงกำหนดเสียงให้กับแอพเพ็ต โดยนำเสียงมาเก็บไว้ที่ตัวแปรและสั่งให้เปิดเสียง AudioClip.play()
 โดยวนรอบจำนวน 5 รอบ

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉ**บับพื้นฐาน.** กรุงเทพา: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

้ วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

- . (2545). Java Programming Volume II. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.
- สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/ %BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf

สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555 จาก <u>www.thaiglossary.org</u>

Applet Life Cycle. (2014). Retrieved June 2, 2014, from:

http://www.ustudy.in/node/1266

About the classes. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/java/javaOO/classes.html

- About the java-Classes. (2012). Retrieved August 28, 2012, from: <u>http://pages.cs.wisc.edu/~hasti/cs368/JavaTutorial/NOTES/Java-Classes.html</u>
- About the java technology. (2012). Retrieved August 28, 2012, from: <u>http://docs.oracle.com/iavase/tutorial/getStarted/intro/definition.html</u>
- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.
- Armstrong, E. (2004). The J2EE™ 1.4 Tutorial. Network Circle, CA: Sun Microsystem, Inc.

แผนบริหารการสอนประจำบทที่ 7

หัวข้อเนื้อหา

ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) ด้วย AWT ความสัมพันธ์ระหว่าง AWT และ Swing การโปรแกรม GUI ด้วย Swing และ AWT การจัดวางองค์ประกอบ FlowLayout BorderLayout GridLayout สร้าง Basic Event Handling สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกความสำคัญการสร้างส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI)
- 2. บอกความแตกต่างระหว่าง AWT และ Swing
- 3. บอกการโปรแกรม GUI ด้วย Swing และ AWT
- 4. มีทักษะการโปรแกรมภาษาจาวาการจัดองค์ประกอบ GUI
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนดได้

วิธีการสอนและกิจกรรมการเรียนการสอน

- 1. วิธีสอน
 - 1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อ GUI ด้วย Swing และ AWT
 - 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวาการนำGUI ประยุกต์ไปใช้งาน
 - 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง

คอมพิวเตอร์

- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 7
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การชักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สังเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือร้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 7 ส่วนติดต่อกับผู้ใช้แบบกราฟิก (Graphical User Interface: GUI)

การเขียนโปรแกรมภาษาจาวาให้มีการตอบสนองความต้องการของผู้ใช้งาน ทั้งทางด้านการรับ ข้อมูลจากผู้งาน ด้านความน่าสนใจ ความสวยงามของหน้าเว็บ รวมทั้งง่ายต่อการใช้งานนั้น โปรแกรม ภาษาจาวาได้สร้างเครื่องมืออำนวยความสะดวกให้กับผู้พัฒนาโปรแกรมไว้อย่างสมบูรณ์แบบมากๆ ด้วย ส่วนติดต่อกับผู้ใช้งานแบบกราฟิก ด้วยแพ็กเกจของกลุ่ม AWT (Abstract Windows Toolkit) และ กลุ่ม Swing เพื่อให้ความสามารถในการสื่อสารระหว่างระบบของโปรแกรมกับผู้ใช้งานได้ทำงานอย่าง ถูกต้อง ดังนั้นในการสร้าง GUI โปรแกรมต้อง import java.awt.* กับ java.awt.event.* และ javax.swing.* เสมอ

ส่วนติดต่อกับผู้ใช้แบบกราฟิก ของโปรแกรมภาษาจาวาต้องมีใช้ทั้ง AWT และ Swing โดย Swing จะมีความแตกต่างระหว่าง AWT คือ Swing จะมีตัว J นำหน้าออบเจ็กต์ จาก Frame ดังเดิม ของ AWT จะกลายเป็น JFrame เป็นออบเจ็กต์ของ Swing นั่นเอง

ใน Swing จะมีออบเจ็กต์ frame ที่เรียกว่า JFrame ถูกเก็บในส่วน container บนสุด (Top-Level Containers) ที่สามารถแสดงผลออกสู่หน้าจอได้ และส่วนที่เป็น panel (secondary container) ที่ไม่สามารถแสดงผลได้ แต่จะมี JPanel, JScrollPane, JLayeredPane, JSplitPane และ JTabbedPane ทำหน้าที่ในการควบคุมการทำงานของเฟรม และส่วน simply เป็นส่วนด้าน ข้อความ เช่น text fields ต่างๆ

ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) ด้วย AWT

AWT มีส่วนประกอบ 12 แพ็คเกจ ซึ่งหนึ่งในนั้นjava.awt และส่วนย่อย java.awt.event เป็น ส่วนประกอบของ AWT ที่สามารถนำมาใช้งานได้ ดังนี้

- 1. แพ็คเกจ java.awt บรรจุด้วยคลาสกราฟิก core AWT ดังนี้
 - o GUI Component classes (เช่น Button, TextField, and Label),
 - o GUI Container classes (เช่น Frame, Panel, Dialog and ScrollPane),
 - o Layout managers (เช่น FlowLayout, BorderLayout and GridLayout),
 - o Custom graphics classes (เช่น Graphics, Color and Font).
- 2. แพ็คเกจ java.awt.event มีส่วนที่รองรับการทำงานดังนี้
 - Event classes
 (เช่น ActionEvent, MouseEvent, KeyEvent และWindowEvent)
 - o Event Listener Interfaces
 (เช่น ActionListener, MouseListener, KeyListener และWindowListener)
 - o Event Listener Adapter classes (เช่น MouseAdapter, KeyAdapter, and WindowAdapter)

ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) ด้วยจาวา สามารถทำการแบ่งโครงสร้างหน้าเว็บที่แสดง ข้อมูลและรับข้อมูลนั้นมีส่วนประกอบหลัก 2 ส่วน (elements) (ภาพที่ 7.1)



ภาพที่ 7.1 แสดงส่วนประกอบหลักของ GUI ที่มา (J4a_GUI, 2014)

จากภาพที่ 7.1 แสดงส่วนประกอบหลักของ GUI ซึ่งส่วนประกอบ มีดังนี้

Containers เป็นส่วนที่ทำหน้าที่ในเก็บ Components ต่างๆ ที่ต้องแสดงหน้าเว็บ ตามที่ กำหนด ซึ่งอาจเป็นโครงร่างในการออกแบบการวาง layout (เช่น flow หรือ grid) และส่วนของ container สามารถเก็บส่วนที่เป็น sub-containers ได้

Components เป็นส่วนประกอบเป็นส่วนที่แสดงข้อมูล บนหน้าจอเว็บ เช่น Button, Label, and TextField เป็นต้น

Components ของ GUI สามารถทำการควบคุม Microsoft ActiveX Control และ Eclipse's Standard Widget Toolkit รวมทั้ง Google Web Toolkit ซึ่งเป็นส่วนที่ยอมผู้เขียน โปรแกรมทำการแลกเปลี่ยน (interact) และการให้ทำการควบคุม (control) ส่วนของการนำไปใช้งาน (application) ทะลุเข้ามายังส่วนของ components ต่างๆ ได้ เช่น button-click และ text-entry

ในการโปรแกรม GUI ทุก component ต้องถูกวางใน container โดยทุก component สามารถวางด้วยเมธอด add (Component c) มีผลทำให้ Container สามารถเข้ามาทำงานได้ แสดงตัวอย่างดังนี้

Panel panel = new Panel();
Button btn = new Button("Press");
panel.add(btn);

// Panel is a Container

// Button is a Component

// The Panel Container adds a Button
Component

คลาส Container ใน AWT

การโปรแกรม GUI ใน Container จะต้องมี frame, Dialog และ Applet (Top-Level Containers) ซึ่งในส่วนของ frame ได้จัดเตรียมส่วนที่เป็น main window สำหรับเขียน โปรแกรมประยุกต์ใช้งาน มีส่วนส่วนประกอบของวินโดว์ ชื่อบาร์ (title bar) ซึ่งประกอบด้วย ไอคอน (icon) ชื่อบาร์(title name) ปุ่มย่อ (minimize) ปุ่มขยาย (maximize) และปุ่มปิด (close) นอกจาก ส่วนของ ชื่อบาร์ frame ยังได้จัดเตรียมส่วนที่เป็น ส่วนเพิ่มเติมเมนูบาร์ (optional menu bar) และ ส่วนที่เป็นพื้นที่ในการแสดงข้อมูล (display area) (ภาพที่ 7.2)



ภาพที่ 7.2 แสดง Top-Level Containers ที่มา (J4a_GUI, 2014)

AWT dialog เป็น pop-up window ที่ใช้ในการปฏิสัมพันธ์ (interacting) กับผู้ใช้งาน ซึ่ง dialog จะมีชื่อของ dialog จะประกอบด้วย icon title ปุ่มปิด (close button) และส่วนแสดง ข้อความ (ภาพที่ 7.3)

🛓 A Dialog Box	×
Enter a number	

ภาพที่ 7.3 แสดงกล่อง dialog ที่มา (J4a_GUI, 2014)

Secondary Container (Panel และ ScrollPane)

Secondary containers เป็นส่วนของพื้นที่ภายในส่วน top-level container ซึ่งประกอบด้วย

- Panel เป็นพื้นที่กล่องสี่เหลี่ยมอยู่ในระดับ higher-level container ใช้เป็น ตัวกำหนด layout ของรูปแบบขององค์ประกอบ GUI อาจอยู่ในรูปของ grid หรือ flow ก็ได้
- ScrollPane เป็นส่วนที่ใช้ในการควบคุมการเลื่อนซ้าย/ขวาของหน้าเว็บ (vertical scrolling) การเลื่อนขึ้น/ลง (horizontal scrolling)
- o อื่นๆ

โครงสร้างของ AWT Container Classes

ระบบในส่วนติดต่อกับผู้ใช้แบบกราฟิก มีโครงสร้าง (ภาพที่ 7.4)



ภาพที่ 7.4 แสดงระบบโครงสร้างของ GUI ด้วย AWT ที่มา (J4a_GUI, 2014)

คลาสที่เป็น Component ของ AWT

AWT ได้จัดเตรียม component พร้อมใช้งานและการนำมาใช้ใหม่ ประกอบด้วย Button, TextField, Label, Checkbox, CheckboxGroup (radio buttons), List, และ Choice (ภาพที่ 7.5)



ภาพที่ 7.5 แสดง Component ของ AWT ที่มา (J4a_GUI, 2014)

ความสัมพันธ์ระหว่าง AWT และ Swing

Abstract Window Toolkit (AWT) ได้ถูกพัฒนามาตั้งแต่สร้างภาษาจาวามา ได้เตรียมให้ platform มีความอิสระจาก APIs ด้วยการให้สามารถทำการ interface components ได้ ซึ่ง AWT แต่ละcomponent จะถูกปรับเปลี่ยน (render) อ็อบเจกต์ที่เขียนเป็นส่วนหนึ่งของระบบปฏิบัติการ window ซึ่งถ้าพิจารณา Swing components มักพบว่ามีความยึดหยุ่น (lightweight) กว่าด้วยเพราะ มันไม่ยึดติดกับระบบตำแหน่งของระบบปฏิบัติการ window ซึ่ง Swing จะเป็นส่วนขยายจาก AWT หรือมาทำงานส่วนเพิ่มเติมจาก AWT มากกว่า แต่ยังมีส่วนที่ต้องทำงานร่วมกันอย่างแยกไม่ออก ดัง แสดงไดอะแกรมความสัมพันธ์ (ภาพที่ 7.6)



ภาพที่ 7.6 แสดงความสัมพันธ์ระหว่าง AWT กับ Swing ที่มา (J4a_GUI, 2014)

การโปรแกรม GUI ด้วย Swing และ AWT

ขั้นตอนที่ 1 สร้าง JFrame ด้วยรูปแบบการเรียกคลาส JFrame ดังนี้

JFrame fName = new JFrame("Title");

โดย

JFrame	คือ	ตัวโค๊ดคำสั่งการสร้างเฟรม
fName	คือ	ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
Title	คือ	ชื่อของเฟรม(วินโดว์)ที่สร้างขึ้น

การสร้าง JFrame สามารถทำการเขียนด้วยการสร้าง frame มาหนึ่งเฟรมด้วยเมธอดที่ได้รับ การสืบทอดมาจาก JFrame ซึ่งได้มาจากการ import javax.swing.* และสั่งปิดวินโดว์ได้ได้ด้วยโค๊ด f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); ดังแสดงตัวโปรแกรมและ ผลการทำงาน ของโปรแกรม (ภาพที่ 7.7)



ภาพที่ 7.7 แสดงขั้นตอนแรกในการสร้าง JFrame

จากภาพที่ 7.7 มีการสร้างอ็อบเจกต์ f จากคลาสต้นแบบ JFrame และมีการกำหนดชื่อ title ว่า My Frame ซึ่ง import มาจาก javax.swing.* ด้วยโค๊ด JFrame f = new JFrame("My Frame");

ขั้นตอนที่ 2 สร้าง JPanel ซึ่งเป็น component เมื่อทำการสร้างแล้ว ต้องนำ JPanel ไปวางบนในส่วน ของ container ในที่นี้ก็คือ JFrame ซึ่งมีรูปแบบของการสร้าง JPanel ดังนี้

JPanel pName = new JPanel(manager);

โดย

JPanel	คือ ตัวโค๊ดคำสั่งการสร้าง panel
pName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
manager	คือ ชื่อของการจัดการการจัดลำดับการทำงาน component

การสร้าง JPanel นั้นเป็นการสร้างเลย์เอาต์ของการจัด component แต่ละอ็อบเจกต์ว่า ต้องการให้มีส่วนประกอบใดบ้าง และส่วนประกอบใดทำงานก่อนหลังตามชนิดของ manager ที่กำหนด ลงในรูปแบบคำสั่ง ถ้าไม่กำหนดค่าของ manager จะถูกกำหนดให้เป็น flow layout นั้นหมายความว่า ในการจัดวาง layout จะทำการเรียงไปตามลำดับที่สั่งมา ดังตัวอย่างโปรแกรม (ภาพที่ 7.8)



ภาพที่ 7.8 แสดงขั้นตอนการสร้าง JPanel

ขั้นตอนที่ 3 สร้างคลาสที่เป็น Component 3.1 ปุ่ม (Button) เป็นส่วนที่รองรับการใช้งานเมาส์มีรูปแบบการสร้าง ดังนี้

JButton *bName* = new **JButton**(text, icon);

โดย

JButton	คือ ตัวโค็ดคำสั่งการสร้าง button
bName	คือ ชื่อของออบเจกต์ที่ต้องการสร้างขึ้นใหม
text	คือ ข้อความที่ต้องการให้แสดงบนปุ่ม
icon	คือ รูปที่ต้องการให้แสดงบนปุ่ม

3.2 ป้ายแสดงข้อความ (Label) เป็นส่วนที่แสดงข้อความให้กับ Component อื่นๆ มีรูปแบบ การสร้าง ดังนี้

JLabel *lName* = new JLabel(text, icon, alignment);

โดย

JLabel	คือ ตัวโค๊ดคำสั่งการสร้าง Label
lName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม

text	คือ ข้อความที่ต้องการให้แสดงบน	เปุ่ม	
icon	คือ รูปที่ต้องการให้แสดงบนปุ่ม		
alignment	คือ ค่ำที่ใช้กำหนดการจัดวางข้อความ		
	SwingConstants.LEFT	คือ แสดงข้อความชิดซ้าย	
	SwingConstants.CENTER	คือ แสดงข้อความกึ่งกลาง	
	SwingConstants.RIGHT	คือ แสดงข้อความชิดขวา	

ตัวอย่างการใช้งาน Button และ Label (ภาพที่ 7.9)

1	import java.awt.*;
2	import javax.swing.*;
3	public class MyButtonLabel {
4	public static void main(String[] args) {
5	JFrame f = new JFrame("My Frame");
6	JPanel p = new JPanel();
7	Icon icon1 = new ImageIcon("Flag.gif");
8	Icon icon2 = new ImageIcon("cool.gif");
9	JButton jb = new JButton("Button", icon1);
10	JLabel jL1 = new JLabel("Data in Label",SwingConstants.RIGHT);
11	JLabel jL2 = new JLabel("Animation in Label", icon2, SwingConstants.LEFT);
12	f.add(p); p.add(jb); p.add(jL1); p.add(jL2);
13	f.setSize(200,100); f.setVisible(true);
14	f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); }
15	}

ภาพที่ 7.9 แสดงโปรแกรมและผลการทำงานของโปรแกรม Button และ Label

จากภาพที่ 7.9 พบว่าก่อนทำการเรียกใช้ภาพ (ต้องทำการจัดเตรียมไฟล์ Flag.gif และ cool.gif) ต้องทำการสร้าง icon ก่อนเช่นบรรทัดที่ 7-8 จึงสามารถนำมาใช้งานได้ดังบรรทัดที่ 10 และ 11 ตามลำดับ และส่วนที่สำคัญมากในการนำ panel วางใน frame ด้วยโค๊ด f.add(p); หลังจากนั้นจึง นำ Component มาวางใน panel ด้วยโค๊ด p.add(jb); p.add(jL1); และ p.add(jL2);

3.3 ช่องรับข้อความ (TextField) เป็นส่วนที่รับข้อความเพื่อนำข้อความไปประมวลผลต่อไป มี รูปแบบการสร้าง ดังนี้

JTextField tName = new JTextField (text, length);

โดย

JTextField	คือ ตัวโค๊ดคำสั่งการสร้าง TextField
tName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
text	คือ ข้อความที่ต้องการให้แสดงอธิบายช่องรับข้อความ
length	คือ จำนวนตัวอักษรที่ต้องการให้แสดงในช่องรับข้อความ

3.4 ช่องรับรหัสผ่าน (PasswordField) เป็นส่วนที่รับรหัสผ่าน โดยข้อมูลที่ป้อนจะไม่แสดง ตัวอักษรให้เห็น มีรูปแบบการสร้าง ดังนี้

JPasswordField *pwName* = new JPasswordField (text, length);

โดย

JPasswordField	คือ ตัวโค๊ดคำสั่งการสร้าง PasswordField
pwName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
text	คือ ข้อความที่ต้องการให้แสดงอธิบายช่องรับรหัสผ่าน
length	คือ จำนวนตัวอักษรในช่องรับรหัสผ่าน

ตัวอย่างการใช้งาน TextField และ PasswordField ทำการกำหนดให้มีข้อความแสดงสื่อสาร กับผู้ใช้โปรแกรมทราบว่าช่องรับข้อมูลนี้เป็นช่อง User Name และสามารถกรอกในช่องรับข้อมูลได้ จำนวน 20 ตัวอักษร ได้ด้วยโค๊ด JTextField jF = new JTextField("User Name", 20); ส่วนการ กำหนดช่องรหัสผ่านมีการกำหนดค่าเริ่มต้นเป็น 12345 และกำหนดให้สามารถกรอกรหัสผ่านได้สูงสุด จำนวน 15 ตัวอักษร เขียนโค๊ดได้เป็น JPasswordField pw = new JPasswordField("12345", 15); ดังตัวโค๊ดและผลการทำงานโปรแกรม (ภาพที่ 7.8)

1	import java.awt.*;
2	import javax.swing.*; Example for TextField and Password
3	public class MyTextPassword {
4	public static void main(String[] args) {
5	JFrame f = new JFrame("My Frame");
6	JPanel p = new JPanel();
7	JLabel jL = new JLabel("Example for TextField and Password");
8	JTextField jF = new JTextField("User Name", 20);
9	JPasswordField pw = new JPasswordField("12345", 15);
10	f.add(p); p.add(jL); p.add(jF); p.add(pw);
11	f.setSize(200,200); f.setVisible(true);
12	f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); }
13	}

ภาพที่ 7.8 แสดงการสร้าง TextField และ PasswordField

3.5 ตัวเลือกหนึ่งรายการ (RadioButton) เป็นตัวเลือกที่สามารถเลือกได้เพียงหนึ่งตัวเลือก เท่านั้น มีรูปแบบการสร้าง ดังนี้

JRadioButton *rbName* = new JRadioButton (text, icon, boolValue);

โดย

JRadioButton	คือ ตัวโค๊ดคำสั่งการสร้าง RadioButton
rbName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
text	คือ ข้อความที่ต้องการให้แสดงในรายการ
icon	คือ รูปที่ต้องการให้แสดงในรายการ
boolValue	้คือ ถ้ำค่าของรายการเป็น true แสดงว่าเป็นค่า
	ที่ถูกเลือกเป็นค่าเริ่มต้น

ซึ่งการสร้าง RadioButton เมื่อมีจำนวนมากๆ มีความจำเป็นต้องทำการรวมกลุ่มของ RadioButton ให้เป็นกลุ่มเดียวกัน สามารถทำได้โดยใช้โค๊ดสร้างอ็อบเจกต์จากเมธอด public ButtonGroup(); เพื่อนำแต่ละ RadioButton มารวมกัน 3.6 รายการข้อมูล (List) เป็นรายการที่สามารถเลือก โดยการเลื่อนรายการที่ต้องการเลือกได้ รูปแบบการสร้างรายการข้อมูล ดังนี้

JList ltName = new JList (item);

โดย

JListคือ ตัวโค๊ดคำสั่งการสร้าง list*ltName*คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่itemคือ ข้อมูลที่ต้องการให้แสดงในรายการซึ่งรายการข้อมูล ใช้งานร่วมกับอ็อบเจกต์บาร์ที่ในการเลื่อนรายการ (Scroll bar) ซึ่งสร้างจากคลาส ScrollPane มีรูปแบบดังนี้

JScrollPane spName = new JScrollPane (listName);

โดย

JScrollPane	คือ ตัวโค๊ดคำสั่งการสร้าง JScrollPane
spName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
listName	คือ เป็นรายข้อมูลของ list

ตัวอย่างการใช้งาน RadioButton และ list (ภาพที่ 7.9)

1	import java.awt.*;
2	import javax.swing.*;
3	public class MyRadioList {
4	public static void main(String[] args) {
5	String Lists[] = {"sun", "mon", "tue", "wen", "thu", "fin", "sat"}; wen thu
6	JFrame f = new JFrame("My Frame");
7	JPanel p = new JPanel();
8	ButtonGroup bg = new ButtonGroup();
9	JRadioButton rb1 = new JRadioButton("Male", true); JRadioButton rb2 = new JRadioButton("FeeMale");
10	JList aList = new JList(Lists); JScrollPane pl = new JScrollPane(aList);
11	bg.add(rb1); bg.add(rb2); aList.setPreferredSize(new Dimension(135, 140));
12	$a List.set Visible Row Count (5); a List.set Selection Mode (List Selection Model. MULTIPLE_INTERVAL_SELECTION); a List.set Selection Mode (List Selection Model. Model (List Selection Model (List $
13	$f.add(p); \qquad p.add(rb1); \qquad p.add(rb2); \qquad p.add(pl); \qquad f.setSize(200,200); \ f.setVisible(true);$
14	f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE); } }

ดังภาพที่ 7.9 แสดงโปรแกรมและผลการทำงานของ RadioButton และ list

3.7 ตัวเลือกรายการมากกว่าหนึ่งตัวเลือก (CheckBox) มีรูปแบบการสร้างรายการข้อมูล ดังนี้

JCheckBox ckName = new JCheckBox (text, icon);

โดย

JCheckBox	คือ ตัวโค๊ดคำสั่งการสร้าง CheckBox
ckName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
text	คือ ข้อมูลที่ต้องการให้แสดงในรายการ
icon	คือ ภาพ [ู] ที่ต้องการให้แสดงในรายการ

3.8 ตัวเลือกรายการหนึ่งตัวเลือก (ComboBox) มีรูปแบบการสร้างรายการข้อมูล ดังนี้

JComboBox *cbName* = new JComboBox (item);

โดย

JComboBox	คือ ตัวโค๊ดคำสั่งการสร้าง ComboBox
cbName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม
item	คือ ข้อมูลที่ต้องการให้แสดงในรายการ

ตัวอย่างการใช้งาน CheckBox และ ComboBox (ภาพที่ 7.10)

1	import java.awt.*;
2	import javax.swing.*;
3	public class MyChkComBox {
4	public static void main(String[] args) {
5	String Lists[] = {"sun", "mon", "tue", "wen", "thu", "fri", "sat"};
6	JFrame f = new JFrame("My Frame");
7	JPanel p = new JPanel();
8	JCheckBox ck1 = new JCheckBox("Jan"); JCheckBox ck2 = new JCheckBox("Feb");
9	JComboBox comB = new JComboBox(Lists);
10	comB.setPreferredSize(new Dimension(135, 40)); comB.setSelectedItem("fri"); comB.setMaximumRowCount(3);
11	f.add(p); p.add(ck1); p.add(ck2); p.add(comB); f.setSize(200,200); f.setVisible(true);
12	f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
13	}
14	}

ภาพที่ 7.10 แสดงโปรแกรมและผลการทำงาน CheckBox และ ComboBox

3.9 เมนู (Menu) เป็นตัวรายการให้ผู้ใช้งานเลือกรายการต่างๆ ซึ่งสามารถเพิ่มรายการตามที่ ต้องการได้ มีรูปแบบและขั้นตอนในการสร้างเมนู ดังนี้

> 3.9.1 สร้างเมนูบาร์ด้วย JMenuBar MyMenuBar = new JMenuBar(); คือ ชื่อเมนูบาร์ โดย MyMenuBar 3.9.2 สร้างเมนูด้วย JMenu MyMenu = new JMenu(StringName); คือ ชื่อรายการในเมนู โดย MyMenu 3.9.3 สร้างรายการเมนูย่อยด้วย JMenultem MyMenultem = new JMenultem(iTemName); คือ ชื่อรายการเมนูย่อย โดย MyMenultem คือ ข้อความในรายการเมนูย่อย iTemName 3.9.4 การเพิ่มรายการเมนูย่อยลงในเมนูด้วย MyMenu.add(MyMenultem); 3.9.5 การเพิ่มเมนูลงในเมนูบาร์ด้วย MyMenuBar.add(MyMenu); 3.9.6 การสั่งให้เมนูบาร์ลงในเฟรมด้วย FrameName.setJMenuBar(MyMenuBar);

้ตัวอย่างโปรแกรมต้องการสร้างการแสดงผลดังนี้

{

- O สร้างเมนูบาร์ ให้มีรายการในเมนูหลักเป็น File และ Help
- O ส่วนชื่อรายการเมนูของ File มี New Open และ Exit
- O รายการเมนูย่อยของ File ประกอบด้วย Window และ Massage

สามารถเขียนโปรแกรมด้านล่าง และแสดงผลการทำงาน (ภาพที่ 7.11)

```
import java.awt.*;
import javax.swing.*;
public class MyMenuBar
      public static void main(String[] args)
      {
             JFrame f = new JFrame("My Menu Bar");
             JPanel p = new Jpanel();
             JmenuBar MyMenuBar = new JmenuBar();
             Jmenu MyMenuFile = new Jmenu("File");
             Jmenu MyMenuHelp = new Jmenu("Help");
             Jmenu MyMenuNew = new Jmenu("New");
             Jmenultem MyMenultem1 = new Jmenultem("Window");
```

```
Jmenultem MyMenultem2 = new Jmenultem("Massage");
             MyMenuNew.add(MyMenuItem1);
             MyMenuNew.addSeparator();
             MyMenuNew.add(MyMenuItem2);
             Jmenultem MyMenultemOpen = new menultem("Open");
             Jmenultem MyMenultemExit = new Jmenultem("Exit");
             MyMenuFile.add(MyMenuNew);
             MyMenuFile.addSeparator();
             MyMenuFile.add(MyMenuItemOpen);
             MyMenuFile.addSeparator();
             MyMenuFile.add(MyMenuItemExit);
             MyMenuBar.add(MyMenuFile);
             MyMenuBar.add(MyMenuFile);
             MyMenuBar.add(MyMenuHelp);
             f.setJMenuBar(MyMenuBar);
             f.add(p);
             f.setSize(300,200);
             f.setVisible(true);
             f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
       }
}
```

ผลการทำงานของโปรแกรม (ภาพที่ 7.11)

🕌 My N	lenu Bar	<u> </u>	٢.
File He	lp		
New ▶	Window		
Open	Massage		
Exit			

ภาพที่ 7.11 แสดงผลการทำงานของเมนูบาร์

ขั้นตอนที่ 4 การจัดวางองค์ประกอบ (Layout Manager)

ในการจัดพื้นที่ส่วนของหน้าที่ติดต่อกับผู้ใช้งานเป็นสิ่งที่ต้องให้มีการสื่อสารที่ตรงกันระหว่างตัว ระบบกับผู้ใช้งาน ซึ่งการจัดองค์ประกอบคอมโพเนนต์ (Component) ลงบนพาเนล (panel)

FlowLayout เป็นพื้นฐานที่ได้ถูกกำหนดมาเป็นค่ามาตรฐานหรือ default นั้นเป็นการจัดวาง องค์ประกอบให้ทำการแสดงที่เรียงลำดับตามบรรทัดแรกให้เต็มบรรทัดก่อนจึงทำการให้แสดงในบรรทัด ถัดไป จนครบองค์ประกอบที่ถูกสั่งให้ทำการแสดง ซึ่งมีรูปแบบดังนี้

JPanel flName = new JPanel();
flName.setLayout(new FlowLayout());

โดย

JPanel	คือ ตัวโค๊ดคำสั่งการสร้าง Panel
flName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
setLayout();	คือ เมธอดสั่งให้ดำเนินการจัดองค์ประกอบในพาเนล

ตัวอย่างโปรแกรมการจัดองค์ประกอบดังด้านล่าง และผลการทำงานโปรแกรมแสดง (ภาพที่ 7.12 และภาพที่ 7.13)

import java.awt.*; import javax.swing.*; public class MyLayOut { public static void main(String[] args) { JFrame f = new JFrame("My Layout by default"); JPanel FL = new JPanel(); FL.setLayout(new FlowLayout()); JButton First = new JButton("First"); JButton Second = new JButton("Second"); JButton Third = new JButton("Third"); JButton Fourth = new JButton("Fourth"); JButton Fifth = new JButton("Fifth"); JButton Sixth = new JButton("Sixth"); JButton Seventh = new JButton("Seventh"); JButton Eighth = new JButton("Eigth");

```
JButton Tenth = new JButton("Tenth");
               FL.add(First);
               FL.add(Second);
               FL.add(Third);
               FL.add(Fourth);
               FL.add(Fifth);
               FL.add(Sixth);
               FL.add(Seventh);
               FL.add(Eighth);
               FL.add(Ninth);
               FL.add(Tenth);
               f.add(FL);
                      f.setSize(300,200);
                      f.setVisible(true);
                      f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        }
}
```

ผลการทำงานการจัดองค์ประกอบโดยขนาดของ Panel เป็นตาม f.setSize(300,200); (ภาพที่

7.12)

실 My Layout by d	efault	
First	Second	Third
Fourth	Fifth	Sixth
Seventh	Eigth	Ninth
	Tenth	

ภาพที่ 7.12 แสดงการจัดเรียง Button ตามขนาดที่กำหนด

169

JButton Ninth = new JButton("Ninth");

ผลการทำงานการจัดองค์ประกอบโดยขนาดของ Panel โดยทำการกำหนดขนาดให้ใหม่ตามที่ ผู้ใช้งานต้องการพบว่า ตัวคอมโพเนนต์ ที่เป็น Button จำนวน 10 ปุ่มมีการจัดเรียงใหม่ให้เหมาะสมกับ ขนาดของ Frame ที่อยู่บน panel (ภาพที่ 7.13)

🕌 My Layout by	default		[_ 0	×
First	Second	Third	Fourth	Fifth	
Sixth	Seventh	Eigth	Ninth	Tenth	

ภาพที่ 7.13 แสดงการจัดเรียง Button ตามขนาดที่ทำการขยายตามผู้ใช้งานกำหนด

BorderLayout เป็นการจัดวางองค์ประกอบให้ทำการแสดง ตามทิศทั้ง 5 ตำแหน่ง คือ ทิศเหนือ (North) ทิศใต้ (South) ทิศตะวันออก (East) ทิศตะวันตก (West) และจัดให้อยู่กึ่งกลาง ซึ่งมี รูปแบบดังนี้

JPanel bflName = new JPanel();
flName.setLayout(new BorderLayout());

โดย

JPanel	คือ ตัวโค๊ดคำสั่งการสร้าง Panel
bflName	คือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่
setLayout(new Border	Layout());
	คือ เมธอดสั่งให้ดำเนินการจัดองค์ประกอบในพาเนล
	ตามทิศทางของกรอบที่กำหนด ดังนี้
pName.add(bflName,	BorderLayout.Position);

โดย

pName	คือ ชื่อของพาเนลที่ต้องการให้อ็อบเจกต์ bflName มาเพิ่ม
BorderLayout.Position	คือ สั่งให้ทำการแสดงข้อมูลในกรอบตาม Position
Position	คือ ทิศทางที่ต้องการให้แสดงในกรอบของพาเนล
	มี 5 ตำแหน่ง NORTH, SOUTH, EAST, WEST และ
	CENTER

ตัวอย่างโปรแกรมด้านล่าง

```
import java.awt.*;
import javax.swing.*;
public class MyBorderLayOut
{
       public static void main(String[] args)
       {
              JFrame f = new JFrame("My BorderLayout");
              JPanel FL = new JPanel();
              FL.setLayout(new BorderLayout());
              JPanel BFL = new JPanel();
              //สร้างพาเนลสำหรับBorderLayOut
              BFL.setLayout(new BorderLayout());
              //กำหนดBorderLayOut ทำงาน
              FL.add(BFL, BorderLayout.NORTH);
              //กำหนดBorderLayOut ลงในพาเนลด้านบน
              JButton First = new JButton("First");
              //สร้างปุ่ม First
              JButton Second = new JButton("Second");
              JButton Third = new JButton("Third");
              JButton Fourth = new JButton("Fourth");
              JButton Fifth = new JButton("Fifth");
              BFL.add(First, BorderLayout.WEST);
              //กำหนดให้ปุ่ม First อยู่ด้านตะวันตกของกรอบ
              BFL.add(Second, BorderLayout.NORTH);
              BFL.add(Third, BorderLayout.CENTER);
              BFL.add(Fourth, BorderLayout.EAST);
              BFL.add(Fifth, BorderLayout.SOUTH);
              f.add(FL);
              //กำหนดให้พาเนลลงในเฟรม
              f.setSize(300,200);
              f.setVisible(true);
              f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);
        }
```

}

ผลการทำงานโปรแกรม (ภาพที่ 7.14)

My BorderL	ayout	- 0 <mark>- x</mark>
	Second	
First	Third	Fourth
	Fifth	
	Fifth	

ภาพที่ 7.14 แสดงการทำงานของโปรแกรมการกำหนด BorderLayOut

ข้อสังเกต ห้ามทำการสั่งให้แสดงในทิศทางซ้ำกันในพาเนลเดียวกัน เพาะทำให้ปุ่มที่สั่งให้แสดง ที่หลังทับซ้อนการแสดงปุ่มเดิม มีผลทำให้พาเนลไม่สามารถแสดงปุ่มเดิมได้ (ปุ่มเดิมหายไป) ดังนั้นการ แสดงด้วยการใช้ BorderLayOut ในหนึ่งพาเนลจึงสามารถแสดงได้เพียง 5 ทิศทาง ถ้าต้องการให้แสดง มากกว่า 5 ตำแหน่งหรือ 5 ปุ่ม ต้องทำการสร้างพาเนลเพิ่มมาอีกหนึ่งพาเนล หรือ อาจใช้วิธีการสั่งแสดง ปุ่มลงในตาราง ดังรายละเอียดของ GridLayout

GridLayout เป็นการจัดวางองค์ประกอบให้ทำการแสดงในช่อง (grid) เป็นลักษณะตารางที่ ประกอบด้วยแถวและคอลัมน์ โดยทำการจัดวางการแสดงจากด้านบนลงด้านล่าง และจากด้านซ้ายไป ด้านขวา ซึ่งมีรูปแบบดังนี้

JPanel glName = new JPanel();
glName.setLayout(new GridLayout(Row, Column));

โดย

JPanelคือ ตัวโค๊ดคำสั่งการสร้าง PanelglNameคือ ชื่อของอ็อบเจกต์ที่ต้องการสร้างขึ้นใหม่setLayout(new GridLayout(Row, Column));คือ เมธอดสั่งให้ดำเนินการจัดองค์ประกอบในพาเนลตามจำนวนขนาดทีกำหนดของ แถว (Row) และคอลัมน์(Column)

ตัวอย่างโปรแกรมด้านล่าง

import java.awt.*; import javax.swing.*; public class MyGridLayOut { public static void main(String[] args) { JFrame f = new JFrame("My GridLayout"); JPanel GL = new JPanel(); GL.setLayout(new GridLayout(5, 2)); JButton First = new JButton("First");

JButton Second = new JButton("Second"); JButton Third = new JButton("Third"); JButton Fourth = new JButton("Fourth"); JButton Fifth = new JButton("Fifth"); JButton Sixth = new JButton("Sixth"); JButton Seventh = new JButton("Seventh"); JButton Eighth = new JButton("Eighth"); JButton Ninth = new JButton("Ninth"); JButton Tenth = new JButton("Tenth");

GL.add(First); GL.add(Second); GL.add(Third); GL.add(Fourth); GL.add(Fifth); GL.add(Sixth); GL.add(Seventh); GL.add(Seventh); GL.add(Ninth); GL.add(Ninth);

f.add(GL); f.setSize(300,200); f.setVisible(true); f.setDefaultCloseOperation(JFrame.EXIT_ON_CLOSE);

}

}

ny ondeayour		
First	Second	
Third	Fourth	
Fifth	Sixth	
Seventh	Eighth	
Ninth	Tenth	

ผลการทำงานโปรแกรมการจัด GridLayout (ภาพที่ 7.15)



สร้าง Basic Event Handling

จากการเขียนส่วนติดต่อกับผู้ใช้งานข้างต้น เป็นการสื่อสารระหว่างตัวโปรแกรมกับผู้ใช้งาน เบื้องต้นที่ทำให้ผู้ใช้งานได้เห็นการแสดง ข้อมูลของตัวโปรแกรมที่ต้องการให้ผู้ใช้งานได้ทำการกำหนด ข้อมูลให้กับตัวโปรแกรม นำไปทำการประมวลผลต่อนั้น ดังนั้นการทำงานของโปรแกรมคอมพิวเตอร์ หลังจากได้ทำการสร้างหน้าต่างให้กับผู้ใช้งานแล้ว งานที่สำคัญอย่างยิ่งต่อจากนี้คือ การตรวจสอบการ ตอบสนองของผู้ใช้งานว่าได้ทำการสื่อสารกับมาได้การ คลิกเมาส์ หรือการกรอกข้อความและทำการกด Enter หรือยัง ดังนั้นตัวโปรแกรมต้องทำการสร้างออบเจ็กต์มาตรวจเหตุการณ์ (Even) อย่างต่อเนื่อง ตลอดเวลา ทำหน้าที่คอยเฝ้าระวังหรือคอยฟัง (Listener) ว่ามีการตอบสนองทางเมาส์หรือคีย์บอร์ด เมื่อใด จึงเรียกกระบวนการรวมๆ นี้ว่า Event Listener แต่ถ้าต้องการให้ออบเจ็กต์ทำหน้าที่ในการ ตรวจสอบกระบวนการการคลิกปุ่มของเมาส์มา 1 ออบเจ็กต์ ตัวโปรแกรมก็ต้องทำการส่งต่อเหตุการณ์ ไป ทำงานต่อส่วนอื่น ซึ่งขบวนการการจัดการรองรับเหตุการณ์ที่จะเกิดขึ้นแล้วส่งต่อไปยังขบวนการอื่น เช่นนี้จึงถูกว่า Event Handling นั่นเอง

โดยมีโค๊ดการทำงานแต่ละขั้นตอน ดังนี้

1. สร้างออบเจกต์ JRadioButton ด้วย

JRadioButton rB = new JRadioButton ("Single Choice");

 กำหนด Event ให้กับ ออบเจกต์ JRadioButton และ addActionListener (listener) เป็นคำสั่งที่ใช้ในการดักฟังเหตุการณ์เมื่อมีการคลิกที่คอมโพเนนต์ ด้วย

```
rB.addActionListener ( new ActionListener ( )
{
    public void actionPerformed ( ActionEvent event )
    {
        JRadioButton rB = ( JRadioButton ) event.getSource ( );
        boolean isSelected = rB.isSelected ( );
        if ( isSelected )
        JOptionPane.showMessageDialog (null,"Click to selected." );
        else
        JOptionPane.showMessageDialog (null,"Click to unselected.");
        }
    }
}
```

- การคืนค่าคอมโพเนนต์ เมื่อมีการเกิดเหตุการณ์ (Event) ขึ้น ด้วย JRadioButton rB = (JRadioButton) event.getSource ();
 โดย event.getSource () เป็นการคืนค่าคอมโพเนนต์ที่เกิดเหตุการณ์ขึ้น จึงแปลงเป็น ประเภท JRadioButton โดยเหตุการณ์เป็นออบเจกต์ที่สร้างจาก Class ActionEvent
- การตรวจสอบสถานะของ RadioButton ว่าถูกเลือกอยู่หรือไม่ถูกเลือกอยู่ ด้วย
 boolean isSelected = radioButton.isSelected ();
 - โดย * isSelected () จะคืนค่ากลับมาเป็นตัวแปรประเภท boolean
- 5. การแสดง Message Dialog ด้วย

JOptionPane.showMessageDialog (null, "Click to selected.");

ตัวอย่างโปรแกรมด้านล่าง

```
import java.awt.event.ActionEvent;
import java.awt.event.ActionListener;
import javax.swing.JFrame;
import javax.swing.JOptionPane;
import javax.swing.JRadioButton;
import javax.swing.WindowConstants;
public class AppListener
{
  public static void main (String [] args)
  {
     JFrame mF = new JFrame ();
     mF.setLayout ( null );
     JRadioButton rB = new JRadioButton ("Single Choice");
     rB.setLocation (100, 100);
     rB.setSize (120, 30);
     rB.addActionListener ( new ActionListener ( )
       {
           public void actionPerformed ( ActionEvent event )
              {
              JRadioButton rB = ( JRadioButton ) event.getSource ( );
              boolean isSelected = rB.isSelected ( );
              if (isSelected)
                      JOptionPane.showMessageDialog ( null, "Click to selected." );
                 else
                 JOptionPane.showMessageDialog ( null, "Click to un-selected." );
              }
        }
     );
     mF.add (rB);
     mF.setSize ( 500, 300 );
     mF.setDefaultCloseOperation (WindowConstants.EXIT ON CLOSE);
     mF.setVisible ( true );
  }
}
```

ผลการทำงานของโปรแกรม (ภาพที่ 7.16 และ 7.17) เมื่อผู้ใช้งานได้ทำการคลิกเลือก จะได้ Massage แสดงผลการทำงานของ Listener

<u>ب</u>		- • •
(Single Choice	

ภาพที่ 7.16 แสดงผลการทำงานโปรแกรม



ภาพที่ 7.17 แสดงผลการทำงานโปรแกรมและทำการคลิกเลือก RadioButton

ตัวอย่างโปรแกรม

ตามตัวอย่างที่ 7.1 เป็นโปรแกรมการใช้ Frame และ Panel พร้อมการกำหนดสี

```
import java.awt.Color;
import javax.swing.BorderFactory;
import javax.swing.BoxLayout;
import javax.swing.JFrame;
import javax.swing.JPanel;
import javax.swing.border.EtchedBorder;
public class Lab7 1
{
public static void main(String[] args) throws Exception
       ſ
 JFrame frame = new JFrame();
 frame.setVisible(true);
 frame.setSize(500, 200);
 frame.setLocation(10, 10);
 frame.setTitle("Frame By Java Code Exam");
 frame.setDefaultCloseOperation(JFrame.EXIT ON CLOSE);
 JPanel container = new JPanel();
 container.setLayout(new BoxLayout(container, BoxLayout.Y AXIS));
 JPanel panel1 = new JPanel();
 panel1.setBorder(BorderFactory.createLineBorder(Color.black));
 panel1.setBackground(Color.BLACK);
 container.add(panel1);
 JPanel panel2 = new JPanel();
 panel2.setBorder(BorderFactory.createEtchedBorder(EtchedBorder.RAISED));
 panel2.setBackground(Color.BLUE);
 container.add(panel2);
 JPanel panel3 = new JPanel();
 panel3.setBorder(BorderFactory.createEtchedBorder(EtchedBorder.LOWERED));
 panel3.setBackground(Color.CYAN);
 container.add(panel3);
```

```
JPanel panel4 = new JPanel();
panel4.setBorder(BorderFactory.createRaisedBevelBorder());
panel4.setBackground(Color.DARK_GRAY);
container.add(panel4);
JPanel panel5 = new JPanel();
panel5.setBorder(BorderFactory.createLoweredBevelBorder());
panel5.setBackground(Color.GREEN);
container.add(panel5);
frame.add(container);
}
```

ผลการทดลอง

🕌 Frame By Java Code Exam	

ภาพที่ 7.18 แสดงผลการทำงานโปรแกรม

ให้นักศึกษาปรึกษาและค้นคว้าในเขียนโปรแกรม โดยใช้เครื่องมือ ComboBox ให้ได้ผลการ ทำงานโปรแกรม ดังภาพที่ 7.19

🖆 ComboBox By JFrame		
	Bird ▼ Bird Cat Dog Rabbit Pig	

ภาพที่ 7.19 แสดงผลการทำงานโปรแกรมด้าน ComboBox

ตัวอย่าง 7.3 โปรแกรมในการแสดงค่าของเมธอด drawOval() กับเมธอด fillOval()

```
import javax.swing.JFrame;
import javax.swing.JOptionPane;
public class Lab7_3
{
    public static void main(String[] args) throws Exception
        {
            JFrame frame = new JFrame();
            String score = JOptionPane.showInputDialog("Input Your Score : ");
            JOptionPane.showMessageDialog(frame, "Your Score : " + score);
        }
}
```

สรุป

GUI มีส่วนประกอบหลักดังนี้ Containers เป็นส่วนที่ทำหน้าที่ในเก็บ Components ้ต่างๆ ที่ต้องแสดงหน้าเว็บ และComponents เป็นส่วนประกอบเป็นส่วนที่แสดงข้อมูล บนหน้าจอ เว็บ เช่น Button, Label, และ TextField ต่างๆ ในการโปรแกรม GUI ทุก component ต้องถูกวาง ใน container โดยทุก component สามารถวางด้วยเมธอด add (Component c) มีผลทำให้ Container สามารถเข้ามาทำงานได้ ซึ่งการโปรแกรม GUI ด้วย Swing และ AWT มีขั้นตอนดังนี้ ขั้นตอนที่ 1 สร้าง JFrame ด้วย JFrame *fName* = new JFrame("*Title*"); ขั้นตอนที่ 2 สร้าง JPanel ซึ่งเป็น component และต้องนำ JPanel ไปวางบนในส่วนของ container คือ JFrame ซึ่งมี รูปแบบของการสร้าง JPanel ด้วยJPanel pName = new JPanel(manager); ขั้นตอนที่ 3 สร้าง ้คลาสที่เป็น Component ที่ประกอบด้วยส่วนใดส่วนหนึ่ง ดังนี้ ป่ม (Button) ป้ายแสดงข้อความ (Label) ช่องรับข้อความ (TextField) ช่องรับรหัสผ่าน (PasswordField) ตัวเลือกหนึ่งรายการ (RadioButton) รายการข้อมูล (List) ตัวเลือกรายการมากกว่าหนึ่งตัวเลือก (CheckBox) ตัวเลือก รายการหนึ่งตัวเลือก (ComboBox) เมนู (Menu) มีรูปแบบและขั้นตอนในการสร้างเมนู และขั้นตอน การจัดวางองค์ประกอบ (Layout สดท้าย **ขั้นตอนที่ 4** Manager) ด้วย FlowLayout BorderLayout หรือ GridLayout เป็นต้น ส่วน สร้าง Basic Event Handling มีขั้นตอนในการสร้าง คือ 1. สร้างออบเจกต์ JRadioButton 2. กำหนด Event ให้กับ ออบเจกต์ JRadioButton และ addActionListener (listener) 3. การคืนค่าคอมโพเนนต์ เมื่อมีการเกิดเหตุการณ์ (Event) ขึ้น 4. การตรวจสอบสถานะของ RadioButton ว่าถูกเลือกอยู่หรือไม่ถูกเลือกอยู่ และ 5. การแสดง Message Dialog
แบบฝึกหัด

- 1. ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) มีความสำคัญอย่างไร
- 2. ส่วนติดต่อกับผู้ใช้แบบกราฟิก (GUI) มีกี่แพ็คเกจๆ อะไรบ้าง
- 3. โครงสร้างของ AWT Container Classes ประกอบด้วยส่วนใด้บ้าง
- 4. จงเขียนความแตกต่างระหว่าง Button กับ Label ของ JFrame
- 5. จงเขียนความแตกต่างระหว่าง TextField กับ PasswordField ของ JFrame
- 6. จงเขียนความแตกต่างระหว่าง ComboBox กับ CheckBox ของ JFrame
- 7. จงเขียนความแตกต่างระหว่าง FlowLayout กับ BorderLayout ของ JFrame
- 8. จงเขียนขบวนการพัฒนา Event Listener ว่าต้องใช้องค์ประกอบใดบ้างในการสร้าง
- จากโค็ดตัวอย่างด้านล่างต้องการให้ทำการปรับปรุงให้สามารถแสดงผลได้ import java.awt.*;

import java.awt.event.*;

public class TestEventPanel extends Panel {
Button b1,b2;

TestEventPanel(){

super();

setLayout(new FlowLayout());

setBackground(new Color(0).black);

b1 = new Button("call event on the frame");

add(b1);

b2 = new Button("close the parent frame"); add(b2);

}

}

10. จากโค๊ดด้านล่างจงเขียนแสดงผลการทำงาน

import javax.swing.*;

import java.awt.*;

import java.awt.event.*;

public class ButtonHomeWork extends JFrame {

private JButton B1,B2,B3;

ButtonHomeWork() {

setTitle ("Using Command Button ");

Container c = getContentPane();

c.setLayout (new FlowLayout());

B1 = new JButton("Red"); c.add (B1);

```
B2 = new JButton("Green"); c.add (B2);
B3 = new JButton("Blue"); c.add (B3); }
public static void main(String []args) {
ButtonHomeWork f = new ButtonHomeWork();
f.setSize(300,100); f.setVisible(true);
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e)
{System.exit(0); }
} );
}
svเขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด
```

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉ**บับพื้นฐาน.** กรุงเทพฯ: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

- . (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.
- สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/ %BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf

- สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555, จาก <u>www.thaiglossary.org</u>
- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Java GUI. ค้นเมื่อ Jun 16, 2012, จาก

http://javacodeexam.blogspot.com/2014/02/java-gui-jpanel.html

Java Swing. ค้นเมื่อ Jun 16, 2012, จาก

http://en.wikipedia.org/wiki/Swing (Java)

แผนบริหารการสอนประจำบทที่ 8

หัวข้อเนื้อหา

ไฟล์สตรีม ไบท์สตรีม การใช้คำสั่ง try ... catch / finally คาร์แร็คเตอร์สตรีม สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกความสำคัญการสร้างไฟล์สตรีม
- 2. บอกการตรวจสอบการทำงานด้วย try ... catch / finally
- 3. บอกการโปรแกรมคาร์แร็คเตอร์สตรีม
- 4. มีทักษะการโปรแกรมภาษาจาวาการรับ-ส่งข้อมูลด้วยไฟล์สตรีม
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนด

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อ ไฟล์สตรีม ไบท์สตรีม การใช้คำสั่ง try ... catch และ คาร์แร็คเตอร์สตรีม

 1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวาการการส่งข้อมูลไปยังไฟล์ พร้อมทั้งประยุกต์ไปใช้งานในการตัวจับด้วย try ... catch

 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์

- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 8
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั้งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 8 การติดต่อกับแฟ้มข้อมูล (Data file Connection)

การเขียนโปรแกรมให้มีประสิทธิภาพ ทั้งในด้านความถูกต้อง ความรวดเร็ว นั้นจำเป็นอย่างยิ่ง ต้องทำการจัดการกับข้อมูลก่อนทำการประมวลผล และหลังการประมวลผลให้เป็นระบบ การเขียน โปรแกรมพื้นฐานทั่วไป การประมวลผลข้อมูล จะทำการคำนวณข้อมูลต่างๆ ผ่านทางตัวแปรหรือ ออบเจ็กต์ แล้วนำผลการประมวลผล ไปแสดงผ่านทางหน้าจอภาพ เป็นการจบการทำงาน แต่ถ้า ต้องการเขียนโปรแกรมให้มีประสิทธิภาพสูงขึ้น ในการประมวลผลต้องการที่จะนำผลการคำนวณที่ ประมวลผลแล้ว ไปทำการจัดเก็บไว้ใช้งานในภายหลัง โปรแกรมต้องทำการติดต่อกับหน่วยความจำ ภายนอกซีพียู เพื่อนำข้อมูลที่ได้ไปบันทึกเป็นการสำรองข้อมูลไว้ ซึ่งหน่วยความจำภายนอกที่สามารถทำ การติดต่อได้สะดวกที่สุด เป็นการติดต่อกับแฟ้มข้อมูล (File) โดยแฟ้มข้อมูลนั้น เป็นตัวติดต่อกับ โปรแกรมซึ่งทำหน้าที่ทั้งสองด้านคือเป็น แฟ้มอินพุต (Input File) คือ เป็นการให้ตัวโปรแกรมทำการ อ่านข้อมูลจากแฟ้มข้อมูลมาประมวลผล ส่วนแฟ้มเอาต์พุต (Output File) เป็นการนำข้อมูลจาก การประมวลผลไปทำการจัดเก็บไว้ในแฟ้มข้อมูล นั่นเอง

ไฟล์สตรีม (File Stream)

ในการส่งข้อมูลระหว่างตัวโปรแกรม (Java Program) กับแหล่งที่เก็บข้อมูล (Data Source) จะเป็นกระบวนการที่เกิดขึ้นเสมอเมื่อตัวโปรแกรมมีการติดต่อกับแฟ้มข้อมูลในหน่วยความจำ สามารถ เกิดได้ 2 ลักษณะคือ ตัวโปรแกรมที่มีการสั่งให้นำข้อมูลจากแฟ้มข้อมูลในหน่วยความจำมาทำการ ประมวลผลที่ตัวโปรแกรม เรียกว่า การอ่าน (Read) ส่วนในด้านตรงกันข้ามคือ เมื่อตัวโปรแกรมต้องการ ส่งข้อมูลไปเก็บไว้ยังหน่วยความจำ เรียกว่า การเขียน (Write) โดยการเคลื่อนย้ายข้อมูลระหว่างตัว โปรแกรมกับหน่วยความจำมีลักษณะการส่งข้อมูลที่ทยอยส่งข้อมูลโดยทำการแบ่งข้อมูลเป็นส่วนย่อยๆ แล้วทำการส่งที่ส่วนผ่านทางท่อข้อมูลเสมือนสายน้ำที่ทยอยผ่านท่อจากฝั่งหนึ่งไปยังอีกฝั่งหนึ่ง จึงเรียก กระบวนการนี้ว่า สตรีม (Stream) (ภาพที่ 8.1)



ภาพที่ 8.1 แสดงการติดต่อกับแฟ้มข้อมูลกับตัวโปรแกรม

จากแนวคิดในการเคลื่อนย้ายข้อมูล จากภาพที่ 8.1 เป็นการทยอยข้อมูลส่วนเล็กๆ ผ่านทางท่อ ที่เรียกว่า สตรีม นั่น ในการติดต่อตัวโปรแกรมกับแฟ้มข้อมูล ต้องทำการตรวจสอบข้อมูลอยู่ตลอดเวลา ว่าข้อมูลระหว่างต้นทางกับปลายทางนั่น ได้ทำการเคลื่อนย้ายไปหมดสิ้นแล้วหรือยัง ด้วยในการติดต่อ แฟ้มข้อมูล ตัวโปรแกรมต้องทำการเปิด stream เพื่อทำการเชื่อมต่อท่อที่ใช้ในการเคลื่อนย้ายข้อมูล ต่อมาทำการตรวจสอบว่ามีการอ่านหรือเขียนข้อมูลหมดหรือยัง ถ้าเคลื่อนย้ายหมดแล้ว (ถ้าย้ายหมด โดยค่าใน Stream มีค่าเท่ากับ -1) จึงทำการปิด stream (ภาพที่ 8.2)



ภาพที่ 8.2 แสดงกระบวนการ File Stream

จากภาพที่ 8.2 เป็นโฟลว์ชาร์ต แสดงกระบวนการทำงานของสตรีมแฟ้มข้อมูล ที่มีการ เคลื่อนย้ายข้อมูล ในการขั้นตอนการอ่านหรือการเขียนข้อมูล ในแต่ละส่วนย่อยๆ ของข้อมูลนั้นสามารถ แบ่งลักษณะของข้อมูลว่าในการติดต่อแต่ละครั้งเป็นการของสตรีม ข้อมูลที่ได้เป็นลักษณะทีละ 1 ไบท์ (8 Bit) หรือข้อมูลเป็นลักษณะทีละ 1 ตัวอักษร (2 ไบท์: 16 บิท) จึงเรียกการสตรีมนี้ว่า ไบท์สตรีม (Byte Stream) และ คาร์แร็คเตอร์สตรีม (Character Stream) ตามลำดับ

ไบท์สตรีม (Byte Stream)

การเคลื่อนย้ายข้อมูลในลักษณะไบท์สตรีม เป็นการเคลื่อนย้ายข้อมูลระดับที่ละ 8 บิท หรือ 1 ไบท์ จึงทำการให้ข้อมูลที่เคลื่อนย้าย สามารถเรียกข้อมูล ว่า ข้อมูลรูปแบบภาษาเครื่อง (Machine Formatted Data) ในการเคลื่อนย้ายข้อมูล ด้วยคำสั่งการอ่านหรือการเขียน ต้องทำการเรียกแพ็กเกจ ด้วยการ import java.io.FileInputStream; และ import java.io.FileOutputStream; และต้องใช้ คลาส FileInputStream สำหรับการอ่าน และ FileOutputStream สำหรับการเขียน โดยมีรูปแบบ การสั่งงานดังนี้

การอ่านไฟล์สตรีม (File Stream Reading)

ในการทำงานของ File Stream ยังต้องมีความเข้าใจเพิ่มเติมเรื่องการจัดการข้อมูลที่ได้ทำการ อ่านหรือเขียนระหว่างตัวโปรแกรมกับแฟ้มข้อมูล ต้องมีการตรวจสอบความถูกต้องหรือ การดักจับเออ เรอร์ (Error) ทั้งด้านอินพุตและเอาต์พุต (IOException) ว่าแฟ้มข้อมูลที่ได้ทำการติดต่อนั้นมีอยู่จริง หรือมีข้อมูลอยู่หรือไม่ ถ้าไม่มีการทำงานของ IOException จะทำการฟ้องชนิดของการ Error ให้ผู้ใช้ โปรแกรมทราบข้อผิดพลาด ดังนั้นการสร้างคลาสสำหรับไฟล์สตรีม จึงมีรูปแบบดังนี้

public static void main(String[] args) throws IOException

มีรูปแบบการสั่งงาน การอ่านไฟล์สตรีม (File Stream Reading) ดังนี้

File filename	= new File ("path");
FileInputStream fInput	= new FileInputStream(filename);
<pre>fInput.read();</pre>	

โดย

path	ตำแหน่งที่อยู่ของแฟ้มข้อมูล
filename	ออบเจกต์ ที่สร้างขึ้นจากคลาส File เพื่อกำหนด File Stream ก่อนเสมอ
	ซึ่งสามารถสร้าง File Stream โดยกำหนด Path มาประกอบด้วยได้โดย
	ใช้ ":" ในการระบุ Drive และ ใช้ "/" ระบุ Path
fInput	ออบเจกต์ ที่สร้างขึ้นจากคลาส FileInputStream
read()	เมธอดในการอ่านข้อมูลจาก FileStream

เพื่อให้การทำงานด้าน File Stream นั้นไม่มีการ Error ด้าน Exception ดังนั้น ผู้พัฒนาโปรแกรมต้องทำการสร้างเตรียมแฟ้มข้อมูลที่ต้องการให้ตัวโปรแกรมได้ทำการติดต่อให้ เรียบร้อยก่อนทำการเรียกใช้งานไม่ว่าเป็นการอ่านหรือการเขียนสตรีม เช่น ในตัวอย่างโปรแกรมต่อไปนี้ มีการติดต่อกับแฟ้มข้อมูล DataForStream.txt อยู่ที่ในไดรฟ์ d: และในโฟลเดอร์ Java ดังนั้น จึงต้อง สร้างไฟล์ DataForStream.txt ด้วย NotePad แล้วทำการบันทึกไว้ใน d:\Java (ภาพที่ 8.3)

ſ	DataForStream - Notepad	×
	The data in this fire are data for file Stream.	*
	٩	.▼ 11. ₹

ภาพที่ 8.3 แสดงข้อความในแฟ้ม DataForStream.txt

ตัวอย่างการอ่านข้อมูลแบบ ไบท์สตรีม (ภาพที่ 8.4)

1	import java.io.*;
2	class MyReadStream {
3	<pre>public static void main(String[] args) throws IOException {</pre>
4	<pre>File fInStream = new File("D:/Java/DataForStream.txt");</pre>
5	FileInputStream fInput= new FileInputStream(fInStream);
6	int c;
7	<pre>while ((c=fInput.read()) != -1)</pre>
8	{ System.out.print(c); }
9	System.out.print("\n Read Data Complete");
10	}
11	}

ภาพที่ 8.4 แสดงโปรแกรมการอ่านไฟล์สตรีมด้วย Byte Stream

ผลการทำงานโปรแกรมการอ่านข้อมูลแบบ ไบท์สตรีม (ภาพที่ 8.5)



ภาพที่ 8.5 แสดงผลการทำงานโปรแกรม

จากผลการทำงานโปรแกรมดังภาพ 8.5 พบว่าโปรแกรมได้ทำการอ่านข้อมูลจาก DataForStream.txt เป็นค่าตัวเลขทั้งหมด ด้วยเพราะโปรแกรมเป็นการอ่านด้วย Byte Stream คือ การอ่านเป็น Machine Code นั่นเอง

ดังนั้นถ้าต้องการให้มีการอ่านข้อมูลที่อ่านจาก Text File สามารถอ่านได้ด้วยภาษาที่ถูกแปลง โค๊ดเรียบร้อยแล้ว ต้องทำการส่งข้อมูลผ่านทางเมธอดของการ write() ไฟล์ ในหัวข้อถัดไป

การเขียนไฟล์สตรีม (File Stream Writing)

มีรูปแบบการสั่งงาน ดังด้านล่าง

File filename	= new File ("path");
FileOutputStream fOutput	= new FileOutputStream(filename, append);
fOutput.write(data);	

โดย

path	ตำแหน่งที่อยู่ของแฟ้มข้อมูล
filename	ออบเจกต์ ที่สร้างขึ้นจากคลาส File เพื่อกำหนด File Stream ก่อนเสมอ
	ซึ่งสามารถสร้าง File Stream โดยกำหนด Path มาประกอบด้วยได้โดย
	สำหรับ ระบบวินโดว์ ใช้ ":" ในการระบุ Drive และ ใช้ "\" ระบุ Path
fOutput	ออบเจกต์ ที่สร้างขึ้นจากคลาส FileOutputStream
append	เป็นค่าที่กำหนดลักษณะการเขียนข้อมูลลงแฟ้มข้อมูล
	เท่ากับ true เป็นการสั่งให้เขียนต่อท [้] ้ยข้อมูลเดิม
	ถ้าเป็น false เป็นการสั่งให้เขียนทับแฟ้มที่ก่ำหนด (ข้อมูลเดิมถูกลบ)
data	ข้อมูลที่ต้องการเขียนลงในแฟ้มข้อมูล

ตัวอย่างการเขียนข้อมูลแบบ ไบท์สตรีม (ภาพที่ 8.6)

1	import java.io.*;	
2	class MyWriteStream {	
3	<pre>public static void main(String[] args) throws IOException {</pre>	
4	<pre>File fInStream = new File("D:/Java/DataForStream.txt");</pre>	
5	FileInputStream fInput= new FileInputStream(fInStream);	
6	FileOutputStream fOutput = new FileOutputStream("DataOut.tx	t");
7	int c;	
8	while ((c=fInput.read()) != -1) {	
9	<mark>System.out.write(c);</mark> //สั่งให้แสดงข้อความจอภาพ	
10	fOutput.write(c); } //สั่งเขียนข้อมูลลงใน DataOut.txt	
11	System.out.print("\n Write Data Complete");	
12	} }	

ผลการทำงานของโปรแกรมการ write Stream ในบรรทัดที่ 9 และ 11 (ภาพที่ 8.7)

Run
The data in this fire are data for file Stream.
Write Data Complete
Output completed (0 sec consumed) - Normal Termination

ภาพที่ 8.7 แสดงผลการทำงานของโปรแกรม

ผลการทำงานของโปรแกรมการ write Stream ในบรรทัดที่ 10 (ภาพที่ 8.8)



ภาพที่ 8.8 แสดงผลการทำงานของโปรแกรมที่ได้ทำการสร้าง DataOut.txt

จากการทำงานของโปรแกรมในภาพที่ 8.6 เป็นการสั่งให้อ่านข้อมูล(บรรทัดที่ 4) ที่แฟ้มข้อมูล D:/Java/DataForStream.txt มาเก็บไว้ที่ออบเจกต์ flnput ในตัวแปร c ด้วยคำสั่ง c=flnput.read() และทำการสั่งให้แสดงผลการอ่านด้วย System.out.write(c); ได้ผลการทำงานดังในภาพที่ 8.7 ส่วน การสั่งให้ทำการเขียนข้อมูลลงในแฟ้มข้อมูลด้วยคำสั่ง fOutput.write(c); ซึ่งจะการส่งข้อมูลไปยัง แฟ้มข้อมูล DataOut.txt ซึ่งได้ทำการกำหนดออบเจกต์ ชื่อ fOutput ด้วยคำสั่งในบรรทัดที่ 6 FileOutputStream fOutput = new FileOutputStream("DataOut.txt"); ดังนั้นเมื่อทำการสั่ง Run โปรแกรมแล้วนอกจากจะต้องตรวจสอบผลการทำงานได้ดังภาพที่ 8.7 แล้ว ต้องทำการตรวจสอบ แฟ้มข้อมูล DataOut.txt ด้วยว่ามีการเขียนข้อมูลลงในแฟ้มข้อมูล ดังแสดงในภาพที่ 8.8 ด้วย จึงถือว่า โปรแกรมทำงานได้สมบูรณ์แบบจริง ซึ่งสามารถทำการค้นหาได้ในโฟลเดอร์ที่อยู่ในที่อยู่ตัวโค๊ดโปรแกรม ในที่นี้แฟ้มข้อมูลอยู่ในที่เดียวกันกับแฟ้มข้อมูลสำหรับในการอ่าน คือ D:\Java จากนั้นสามารถเปิด แฟ้มข้อมูลด้วย NotePad หรือ Editor ตัวใดก็ได้ ซึ่งข้อมูลคือตัวอักษรทั้งหมด โปรแกรมได้ทำการอ่าน มาเก็บไว้ในตัวแปร c แล้วทำการเขียนที่ละตัวอักษรลงบนแฟ้ม DataOut.txt จนครบ

การใช้คำสั่ง try ... catch / finally

เป็นการจัดการ Exception มีโครงสร้างการทำงานคล้ายกับกลุ่มคำสั่ง if / else กล่าวคือถ้ามี การตรวจสอบเงื่อนไข ในที่นี้คือ Exception ที่อยู่หลัง try ว่ามีเหตุการณ์เกิดขึ้น ถือว่าถูกต้อง ก็จะให้ โปรแกรมมาประมวลผลหลังคำสั่ง catch ส่วน finally เป็นชุดคำสั่งให้ทำทุกเหตุการณ์ก่อนออกจาก try ดังแสดงตัวอย่างโปรแกรมดังภาพที่ 8.9 และผลการทำงานโปรแกรม (ภาพที่ 8.10)



ภาพที่ 8.9 แสดงโปรแกรมในการตรวจจับ Exception

----- Run -----

ERROR: DataForTryCatch.txt file not Found

java.io.FileNotFoundException: D:\Java\DataForTryCatch.txt (The system cannot find the file specified)

Output completed (0 sec consumed) - Normal Termination

ภาพที่ 8.10 แสดงผลการทำงานโปรแกรมในการตรวจจับ Exception

จากโปรแกรมภาพที่ 8.9 เป็นการเขียนโปรแกรมมาทำการมาตรวจว่าแฟ้มข้อมูล D:/Java/DataForTryCatch.txt มีอยู่ในหน่วยความจำหรือไม่ โดยการดักจับ Exception ของ try กับ catch ซึ่งใน catch มีการดักจับด้วยคำสั่ง (FileNotFoundException e) โดยทำการแสดงผลดักจับ ด้วยคำสั่ง System.out.println("ERROR: DataForTryCatch.txt file not Found"); ซึ่งก็แสดงผลดังที่ กำหนดของผู้เขียนโปรแกรม ส่วนในการฟ้อง Error ด้วยตัว try...catch เองนั้นได้ทำการแสดง Error ออกมาด้วยข้อความ java.io.FileNotFoundException: D:\Java\DataForTryCatch.txt (The system cannot find the file specified) ด้วยเพาะในโฟลเดอร์ D:\Java ไม่มีแฟ้มข้อมูล DataForTryCatch.txt นั่นเอง (ภาพที่ 8.10)

คาร์แร็คเตอร์สตรีม(Character Stream)

คาร์แร็คเตอร์สตรีม เป็นสตรีมที่มีการจัดการข้อมูลที่ได้ทำการติดต่อกับแฟ้มข้อมูล เป็นภาษาที่ สามารถอ่านข้อมูลได้ทันที่ และยังลดปัญหาของการติดต่อแฟ้มข้อมูลที่อยู่ในหน่วยความจำ ด้านในการ สร้างหรือเรียกใช้ไม่จำเป็นต้องมีแฟ้มข้อมูลอยู่ก่อน กล่าวคือเมื่อทำการเรียกใช้แฟ้มข้อมูลที่ไม่มีอยู่ใน หน่วยความจำ ก็จะทำการสร้างแฟ้มข้อมูลขึ้นมาตอบสนองการติดต่อของ Stream ได้ทันที โดยไม่มีการ ฟ้องการ error บน Exception e

การอ่านข้อมูลคาร์แร็คเตอร์สตรีม มีรูปแบบดังนี้

File filename	= new File ("path");
FileReader fInput	= new FileReader (filename);
data	= fInput .write();

โดย

ตำแหน่งที่อยู่ของแฟ้มข้อมูล
้ออบเจกต์ ที่สร้างขึ้นจากคลาส File เพื่อกำหนด File Stream ก่อนเสมอ
ซึ่งสามารถสร้าง File Stream โดยกำหนด Path มาประกอบด้วยได้โดย
สำหรับ ระบบวินโดว์ ใช้ ":" ในการระบุ Drive และ ใช้ "\" ระบุ Path
ออบเจกต์ ที่สร้างขึ้นจากคลาส FileReader
ข้อมูลที่อ่านจากแฟ้มข้อมูล

การเขียนข้อมูลคาร์แร็คเตอร์สตรีม มีรูปแบบดังนี้

File filename	= new File ("path");
FileWriter fOutput	= new FileWriter(filename [, append]);
fOutput.write(data);	

โดย

path	ตำแหน่งที่อยู่ของแฟ้มข้อมูล
filename	ออบเจกต์ ที่สร้างขึ้นจากคลาส File เพื่อกำหนด File Stream ก่อนเสมอ
	ซึ่งสามารถสร้าง File Stream โดยกำหนด Path มาประกอบด้วยได้โดย
	สำหรับ ระบบวินโดว์ ใช้ ":" ในการระบุ Drive และ ใช้ "\" ระบุ Path
fOutput	ออบเจกต์ ที่สร้างขึ้นจากคลาส FileWriter
[, append]	เป็นค่าที่กำหนดลักษณะการเขียนข้อมูลลงแฟ้มข้อมูลถ้าไม่กำหนด(มีค่าfalse)
	ค่าเป็น false เป็นการสั่งให้เขียนทับแฟ้มที่กำหนด (ข้อมูลเดิมถูกลบ)
	ถ้าเท่ากับ true เป็นการสั่งให้เขียนต่อท้ายข้อมูลเดิม
data	ข้อมูลที่ต้องการเขียนลงในแฟ้มข้อมูล

ตัวอย่างโปรแกรมดังแสดงด้านล่าง

```
import java.io.*;
import java.util.Scanner;
public class MyFileWriter
{
       public static void main(String[] args)
       {
       String path = "D:/Java/DataForChar.txt";
       FileWriter fOutput = null;
       Try
               {
               fOutput = new FileWriter(path);
               String s;
               Do
                      {
                       System.out.print("Press Character : ");
                       s = new Scanner(System.in).nextLine();
                              if (!(s.equals("*")))
                                                      {
                              fOutput.write(s);
                                                      }
                       }
```

```
while (!(s.equals("*")));
        }
        catch (IOException e)
        {
       System.out.println(e);
               }
        finally
        {
               try
               {
               if (fOutput != null)
                       {
                       fOutput.close();
                       }
               System.out.println("Saved Complete " + path);
               }
               catch (IOException e)
               {
               System.out.print(e);
               }
        }
FileReader fInput = null;
try
{
  fInput = new FileReader(path);
        int r;
       while ((r=fInput.read()) != -1)
               {
               System.out.print(r);
               }
       System.out.print("00000");
        }
catch (IOException e)
{
        System.out.print(e);
        }
```

```
finally
                {
                try
                        {
                        if (fInput != null)
                        {
                                fInput.close();
                        }
                        System.out.println("Complete ");
                }
                 catch (IOException e)
                        {
                        System.out.println(e);
                        }
                }
        }
}
```

จากโปรแกรมด้านบน มีการกำหนดตำแหน่ง (D:/Java) และชื่อของแฟ้มข้อมูล (DataForChar.txt) เก็บไว้ที่ path ด้วยคำสั่ง

path = "D:/Java/DataForChar.txt";

สร้างออบเจกต์ที่ใช้ในการ write file ด้วยคำสั่ง FileWriter fOutput = null; fOutput = new FileWriter(path);

ในการรับตัวอักษรจากแป้นคีย์บอร์ดในโปรแกรมนี้ได้ใช้การ Scanner โดยมีการตรวจสอบว่า ผู้ใช้งานต้องการหยุดเพิ่มตัวอักษรได้ด้วยการกดแป้นคีย์บอร์ด * ได้ด้วยคำสั่ง if (!(s.equals("*"))) และ ทำการ write ลงแฟ้มข้อมูลด้วยคำสั่ง fOutput.write(s); โดยให้ทำการส่งข้อมูลจนหมดด้วยคำสั่ง while (!(s.equals("*"))); โดยสามารถเขียนได้ดังนี้

```
String s;
                Do
                        {
                        System.out.print("Press Character : ");
                        s = new Scanner(System.in).nextLine();
                        if (!(s.equals("*")))
                                {
                                fOutput.write(s);
                                                         }
                        }
                while (!(s.equals("*")));
ซึ่งการรับข้อมูลจากการ Scanner ถูกควบคุมการทำงานด้วย
                        { ... }
                try
                catch { ... }
                finally { ... }
```

```
ส่วนการอ่านคาร์แร็คเตอร์จากแฟ้มข้อมูลใช้คำสั่ง
```

```
FileReader fInput = null;
fInput = new FileReader(path);
```

```
และทำการอ่านและแสดงผลการอ่านจอภาพด้วยคำสั่ง
int r;
while ((r=fInput.read()) != -1)
{
System.out.print(r);
}
```

้ส่วนกลุ่มคำสั่งสุดท้ายเป็นการคือค่าของ Exception กลับสู่ปกติด้วยการปิด Exception ต่างๆ ที่ไม่ใช้ งาน เพื่อให้หน่วยความจำได้มีพื้นที่ว่าง ยกเลิกการจองทั้งหมดลงด้วยคำสั่ง

```
finally {
 try {
    if (fInput != null) {
      fInput.close();
}
System.out.println("Complete ");
}
```

```
catch (IOException e)
{
System.out.println(e);
}
```

หลักจากทำการ Compiler แล้วทำการสั่ง Run โดยต้องทำการ ตัวอักษรที่ละตัวโดยผ่านทาง คีย์บอร์ด ดังนั้นในที่นี้จึงทำการสั่ง Run ผ่านทาง DosPrompt ด้วยการทำงาน (ภาพที่ 8.11)



ภาพที่ 8.11 แสดงขั้นตอนในการสั่งทำงานและป้อนข้อความ

จากนั้นทำการตรวจสอบผลการ writer ลงในแฟ้มข้อมูลที่ D:/Java/DataForChar.txt ว่ามีการ เก็บตัวอักษรที่กรอกไว้คือ ICE@TRU สมบูรณ์หรือไม่ได้ (ภาพที่ 8.12)



ภาพที่ 8.12 แสดงผลการบันทึกข้อมูลลงแฟ้มข้อมูล

ข้อสังเกต ในการติอดต่อแฟ้มข้อมูลด้วยข้อมูลคาร์แร็คเตอร์สตรีม แฟ้มข้อมูลที่โปรแกรมติดต่อ ครั้งแรกผู้ใช้งานไม่จำเป็นต้องทำการสร้างแฟ้มข้อมูลไว้ก่อนล่วงหน้า ทำให้ผู้เขียนโปแกรมสามารถ พัฒนาโปรแกรมเพื่อตอบสนองผู้ใช้งานทั่วไป ได้อีกสะดวก เป็นการลดขั้นตอนในการเตรียมงานด้าน จัดการแฟ้มข้อมูลได้เป็นอย่างดี

ตัวอย่างโปรแกรม

ตัวอย่างที่ 8.1 เป็นตามตัวอย่าง ในการใช้ Byte Stream และทำการจัดเตรียมแฟ้มข้อมูลที่ใช้ ในการติดต่อกับโปรแกรมแกม ชื่อ D:/Java/DataForLaB8.txt และ LaB8Out.txt โดยให้ได้ผลการ ทำงานดังผลการทดลองภาพที่ 8.13

```
import java.io.*;
class Lab8 1
{
       public static void main(String[] args) throws IOException
               {
               File fInfile
                                              = new File("D:/Java/DataForLaB8.txt");
               FileInputStream fln
                                              = new FileInputStream(fInfile);
               FileOutputStream fOut
                                              = new FileOutputStream("LaB8Out.txt");
               int c;
               while ((c=fIn.read()) != -1)
                       {
                                      System.out.write(c);
                                      fOut.write(c);
                              }
               System.out.print("\n Write Data in Lab");
               }
       }
```

ข้อมูลในแฟ้ม D:/Java/DataForLaB8.txt มีดังภาพที่ 8.13

DataForLaB8	- Notepad	Help			8
Data	for	Lab8,	Byte	Stream.	*
•					

ภาพที่ 8.13 แสดงข้อมูลในแฟ้ม DataForLaB8.txt

ผลการทำงานของโปรแกรมดังภาพที่ 8.14

Run
Data for Lab8, Byte Stream
Write Data in Lab
Output completed (0 sec consumed) - Normal Termination

ภาพที่ 8.14 แสดงผลการทำงานของโปรแกรม

ทำการตรวจสอบผลการส่ง Byte Stream มายังแฟ้มข้อมูลที่กำหนดให้เก็บข้อมูลในการ Write ในแฟ้มข้อมูล LaB8Out.txt ให้ถูกต้อง (ภาพที่ 8.15)

LaB8Out - No	otepad mat View	Heln			8
Data	for	Lab8,	Byte	Stream	*
					Ŧ
•				4	н

ภาพที่ 8.15 แสดงผลการ Write โดย Byte Stream มายังแฟ้ม LaB8Out.txt

ตัวอย่างที่ 8.2 ให้นักศึกษาศึกษาและค้นคว้าทำการเขียนโปรแกรม เรื่อง try... catch ให้ได้ผล การทำงานโปรแกรม (ภาพที่ 8.16)

----- Run -----

ERROR: By TryCatch

java.io.FileNotFoundException: D:\Java\LabTryCatch.txt (The system cannot find the file specified)

Output completed (0 sec consumed) - Normal Termination

ภาพที่ 8.16 แสดงผลการทำงานโปรแกรมด้าน try... catch

ตัวอย่าง 8.3 โปรแกรมในการแสดง character stream โดยใช้ try... catch ร่วมกับการใช้ คำสั่งการติดต่อแฟ้มข้อมูลระดับไฟล์ ด้วยคำสั่ง FileReader และ FileWriter

```
import java.io.*;
import java.util.Scanner;
public class Lab8_3
{
        public static void main(String[] args)
        {
        String file = "D:/Java/DataLab8_3.txt";
        FileWriter fOutput = null;
        Try
               {
                fOutput = new FileWriter(file);
                String s;
                do
                {
                        System.out.print("Press Character : ");
                        s = new Scanner(System.in).nextLine();
                        if (!(s.equals("!")))
                       {
                                fOutput.write(s);
                       }
                }
                while (!(s.equals("!")));
                }
        catch (IOException e)
                {
                        System.out.println(e);
                }
        finally {
                  try
                  {
                       if (fOutput != null)
                       {
                                fOutput.close();
                       }
```

```
System.out.println("Saved Complete " + file);
               }
catch (IOException e)
               {
               System.out.print(e);
               }
        }
       FileReader fInput = null;
       try
        {
               fInput = new FileReader(file);
               int r;
               while ((r=fInput.read()) != -1)
               {
                       System.out.print(r);
               }
         }
       catch (IOException e)
        {
               System.out.print(e);
               }
       finally
        {
        try
        {
               if (fInput != null)
               {
               fInput.close();
               }
       System.out.println("Complete ");
        }
       catch (IOException e)
        {
               System.out.println(e);
               }
        }
}
```

}

สรุป

ในการเขียนโปรแกรมเพื่อทำการส่งข้อมูลระหว่างตัวโปรแกรม 2 แฟ้มข้อมูลสามารถทำได้โดย การอ่านไฟล์สตรีม ซึ่งใช้โปรแกรมต้องทำการตรวจสอบความถูกต้องของข้อมูลในแฟ้มที่ต้องการติดต่อ ด้วยโค๊ด public static void main(String] args) throws IOException ก่อนด้วยการสั่งงาน ดังนี้ File *filename* = new File("path"); FileInputStream *finput*= new FileInputStream (filename); และ finput.read(); ส่วน การเขียนไฟล์สตรีม มีรูปแบบการสั่งงาน ดังนี้ **File** *filename* = new File("path"); FileOutputStream *fOutput* = new FileOutputStream (filename, append); fOutput.write (data); การใช้คำสั่ง try ... catch / finally เป็นการจัดการ Exception มี โครงสร้างการทำงานคล้ายกับกลุ่มคำสั่ง if / else ส่วนด้านคาร์แร็คเตอร์สตรีม เป็นการจัดการข้อมูล ที่ได้ทำการติดต่อกับแฟ้มข้อมูล เมื่อทำการเรียกใช้แฟ้มข้อมูลที่ไม่มีอยู่ในหน่วยความจำด้วยการอ่าน ข้อมูลคาร์แร็คเตอร์สตรีม มีรูปแบบคือ File *filename* = new File("path"); FileReader *finput* = new FileReader(filename); data = finput.write(); ด้านการเขียนข้อมูลคาร์แร็คเตอร์สตรีม มี รูปแบบ **File** *filename* = new File("path"); FileWriter *fOutput* = new FileWriter(filename [, append]); fOutput.write(data); เป็นตัน

แบบฝึกหัด

- 1. จงเขียนความแตกต่างระหว่าง Byte Stream กับ Character Stream
- 2. จงเขียนความสัมพันธ์ระหว่าง try { ... } กับ catch { ... }
- 3. จงเขียนโค้ดการสร้างออบเจกต์จากคลาส FileInputStream พร้อมอธิบายความหมาย
- 4. จงเขียนโค๊ดการสร้างออบเจกต์จากคลาส FileOutputStream พร้อมอธิบายความหมาย
- 5. จงเขียนโค้ดการการอ่านข้อมูลจากแฟ้มข้อมูลจนถึงจุดจบของแฟ้มข้อมูล (EOF: End of File)
- 6. จงเขียนโค๊ดการการเขียนข้อมูลไปยังแฟ้มข้อมูลจนถึงข้อมูลชุดสุดท้าย
- 7. อยากทราบว่าโค้ด while (!(s.equals("*"))); มีความหมายว่าอย่างไร
- 8. อยากทราบว่าโค๊ด catch (IOException e) มีความหมายว่าอย่างไร
- 9. อยากทราบว่าโค๊ด System.out.print(r); มีความหมายว่าอย่างไร
- 10. อยากทราบว่าโค๊ด fOutput.write(r); มีความหมายว่าอย่างไร

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉ**บับพื้นฐาน.** กรุงเทพฯ: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

- . (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.
- สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/ %BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf

- สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555, จาก <u>www.thaiglossary.org</u>
- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Java GUI. ค้นเมื่อ Jun 16, 2012, จาก

http://javacodeexam.blogspot.com/2014/02/java-gui-jpanel.html

Java Swing. ค้นเมื่อ Jun 16, 2012, จาก

http://en.wikipedia.org/wiki/Swing (Java)

แผนบริหารการสอนประจำบทที่ 9

หัวข้อเนื้อหา

เน็ตบีนส์ (Netbeans) มายเอ็สคิวแอล (MySql) การเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล การกำหนดคุณลักษณะมายเอ็สคิวแอล สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อนักศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกได้ว่าทำไมต้องใช้เน็ตบีนส์
- 2. อธิบายการคำสำคัญของมายเอ็สคิวแอล
- 3. บอกการโปรแกรมการเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล
- 4. มีทักษะการโปรแกรมภาษาจาวากับมายเอ็สคิวแอล
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนด

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อ เน็ตบีนส์ (Netbeans) และมายเอ็สคิวแอล (MySql)

1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวาการเชื่อมต่อ เน็ตบีนส์ (Netbeans) กับมายเอ็สคิวแอล (MySql)

 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์

- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 9
- 2. ระบบอินเทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม Edit Plus
- 4. ตัวแปลโปรแกรมภาษาจาวา (Java Compiler)
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั่งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 9 การประยุกต์การติดต่อฐานข้อมูลกับเครือข่าย (Applied connected Database with Network)

ในการเขียนโปรแกรมด้วยภาษาจาวา นักเขียนโปรแกรมนิยมใช้เครื่องมือที่ช่วยในการเขียนโค๊ด ้ประกอบด้วย 2 ตัว คือโปรแกรม Edit Plus เหมาะกับการพัฒนาโปรแกรมในระดับเบื้องต้น และเมื่อ ้ต้องการพัฒนาโปรแกรมภาษาจาวาให้มีความสามารถที่หลากหลายมากขึ้น รวมทั้งในการนำโปรแกรม ภาษาจาวามาทำการเชื่อมต่อกับฐานข้อมูล ที่มีการทำงานในลักษณะการเก็บข้อมูลไว้ในฐานข้อมูลบน ้เครื่องแม่ข่ายจริง หรือ ทำการเก็บข้อมูลไว้ในฐานข้อมูลบนเครื่องแม่ข่าย (Server) จำลอง ซึ่งต้องใช้ ้โปรแกรมตัวที่สองมาช่วยประกอบการทำงานให้อำนวยความสะดวกมากขึ้นด้วยNetbeansIDE ส่วน ด้านฐานข้อมูลที่นำมาประกอบการประยุกต์ในการทำงานนั้น พบว่าถ้าผู้พัฒนาโปรแกรมต้องการใช้ เทคโนโลยีที่มีระบบการทำงานแบบเบ็ดเสร็จ และอำนวยความสะดวกในทุกๆ ด้านโดยมีงบประมาณใน การพัฒนาโปรแกรมประยุกต์อย่างไม่จำกัด มักนิยมใช้เทคโนโลยีของไมโครซอฟต์ ซึ่งมีฐานข้อมูลให้ เลือกใช้สองระดับด้วยกันคือ ไมโครซอฟต์แอ็คเซส (Microsoft Access) เหมาะกับการพัฒนาฐานข้อมูล ที่นำมาใช้กับองค์กรขนาดเล็ก แต่ถ้าต้องการให้ระบบฐานข้อมูลมีอย่างไม่จำกัด โปรแกรมเอสคิวแอล เซิฟเวอร์ (SQL Server) สามารถทำงานได้เต็มระบบสมบูรณ์แบบที่ครบถ้วน ด้วยระบบฐานข้อมูลของ ตระกูลไมโครซอฟต์ ต้องมีงบประมาณในการดำเนินการ พัฒนาอย่างเหมาะสมด้วยเช่นกัน แต่ในการ พัฒนาโปรแกรมประยุกต์ในโลกเทคโนโลยีปัจจุบัน ก็มีทางเลือกให้กับนักพัฒนาโปรแกรมรุ่นใหม่ ได้มี เวที หรือช่องทางในการทำงานโดยไม่ผูกกับงบประมาณในการสร้างสรรค์งานได้อย่างเหมาะสม จึงมี ึกลุ่มที่รวมตัวกันที่ต้องการพัฒนาเทคโนโลยีให้เหมาะสมกับเหตุการณ์ปัจจุบันอย่างต่อเนื่อง โดยทำการ สร้างระบบฐานข้อมูลที่เป็นลักษณะที่ทุกคน มีส่วนร่วมสร้าง ร่วมพัฒนา ร่วมแบ่งบัน และมีความเป็น เจ้าของร่วมกัน ไม่นำเทคโนโลยีที่พัฒนาร่วมกันไปจำหน่าย แต่สามารถนำระบบฐานข้อมูลไปประยุกต์ใช้ กับองค์กรของแต่ละองค์กรได้อย่างอิสระเต็มที่ จึงทำให้เกิดระบบลิขสิทธิ์ที่เรียกว่า โอเพ็นซอร์ส (Open Source) ซึ่งฐานข้อมูลที่เป็นโอเพ็นซอร์ส ได้แก่ มายเอ็สคิวแอล (MySQL) นั่นเอง

ในบทที่นี้ประกอบด้วยเนื้อหา 4 ประเด็นคือ

- เน็ตบีนส์ (Netbeans)
- O มายเอ็สคิวแอล (MySql)
- การเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล
- O การกำหนดคุณลักษณะมายเอ็สคิวแอล

เน็ตบีนส์ (Netbeans)

เน็ตบีนส์ เป็นเครื่องมือสำหรับนักเขียนโปรแกรมที่ต้องพัฒนาโปรแกรมประยุกต์ (Application) ด้วยภาษาจาวา โดยถูกสร้างขึ้นเป็นโปรเจกต์นักศึกษา กลุ่ม "rock solid software" ใน ปี ค.ศ. 1998 โดยตั้งชื่อว่า เน็ตบีนส์ (NetBeans) และได้เผยแพร่ให้กับนักเขียนโปรแกรม และบุคคล ทั่วไปนำไปใช้งานได้ฟรีในรูปแบบโอเพ็นซอร์ส (Opensource software) ในปี ค.ศ. 2000 บริษัทซัน ไม โครซิสเต็มส์ ผู้พัฒนาภาษาจาวา ได้เข้ามาเป็นผู้สนับสนุนหลักในการพัฒนาเน็ตบีนส์ และได้ทำออกมา ในรูปแบบโอเพ็นซอร์ส โดยผู้ใช้งานไม่จำเป็นที่จะต้องเสียเงิน เพื่อซื้อมาใช้งาน และยังได้เปิดเผยซอร์ สโค๊ด (Soure code) ให้ผู้สนใจและนักพัฒนานำไปดัดแปลง แก้ไข ตามกฎของโอเพ็นซอร์ส ปัจจุบันมี นักโปรแกรมเมอร์ทั่วโลกต่างช่วยกันพัฒนาเน็ตบีนส์ให้มีความสามารถสูงยิ่งขึ้น

ปัจจุบัน ชุดพัฒนาเน็ตบีนส์ (NetBeans IDE) ได้รับความนิยมมากยิ่งขึ้น และได้รับการพัฒนา ให้มีความสามารถสูงยิ่งขึ้นเรื่อยๆ นอกจากจะใช้ในการพัฒนาโปรแกรมประยุกต์ ด้วยภาษาจาวาแล้ว ยัง สามารถพัฒนาอื่นๆได้อีกหลากหลายโดยติดตั้งโปรแกรมเสริม (Add-on) ได้จากเว็บไซต์ หรือผ่านตัวอัป เดตเซนเตอร์ (Update Center) ของเน็ตบีนส์ เช่น ภาษาซี/ซีพลัสพลัส (C/C++), Ruby, UML, SOA, Web Application, Java EE, Mobility(Java ME), Java FX, Java Script, PHP เป็นต้น

การติดตั้งเน็ตบีนส์

สามารถดาวน์โหลดได้ที่ <u>https://netbeans.org/downloads/</u> ซึ่งมีหน้าต่างให้เลือกทำการ ดาวน์โหลดได้ ภาพที่ 9.1

🕸 <u>NetBeans</u>	NetBeans IDE	NetBeans Platform	Plugins	Docs & Sup	port Community	y Partners	Sear	ch
HOME / Download	u.	NetBeans Platform	l.					
NetBeans IDE	8.0 Dow	nload					7.4 8.0 Develo	pment Archive
Email address (optional): Subscribe to newsletters:	✓ Monthly✓ NetBeans	Weekly can contact me at this	address	IDE Langu	age: English Note: Gre	Pla ved out technolo	atform: Windows ogies are not suppor	ted for this platform
					NetBeans	IDE Downlo	ad Bundles	
Supported technologie:	s*		J	ava SE	Java EE	C/C++	HTML5 & PHP	All
NetBeans Platform	m SDK			•	•			•
Java SE				•	•			•
Java FX				•	•			•
Java EE					•			•
💷 Java ME								•
ITML5					•		•	•
Iava Card™ 3 Cor	nnected							•
C/C++						•		•
Groovy								•
PHP							•	•
Bundled servers								
GlassFish Server	Open Source E	dition 4.0			•			•
Apache Tomcat 8.	.0.3				•			•
			D	ownload	Download	Download	Download	Download
			Fre	e, 90 MB	Free, 191 MB	Free, 62 MB	Free, 63 MB	Free, 210 MB

ภาพที่ 9.1 แสดงหน้าดาวน์โหลดหลักของ NetBeans

จากภาพที่ 9.1 ให้ทำการเลือกลักษณะที่ต้องการนำ NetBeans มาใช้งานตามความจำเป็น เท่านั้น ในที่นี้ต้องการนำมาทำงานร่วมกับ Java SE เท่านั้น ทำการเลือก Download ขนาด 90 MB ได้ (ภาพที่ 9.2)



ภาพที่ 9.2 แสดงสถานะการดาวน์โหลด netbeans-8.0-javase-...exe ขนาด 89.3 MB

เมื่อดาวน์โหลดสำเร็จให้ทำการติดตั้งตัวโปรแกรม Netbeans ด้วยการดับเบิลคลิกที่ตัว แฟ้มข้อมูลที่ได้ทำการบีบอัดมา ให้ทำการแตกแฟ้มข้อมูลตามขั้นตอนที่แนะนำมา (ภาพที่ 9.3)

🜍 netbeans-8.0-javase-windows	23/6/2557 20:39	Application	91,403 KB

ภาพที่ 9.3 ตัวโปรแกรม Netbeans

ทำการติดตั้ง Netbeans จะได้ขั้นตอนการติดตั้งให้เลือก Run (ภาพที่ 9.4)



ภาพที่ 9.4 แสดงเริ่มต้นการติดตั้ง Netbeans

ให้ทำการคลิก Next (ภาพที่ 9.5)

NetBeans IDE Installer	
	Welcome to the NetBeans IDE 8.0 Installer
	The installer will install the NetBeans IDE with the Java SE pack.
S NetBeans IDE	
	Installation Size: 327.3 MB
	Next > Cancel

ภาพที่ 9.5 แสดงยินดีต้อนรับ Netbeans 8.0

ให้ทำการยอบรับเงื่อนไขลิขสิทธิ์ ของบริษัท โอราเคิล จำกัด (Copyright © 1997, 2014, Oracle and/or its affiliates. (ภาพที่ 9.6 และภาพที่ 9.7)

icense Agreement	
Please read the following license agreement carefully.	A Merpeaus IDF
NETBEANS IDE 8.0 ("Product") LICENSE AGREEMENT	
PLEASE READ THE FOLLOWING LICENSE AGREEMENT TERMS AND	
CONDITIONS CAREFULLY, INCLUDING WITHOUT LIMITATION THOSE	
DISPLAYED ELSEWHERE (AS INDICATED BY LINKS LISTED BELOW),	
BEFORE USING THE SOFTWARE. THESE TERMS AND CONDITIONS	
CONSTITUTE A LEGAL AGREEMENT BETWEEN YOU, OR THE ENTITY FO	R
WHICH YOU ARE AN AUTHORIZED REPRESENTATIVE WITH FULL	
AUTHORITY TO ENTER INTO THIS AGREEMENT, AND ORACLE. BY	
CLICKING "ACCEPT" OR THE EQUIVALENT YOU AGREE TO ALL OF	
THE TERMS AND CONDITIONS OF THIS LICENSE AGREEMENT. IF YOU	
DO NOT AGREE TO THIS LICENSE DO NOT CLICK "ACCEPT" OR	
THE EQUIVALENT AND DO NOT INSTALL OR USE THIS SOFTWARE.	
Copyright (c) 1997, 2014, Oracle and/or its affiliates. All	
rights reserved.	
Oracle and Java are registered trademarks of Oracle and/or	
its affiliates. Other names may be trademarks of their	
respective owners.	
A cont the terms in the license agreement	
Accept the value in the left ac dig centering	

ภาพที่ 9.6 แสดงการยอบรับลิขสิทธิ์

the terms in the license agr	reement. Install JUniț			
stall JUnit				
			л.	
		< Back	Next >	Cancel
		< Back	Next >	

ภาพที่ 9.7 แสดงการยอบรับลิขสิทธิ์

ในขั้นตอนต่อไปเป็นขั้นตอนที่สำคัญเป็นการติดตั้ง Netbeans สำหรับการทำงานร่วมกันกับตัว โปรแกรมจาวา โดยต้องระบุตำแหน่งของโปรแกรมให้ถูกต้อง จะได้ไม่เกิดในการทำงานร่วมกันภายหลัง ซึ่งโดยปกติตัวติดตั้งจะทำการพิจารณาเลือกโฟลเดอร์ทีถูกต้องให้แล้วให้ทำการคลิก Next ได้ (ภาพที่ 9.8)

J NetBeans IDE Installer	
NetBeans IDE 8.0 Installation	RatDoone Inc
Choose the installation folder and JDK***.	A MELDEAUS IDE
Install the NetBeans IDE to:	
C:'Program Files (x86)'WetBeans 8.0	Browse,
JDK ¹⁴⁴ for the NetBeans IDE:	
C:\Program Files (x86)\Java\jdk1.7.0_55	- Browse

ภาพที่ 9.8 แสดงตำแหน่งของโปรแกรมที่จะทำการติดตั้ง

จากนั้นทำการติดตั้งด้วยการคลิก Install ซึ่งจะมีการแจ้งผลการทำการติดตั้ง จนครบ 100 % ให้ทำการคลิก Finish (ภาพที่ 9.9)



ภาพที่ 9.9 แสดงสถานการณ์ติดตั้งโปรแกรมและเสร็จสิ้นการติดตั้ง

เริ่มต้นการใช้งานเน็ตบีนส์

 เรียกโปรแกรมเน็ตบีนส์ จาก Start->All Programs ->NetBeans->NetBeans IDE 8.0 ได้ หน้าต่าง (ภาพที่ 9.10)

Seans IDE	Learn & Discover	My NetBeans What's New	Show On Startup 🛂
My NetBeans			
Recent Projects		Install Plugins	
<no recent<="" td=""><td>: project></td><td>Add support for other languages and techno NetBeans Update Center.</td><td>ogies by installing plugins from the</td></no>	: project>	Add support for other languages and techno NetBeans Update Center.	ogies by installing plugins from the
ORACLE			🥌 Java

ภาพที่ 9.10 แสดงหน้าต่างของเน็ตบีนส์

 ทำการสร้างโปรเจ็ค (Project) ด้วยการเลือกรายการเมนู File->New Project... โดยการทำการ เลือก Projects : เป็นแบบ Java Application แล้วทำการเลือก Next (ภาพที่ 9.11)

New Project	[×
Steps	Choose Project	
1. Choose Project	Q, Filter:	
	Categories: Projects:	
	Java Sava Application JavaFX Java Class Lbrary Maven Java Project with Existing Sources Naver Java Project with Existing Sources Image: Samples Java Free-Form Project	
	Description:	
	Creates a new Java SE application in a standard IDE project. You can also generate a main da in the project. Standard projects use an IDE-generated Ant build script to build, run, and debu your project.	ss .g
	< Back Next > Finish Cancel Help	

ภาพที่ 9.11 แสดงหน้าต่าง New Project

- ทำการกำหนดคุณลักษณะของ โปรเจ็ค ที่สร้างขึ้น โดย
 - O Project Name: ให้ตั้งชื่อโปรเจ็คที่ต้องการได้ ดังตัวอย่างนี้ใช้ชื่อ JavaApplication1
 - Project Location: ให้กำหนดที่อยู่ของตัวโค้ดว่าต้องการเก็บที่ใด ดังตัวอย่างนี้ให้เก็บไว้ที่
 D:/java/Netbeans
 - O Created Main Class เป็นการสร้างคลาสที่ต้องการได้ ถ้าไม่ต้องการกำหนดคลาสใหม่ ระบบจะทำการกำหนดให้เป็น javaapplication1.JavaApplication1 แต่ถ้าต้องการสร้าง คลาสใหม่ ชื่อ First ให้อยู่ในแพ็กเกจ javaapplication1 สามารถกำหนดได้ (ภาพที่ 9.12)



ภาพที่ 9.12 แสดงการสร้าง Project ใหม่

เขียนโค๊ดลงในโปรเจ็ค ให้ทำการแสดงข้อความ First time in Java on NetBeans (ภาพที่ 9.13)



ภาพที่ 9.13 แสดงหน้า IDE ของ NetBeans ที่ทำการเขียนโค๊ดได้

 สั่งให้ทำการทดสอบและการทำงานของโค๊ดด้วยการคลิกปุ่ม Play ได้ผลการทำงานของ โปรแกรม (ภาพที่ 9.14)



ภาพที่ 9.14 แสดงผลการทำงานของโค๊ดจาวาใน NetBeans

 ทำการบันทึกโค๊ดที่ทำการเขียนด้วยโปรแกรมภาษาจาวา บนเน็ตบีนต์ โดยปกติเมื่อทำการสั่งให้ทำ การทดสอบโปรแกรมและทำงาน(Run) ตัวเน็ตบีนส์ จะทำการบันทึกเป็นชื่อที่ตั้งโปรเจ็คครั้งแรก อัตโนมัติ ในที่นี้เป็นชื่อ First.java โดยทำการนำไปเก็บที่ แพ็กเกจ JavaApplication1 (ชื่อFolder) ใน ตำแหน่งที่ D:/Java แต่ถ้าต้องการทำการกำหนดชื่อใหม่สามารถทำได้โดย

การเลือก เมนู File->Save As และทำการกำหดนชื่อแฟ้มข้อมูลใหม่ได้ (ภาพที่ 9.15)



ภาพที่ 9.15 แสดงหน้าต่างการบันทึกแฟ้มจาวา (One.java)

มายเอ็สคิวแอล (MySQL)

มายเอ็สคิวแอล เป็นระบบฐานข้อมูลที่มีการยอมรับมากสุด ในกลุ่มของฐานข้อมูลชนิดโอเพ็น -ซอร์ส ด้วยมียอดการติดตั้งไปแล้วมากกว่า 16 ล้านราย ข้อมูลจาก<u>http://www.mysql.com/why-</u> mysql/white-papers/top-10-reasons-to-use-mysql-as-an-embedded-database/ เมื่อวันที่ 23 มิถุนายน 2557 และมียอดผู้ดาวน์โหลดใหม่ไม่ต่ำกว่า 10,000 รายต่อวัน โดยมายเอ็สคิวแอล ถูกจัดอันดับให้เป็นระบบฐานข้อมูลที่มีผู้ใช้เป็นอันดับสามของโลก ด้วยมายเอ็สคิวแอล มีการสนับสนุน จาก Oracle และ Microsoft SQL Serverมายเอ็สคิวแอล มีสมรรถนะและความสามารถในการขยาย ในการนำไปใช้งานทั้งด้านทั้งระบบโทรคมนาคม (Telecommunications) ด้านระบบการรักษาความ ปลอดภัย รวมทั้งการง่ายต่อการจัดเตรียมตามความต้องการสำหรับแพ็กเกจการใช้งาน SMB (Server Massage Block) และแพ็คเกจการใช้งานของซอฟท์แวร์ด้านการศึกษา(educational software) องค์กร ISV (Independent Software Vendor) ที่เป็นผู้พัฒนาระบบฐานข้อมูลที่ใช้มายเอ็สคิวแอล และกลุ่มลูกค้า (ภาพที่ 9.16)



ภาพที่ 9.16 แสดงกลุ่มองค์กรที่ได้ใช้มายเอ็สคิวแอล ที่มา (MySQL, 2014)

มายเอ็สคิวแอล ถูกออกแบบมาช่วยในการลดต้นทุน (cost) ในการพัฒนาระบบ ด้วยกลุ่มที่ ผู้พัฒนาได้จัดเตรียมการทำงานขั้นพื้นฐานให้กับผู้ใช้งานได้นำไปพัฒนาต่อตามลักษณะงานที่แตกต่างกัน ได้โดยไม่ต้องกังวนด้านลิขสิทธิ์ มายเอ็สคิวแอล เป็นระบบที่มีการต้นทุน (cost) ได้อย่างมีประสิทธิภาพ อย่างมาก เมื่อทำการวิเคราะห์เปรียบเทียบในระยะเวลาในการพัฒนาระบบฐานข้อมูล (TCO : Total Costs Over) ทั้งระบบจำนวน 3 ปี พบว่ามายเอ็สคิวแอล เป็นระบบที่มีค่าการบริหารระบบถูกที่สุด เมื่อ เทียบกับระบบของ Microsoft SQL Server และ Sybase ASE ด้วยข้อกำหนดทางฮาร์ดแวร์ (Hardware Configuration: Intel x86_64 Servers: 4, CPUs/Server: 4, Cores/Socket: 8) (ภาพที่ 9.17)


ภาพที่ 9.17 แสดงการเปรียบเทียบ TCO จำนวน 3 ปีของ 3 ระบบ ที่มา (TCO, 2014)

มายเอ็สคิวแอล (MySQL) เป็นระบบจัดการฐานข้อมูลเชิงสัมพันธ์ (Relational Database Management System) โดยใช้ภาษา SQL มายเอ็สคิวแอล สร้างขึ้นโดยชาวสวีเดน 2 คน และชาว ฟินแลนด์ ชื่อ David Axmark, Allan Larsson และ Michael "Monty" Widenius

ปัจจุบันบริษัทซันไมโครซิสเต็มส์ (Sun Microsystems, Inc.) เข้าซื้อกิจการของ MySQL AB เรียบร้อยแล้ว โดยมายเอ็สคิวแอล เป็นซอฟต์แวร์โอเพนซอร์ส ที่มีความแตกต่างจากซอฟต์แวร์โอเพน ซอร์สทั่วไป คือมีทั้งในแบบที่ให้ใช้ฟรี และแบบที่ใช้ในเชิงธุรกิจ

ชุดพัฒนาที่เรียกว่า AppServ เป็นชุดที่ใช้ในการติดตั้งระบบเครือข่ายคอมพิวเตอร์ (Apache HTTP Server) โปรแกรม PHP สำหรับการเขียนการควบคุมหน้าเว็บไซต์ โปรแกรมฐานข้อมูลมายเอ็ส คิวแอล (MySQL) และ phpMyAdmin เป็นชุดพัฒนาที่ติดตั้งและใช้งานได้ทันที สร้างโดยชาวไทย ซึ่ง ชุดพัฒนาสามารถดาวน์โหลด ได้ที่ <u>http://sourceforge.net/projects/appserv/files/</u> (ภาพที่ 9.18)



ภาพที่ 9.18 แสดงแหล่งดาวน์โหลด AppServ

การติดตั้ง มายเอ็สคิวแอลด้วย Appserv

ตัวชุดโปรแกรม Appserv ซึ่งรวมโปรแกรมที่ต้องการนำมาใช้งานในเนื้อหาการประยุกต์การ ติดต่อฐานข้อมูลกับเครือข่าย คือ มายเอ็สคิวแอล และ Apache HTTP Server เป็นโปรแกรมจำลอง เครือข่ายคอมพิวเตอร์ที่ทำหน้าที่เป็นเครื่องแม่ข่าย (Server) ดังนั้นจึงแนะนำให้ทำการติดตั้งในการ ประกอบการศึกษา มีขั้นตอนดังนี้

1. ทำการเปิดชุดโปรแกรมด้วยการ เลือกตัวชุดโปรแกรม Appserv จากที่ดาวน์โหลดและทำ การสั่งทำการติดตั้งด้วยการ คลิก Run (ภาพ 9.19)

appserv-win32-2.5.10	
	Open File - Security Warning
	The publisher could not be verified. Are you sure you want to run this software? Name:\Book\AppServ2_5_10\appServ-win32-2.5.10.exe Publisher: Unknown Publisher Type: Application From: D:\Skul\EdPlan\BasicruaPhyramming\Book\Ap Run Cancel
	☑ Always ask before opening this file
	This file does not have a valid digital signature that verifies its publisher. You should only run software from publishers you trust. How can I decide what software to run?

ภาพ 9.19 แสดงการเปิดโปรแกรม AppServ

2. ดำเนินการติดตั้งตามคำแนะนำของชุดโปรแกรม โดยการคลิก Next (ภาพที่ 9.20)



ภาพที่ 9.20 แสดงยินดีต้อนรับชุดโปรแกรม AppServ

 ทำการยอมรับลิขสิทธิ์ ด้วยการคลิก I Agree และในหน้าการเลือกตำแหน่งในการติดตั้ง สามารถกำหนดให้ทำการติดตั้งได้ ในที่นี้กำหนดเป็น C:\AppServ (เปลี่ยนได้โดยคลิก Browse) จากนั้นให้ทำการคลิก Next เพื่อเริ่มทำการติดตั้ง (ภาพที่ 9.21)

AppServ 2.5.10 Setup		
License Agreement Please review the license terms before installing A	AppServ 2.5.10.	
Press Page Down to see the rest of the agreement.	AppServ 2.5.10 Setup	
GNU LESSER GENERAL PUBLIC LICENSE Version 2. 1, February 1999	Choose Install Location Choose the folder in which to install AppServ 2.5.10.	
Copyright (C) 1991, 1999 Free Software Foundation, Inc. 59 Temple Place, Suite 330, Boston, MA 02111-1307 USA Everyone is permitted to copy and distribute verbatim copies of this license document, but changing it is not allowed. [This is the first released version of the Lesser GPL. It also counts as the successor of the GNU Library Public License, version 2, hence the version number 2.1.]	Setup will install AppServ 2.5.10 in the following folder. To install in a different folder, click Browse and select another folder. Click Next to continue.	
If you accept the terms of the agreement, click I Agree to continue. You must agreement to install AppServ 2.5, 10. Nullsoft Install System v2.18	Destination Folder	
< Back I Agree	Space required: 56.4MB Space available: 222.8GB Nullsoft Install System v2.18	
	< Back Next > Cancel	

ภาพที่ 9.21 แสดงขบวนการติดตั้ง

4. ทำการเลือกโปรแกรมที่ต้องการติดตั้งจากชุดโปรแกรม AppServ ในที่นี้ต้องการเลือก โปรแกรม Apache HTTP Server และโปรแกรม MySQL Database ด้วยการคลิก Next จากนั้น ระบบให้ทำการกำหนดข้อมูลของที่อยู่เครื่องแม่ข่าย (Apache HTTP Server Information) ซึ่งสามารถ กำหนดให้ทำการติดตั้งโปรแกรม Apache บนเว็บไซต์จริงได้ หรือต้องการจำลองการทำงานของเครื่อง แม่ข่ายจำลองได้ โดยทำการกำหนดได้ในช่อง Server Name (e.g. www.appserbnetwork.com). ใน ที่นี้ทำติดตั้งจำลองเครื่องคอมพิวเตอร์นี้ เป็นเครื่องแม่ข่ายที่ทำได้โดยการพิมพ์ localhost ซึ่งเป็นการ ให้การศึกษาการควบคุมเครื่องแม่ข่ายง่าย สะดวก ปลอดภัย ก่อนจะทำการนำไปพัฒนาทำงานบนระบบ เครื่องแม่ข่ายจริง ดังนั้น IP Address ของเครื่องโลคอลโฮส (Local Host) คือ 127.0.0.1 ที่ใช้ติดต่อ หรือเรียกผ่านทางโปรแกรมบราวเซอร์ (Browser) ต่างๆ เช่น Google Chrome, Internet Explorer, Fire Fox เป็นต้น เมื่อทำการติดตั้งที่อยู่ของเครื่องแม่ข่ายแล้ว ต้องทำการกำหนดสิทธิ์ของผู้บริหาร เครื่องแม่ข่ายในช่อง Administrator's Email Address (e.g. webmaster@gmail.com) ในที่นี้ให้ กำหนดเป็น root@localhost.com แล้วทำการคลิก Next (ภาพที่ 9.22)

AppServ 2.5.10 Setup Select Components Select the components you want to install, dear you do not want to install.	the components
AppServ Package Components	Apache HTTP Server Information Please enter your server's information.
 W MySQL Database PHP Hypertext Preprocessor phpMyAdmin 	Server Name (e.g. www.appservnetwork.com) localhost Administrator's Email Address (e.g. webmaster@gmail.com) root@localhost.com Apache HTTP Port (Default : 80) 80
Nullsoft Install System v2.18	Nullsoft Install System v2.18

ภาพที่ 9.22 แสดงรายละเอียดการติดตั้ง Apache HTTP Server

5. ทำการติดตั้ง MySQL Server Configuration ด้วยการกำหนด Enter Root Password และ Re-enter Root password ในที่นี้ให้ใช้รหัสเป็น **1234** ให้เหมือนกันทั้งสองช่อง แล้วทำการคลิก **Next** เพื่อสั่งให้ทำการติดตั้ง MySQL Server Configuration รอดำเนินการจนสำเร็จจะพบการแสดง กล่องข้อความ (Dialog) ว่า **Completing the AppServ 2.5.10 Setup Wizard** ให้ทำการคลิก **Finish** (ภาพที่ 9.23)

MySQL Server Configuration Configure the MySQL Server instance.		
Please enter Root password for MySQL Server. Enter root password Re-enter root password MySQL Server Setting Character Sets and Collations UTF-8 Unicode Old Password Support (PHP MySQL API function.) Enable InnoDB Nullsoft Install System v2.18 	AppServ 2.5.10 Setup	Completing the AppServ 2.5.10 Setup Wizard AppServ 2.5.10 has been installed on your computer. Click Finish to close this wizard. Istart Apache Start Apache Start MySQL

ภาพที่ 9.23 แสดงการติดตั้ง MySQL Server สำเร็จ

เมื่อทำการติดตั้งสำเร็จเรียบร้อยแล้ว สามารถทำการตรวจสอบการทำการของ AppServ ว่า เครือข่ายได้ทำการเปิดเครือข่ายสมบูรณ์หรือไม้ด้วยการตรวจสอบว่า Apache ได้ทำงาน (Start) พร้อมกับระบบฐานข้อมูล MySQL ได้พร้อมใช้งานได้ด้วย ให้ใช้บราวเซอร์ทำการเปิดแม่ข่ายจำลองที่ แอสเดรส หรือใน URL ที่ localhost หรือ IP Address: 127.0.0.1 (ภาพที่ 9.24, 9.25)

nne 🖻	สวระวัฒนที่สุนปสุนอาก	B avitin fun iPhone A	🕒 🗖 Intensive EnglishMEI	I E Home	🦰 แล มีล แร ไม่ สไต	🖻 ออันเม้เสียวอนไอเป็นแ	D MuSOL Invaltuto
фру 🗆	The second second			, include			E mysterate tate
'he A	ppServ C	Open Project - 2	2.5.10 for Window	s			
	he Mu Admin		largian 2 10 2				
- 85 P	DAD	Database Manager v	ersion 2.10.3				
2	The mormat	on version 3.2.0					
Abou	t AppServ Vers	sion 2.5.10 for Windows					
AppS	Serv is a mergir	ng open source software i	installer package for Windows	includes :			
	• Anacho V	Vah Server Version 2.2	8				
	 PHP Scrip 	nt Language Version 5	26				
	MySOL I	Database Version 5.0.5	1b				
	• phpMyAc	lmin Database Manag	er Version 2.10.3				
Chan	ael oa						
REAL	DME						
 AUTH 	HORS						
• COP	YING						
• Offic	ial Site : http:/	//www.AppServNetwork.c	om				
 Host 	ting support t	y:http://www.AppServH	losting.com				
Chang	ge Language	: 🔜 💻					

ภาพที่ 9.24 แสดงการเปิดโดย Google Chrome



ภาพที่ 9.25 แสดงการเปิดโดย Internet Explorer

การเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล

ในการพัฒนาโปรแกรมภาษาจาวาให้มีความสามารถในการเก็บสารสนเทศที่ต้องการอย่างเป็น ระบบ ระเบียบ ง่ายต่อการจัดเก็บ สะดวกในการค้นหานั้น ในระบบต้องอาศัยเทคโนโลยีของเน็ตบีนส์ ร่วมกับมายเอ็สคิวแอล ให้สามารถเชื่อมต่อกันได้ โดยมีขั้นตอนดังนี้

1. เปิดโปรแกรม NetBeans เพื่อทำการเปิด Services ของ Database ผ่านทาง Driver ให้ สามารถเชื่อมต่อ MySQL (ภาพที่ 9.26)



ภาพที่ 9.26 แสดงการเปิดระบบบริการ (Services) ของ NetBeans

เมื่อทำการเปิด Services ก็สามารถนำ NetBeans ทำการเชื่อมต่อกับ MySQL ผ่านทาง
 Localhost ซึ่งในการจัดการกับเครื่องแม่ข่ายจำลอง จำเป็นต้องใช้สิทธิ์ของผู้บริหารระบบ (Administration) ในที่ User Name ก็คือ root และรหัสผ่าน 1234 ตามที่ได้ทำการกำหนดไว้ตอนทำ การติดตั้ง AppServ นั่นเอง สามารถทำได้โดยการ เลือก Database->Driver ทำการคลิกขวาที่ MySQL (Connector/j driver) (ภาพ 9.27)

NetBeans IDE 8.0				
File Edit View Navigate Source	Refactor Run Debug Profile Team To	ols Window Help	Q- Sear	ch (Ctrl+I)
1 🔁 🔚 📲 🍤 🤇	🔍 <default config=""> 💽 🍸 🎇</default>	🕨 - 🎼 - 🕒 -		
Projects Files Servic 8	🛚 🖃 Start Page 📽 🚮 First. java 🕸			
Databases 🛟				
Drivers Java DB (Embedded)	NetBeans IDE	Learn & Discover	My NetBeans	What's New
MySQL (Connector (* Jacobian Connector (* Ja		ns		
Oracle Thin PostgreSQL	Delete Delete			
jdbc:derby://localhost: Web Services	Customize nt Project	s	Install P	lugins
	4		Add suppo	ort for other languages and t

ภาพ 9.27 แสดงการเชื่อม Database กับ MySQL

ทำการกำหนด Configuration ข้อมูลที่จำเป็นในการกำหนดให้กับ Host ที่เป็น localhost
 คือ User Name : คือ root ส่วน Password : คือ 1234 และทำการเลือกให้จำพาสเวิร์ดที่
 Remember Password แล้วทำการคลิก Next (ภาพที่ 9.28)

Customize Co	nnection
Driver Name:	MySQL (Connector/J driver)
Host:	localhost 🗢 Port: 3306
Database:	mysqi
User Name:	root 🧲
Password:	••••
	Remember password
	♠
	Connection Properties Test Connection
IDBC LIRL:	idbc:mvsal://localhost:3306/mvsal2zeroDateTimeBehavior=convertToNull
5556 6.42.	
	< Back Next > Finish Cancel Help

ภาพที่ 9.28 แสดง Wizard ของ MySQL



	File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help
	👚 🚰 鑘 🤚 🧭 🎑 🛛 🗠 🕼 🕞 🐨 🖓 🕨 - 🚯 - 🕦 -
	Projects Files Services 🕷 🖬 Start Page 🕷 🚳 First.java 🕫
New Connection Wizard	Databases
Choose Database Schema	Java DB Jorvers Sector Drivers Sector Drivers
For each database connection, the Services window only displays objects from one Select the schema of the tables to be displayed.	
Select schema:	
	Web Services
	Navigator %
	<no available="" view=""></no>
0	
< Back Next > Finish	Cancel Help

ภาพที่ 9.29 แสดงผลการเชื่อมต่อ NetBeans กับ MySQL สมบูรณ์

การเพิ่ม Library ของ NetBeans กับ MySQL jdbc Driver

ในการทำงานร่วมกันระหว่าง NetBeans กับ MySQL มีความจำเป็นต้องอาศัย Driver ของ ฐานข้อมูล MySQL อย่างต่อเนื่อง ในการแปลงข้อมูลระหว่างกัน สามารถการเพิ่ม Library ของ NetBeans กับ MySQL jdbc Driver โดยการคลิกขวาที่ Libraries แล้วทำการเลือก Add Library แสดงดังภาพ 9.36 แล้วทำการเลือก MySQL JDBC Driver (ภาพ 9.30)



ดังภาพ 9.30 แสดงขั้นตอนการ Add Libraries ของ NetBeans

224



ดังภาพ 9.31 แสดงการเลือก MySQL JDBC Driver

การกำหนดคุณลักษณะมายเอ็สคิวแอล (Configuring MySQL Server Properties)

 ทำการคลิกขวาที่ Databases->MySQL Server at localalhost: 3306 [root] (disconnection) แล้วทำการเลือก Properties... (ภาพที่ 9.32)

NetBeans IDE 8.0						
File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help Q- Search (Ctrl+I)						
*** *** 🚰 📲 🐚 🥙 🔚						
Projects Files Services %	Start Page 🛛 🕺 🗟 FirstDataBa	ase.java 🛛				
Databases MySQL Server at localhost: 3306 [root] (disconnecter)	Create Database		<u> </u>			
	Start Stop	Learn & Discover M	ly NetBeans V			
⊕- b] jdbc:mysql://localhost:3306/mysql?zeroDateTimeBet ⊕-	Connect Disconnect	III	× F			
Hudson Builders Task Repositories	Delete Refresh					
<	Run Administration Tool Properties					
		(L) INS			

ภาพที่ 9.32 แสดงการกำหนด MySQL Server Properties

 o ทำการกำหนด Administration Password เป็น 1234 (สามารถเว้นว่างไว้ได้) ตามที่กำหนดขณะ ทำการติดตั้ง และเมื่อต้องการให้มีการจำรหัส สามารถทำได้โดยการทำเครื่องหมายในบล็อก Remember Password แล้วคลิก OK (ภาพที่ 9.33)

Projects	Files	Services %	🖬 Start Page 🕱 🐼 FirstDataBase.java 🕱					
Dat								
	MyS0	QL Server Prope	ties	×				
⊕- %	Basic Pro	operties Admin P	operties					
🕀 🧟 We	Server	Host Name:	localhost					
Hur	Server	Port Number:	3306					
⊕ 🙀 Tas	Adminis	trator User Name	root					
	Adminis	trator Password:	🤁					
•	V	Remember Passwo	d					
			ОК Са	ncel Help				

ภาพที่ 9.33 แสดงการกำหนดรหัสผ่านของ Root ของ MySQL

O ทำการกำหนดคุณสมบัติของผู้บริหารระบบได้โดยการคลิกแท็บ Admin Properties โดยทำการ กำหนดในส่วนของ Path/URL to admin tool: ส่วนของ Path to start command: และส่วน ของ Path to stop command: ซึ่งมีปุ่ม Browse... ให้ทำการค้นหาตามที่ได้ทำการติดตั้งชุด โปรแกรม AppServ ไว้ตอนต้นและต้องกำหนดให้ตรงแสดงในภาพที่ 9.34 ได้ผลการภาพที่ 9.35

MySQL Server Properties	a terring a production of a	×
Basic Properties Admin Properties		
Path/URL to admin tool:	C: \AppServ \MySQL \bin \mysqladmin.exe	Browse
Arguments:		
Path to start command:	C:\AppServ\MySQL\bin\mysqld-nt.exe	Browse
Arguments:		
Path to stop command:	C:\AppServ\MySQL\bin\mysqladmin.exe	Browse
Arguments:	-u root stop	
	OK Cancel	Help

ภาพที่ 9.34 แสดงข้อกำหนดของ Admin ในการกำหนด Path



ภาพที่ 9.35 แสดงข้อกำหนดของ Admin สมบูรณ์

สรุป

ในการที่ทำการเขียนโปรแกรมจาวาให้สามารถทำงานร่วมกับฐานข้อมูล และเครือข่าย จำเป็นต้องมีโปรแกรมที่ช่วยในการจัดการที่เป็นชุดโปรแกรมที่ได้จัดเตรียมไว้ให้สามารถทำงานร่วมกันได้ อย่างไม่ติดขัด ซึ่งตัวโปรแกรมเน็ตบีนส์ เป็นโปรแกรมที่บริษัท ซัน จำกัด ได้แนะนำและสนับสนุนให้ชุด พัฒนาด้วย ซึ่งเหมาะกับการใช้งานตัวชุดพัฒนาด้านฐานข้อมูลอย่างดีเยี่ยมและถูกต้องตามลิขสิทธิ์ ของ การทำงานได้ด้วยการใช้ชุดโปรแกรมที่ AppServ ซึ่งจะอำนวยความสะดวกด้วยความสามารในด้าน การเพิ่ม Library ของ NetBeans กับ MySQL jdbc Driver รวมทั้งยังสามารถ การกำหนดคุณลักษณะ มายเอ็สคิวแอล ได้อย่างอิสระด้วย

แบบฝึกหัด

- 1. จงเขียนสรุปความแตกต่างระหว่าง EditPlus กับ NetBeans มาพอสังเขป
- 2. จงเขียนสรุปความแตกต่างระหว่าง Microsoft Access กับ MySQL มาพอสังเขป
- 3. จงเขียนคุณลักษณะของ NetBeans ความสามารถร่วมกับโปรแกรมอะไรได้บ้าง
- 4. จงเขียนโค้ดการสร้างออบเจกต์จาก NetBeans พร้อมอธิบายความหมาย
- 5. จงเขียนเหตุผลว่าทำไม MySQL จึงเป็นที่นิยมของนักพัฒนาโปรแกรม
- 6. จงเขียนความสัมพันธ์ระหว่าง MySQL กับ Appserv มาพอสังเขป
- 7. จงให้ความหมายของ localhost กับ 127.0.0.1 คืออะไร
- การกำหนดให้ Administrator's Email Address เป็น root@localhost.com หมายความว่า อย่างไร
- 9. อยากทราบว่าการสร้าง Project ใหม่ของ Netbeans หมายความว่าอย่างไร
- 10. ในการติดตั้ง AppServ สามารถเลือกโปรแกรมที่ต้องการติดตั้งได้อย่างไร

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉ**บับพื้นฐาน.** กรุงเทพฯ: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

- . (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.
- สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/ %BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf

- สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555, จาก <u>www.thaiglossary.org</u>
- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Armstrong, E. (2004). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Appserv. ค้นเมื่อ Jun 23, 2012, จาก

http://www.appservnetwork.com/modules.php?name=Content&pa =showpage&pid=3

Appserv files. ค้นเมื่อ Jun 23, 2012, จาก <u>http://sourceforge.net/projects/appserv/files/</u>

NetBeans MySQL. ค้นเมื่อ Jun 28, 2012, จาก

https://netbeans.org/kb/docs/ide/mysql.html

MySQL. ค้นเมื่อ Jun 23, 2012, จาก

http://www.mysql.com/why-mysql/white-papers/top-10-reasons-to-use-mysqlas-an-embedded-database/

- MySQL Database. ค้นเมื่อ Jun 28, 2012, จาก <u>http://www.java2s.com/Code/Java/Database-SQL-JDBC/</u> <u>CreatingaDatabaseinMySQL.htm</u>
- NetBeans. ค้นเมื่อ Jun 23, 2012, จาก <u>https://netbeans.org/downloads/</u>
- Java Swing. ค้นเมื่อ Jun 16, 2012, จาก <u>http://en.wikipedia.org/wiki/Swing_(Java)</u>

แผนบริหารการสอนประจำบทที่ 10

หัวข้อเนื้อหา

สร้างและทำการเชื่อมต่อฐานข้อมูลให้พร้อมใช้งาน สร้างตารางให้กับฐานข้อมูล การสั่งงานตารางข้อมูล การสั่งการงานฐานข้อมูลด้วย Script แนะนำการโปรแกรมบนเครือข่ายด้วยจาวา สรุป แบบฝึกหัด เอกสารอ้างอิง

วัตถุประสงค์เชิงพฤติกรรม

เมื่อศึกษาได้ศึกษาเนื้อหาในบทเรียนนี้แล้วสามารถ

- 1. บอกวิธีการสร้างตารางในระบบฐานข้อมูล
- 2. อธิบายการควบคุมตารางฐานข้อมูล
- 3. บอกการเขียน Script ทำงานร่วมกับการทำงานของฐานข้อมูล
- 4. มีทักษะการโปรแกรมภาษาจาวาในรูปแบบการเขียน Script
- 5. การเขียนโปรแกรมภาษาจาวาประยุกต์ใช้งานตามที่กำหนด

วิธีการสอนและกิจกรรมการเรียนการสอน

1. วิธีสอน

1.1 ใช้วิธีสอนแบบบรรยายในหัวข้อ การเชื่อมต่อฐานข้อมูล ตารางให้กับฐานข้อมูล การสั่งงานตารางข้อมูล และแนะนำการโปรแกรมบนเครือข่ายด้วยจาวา

1.2 ใช้วิธีสอนแบบปฏิบัติการเขียนโปรแกรมภาษาจาวาการเชื่อมต่อฐานข้อมูลและ การควบคุมผ่านเครือข่าย

- 1.3 ใช้วิธีสอนแบบปฏิบัติการกำหนดโปรแกรมให้นักศึกษาได้ทดลองบนเครื่อง คอมพิวเตอร์
- 2. กิจกรรมการเรียนการสอน
 - 2.1 ให้นักศึกษาปฏิบัติการตามที่ผู้สอนสาธิตการโปรแกรมภาษาจาวา
 - 2.2 ให้นักศึกษาอภิปรายผลการทดลอง
 - 2.3 ทำแบบฝึกหัด

สื่อการเรียนการสอน

- 1. แฟ้มข้อมูลการนำเสนอ ของบทที่ 10
- 2. ระบบอิน เทอร์เน็ต สำหรับค้นคว้าโค๊ดเพิ่มเติม
- 3. เครื่องคอมพิวเตอร์และตัวโปรแกรม NetBeans
- 4. ชุดโปรแกรม Appserv
- 5. ระบบเครือข่ายคอมพิวเตอร์ ในห้องปฏิบัติการคอมพิวเตอร์
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)
- 7. เอกสารประกอบการสอน ตำรา เอกสารที่เกี่ยวข้อง

การวัดและการประเมินผล

- การสังเกตจากความสนใจ และการมีส่วนร่วมระหว่างเรียน ด้วย การซักถามของผู้สอน หรือการตอบสนองด้วยการตอบคำถาม หรือพฤติกรรมในการรับรู้ รับฟัง
- การวัดความสามารถผู้เรียนได้ด้วยผลการทดลอง ใบงานหลังบทเรียน ในด้านทักษะการใช้ เครื่องคอมพิวเตอร์ การเขียนโค๊ดโปรแกรม การแก้ปัญหาในการทดลองระหว่างเรียน
- การตอบคำถามท้ายบท ของผู้เรียนมีความสนใจ กระตือรือรันในการทำงาน การมีภาวะ ผู้นำในห้องเรียน การกล้าเสนอความคิดเห็นในการทำงาน
- 4. สั้งเกตจากการฝึกและผลงานภาคปฏิบัติ
- 5. สังเกตจากความกระตือรือรั้นของนักศึกษา
- ตรวจแบบฝึกหัด

บทที่ 10 การพัฒนาฐานข้อมูลด้วยจาวา (Development Database with Java)

จากการกำหนดคุณลักษณะมายเอ็สคิวแอล (Configuring MySQL Server Properties) ของ บทที่ผ่านมา สามารถนำ NetBeans ทำการเชื่อมต่อกับ MySQL ผ่านทาง Localhost ซึ่งใช้การจัดการ กับเครื่องแม่ข่ายจำลอง ด้วยสิทธิ์ของผู้บริหารระบบ (Administration) คือroot และรหัสผ่าน 1234 ได้ สมบูรณ์เรียบร้อยแล้ว ลำดับถัดมาทำการสร้างแฟ้มฐานข้อมูลและตารางที่ประกอบในการเก็บข้อมูล ตามขั้นตอนต่อไปนี้

สร้างและทำการเชื่อมต่อฐานข้อมูลให้พร้อมใช้งาน (Creating and Connecting to the Database Instance)

ปกติทั่วไปในการติดต่อกับฐานข้อมูลสามารถกระทำได้โดยการกระทำผ่านทางตัวอิดิเตอร์ของ มายเอ็สคิวแอล (SQL editor) ซึ่งตัว NetBeans IDE ได้ติดตั้ง SQL Editor สำหรับจัดการกับ วัตถุประสงค์นี้ไว้เรียบร้อยแล้ว ด้วยการทำการคลิกขวาที่เมนูของโนดหลัก และโนดย่อยๆ ตามที่ ต้องการสั่งงาน ขณะที่ทำการติดต่อกับ MySQL server สามารถทำการสร้างฐานข้อมูลใหม่ที่ใช้ในการ งานได้ ดังตัวอย่างต่อไปนี้จะทำการสร้างฐานข้อมูลชื่อ MyNewDatabase โดย ทำการคลิกขวาที่โนด MySQL Server at locationhost: 3306 [root] และเลือกรายการเมนูย่อย Create Database ระบบจะมีกล่องข้อความ Create MySQL Database ให้ทำการกรอกชื่อฐานข้อมูลลงในช่อง New Database Name: ชื่อ *MyNewDatabase* แล้วทำการคลิก OK (ภาพที่ 10.1)



ภาพที่ 10.1 แสดงขั้นตอนในการสร้างฐานข้อมูลใหม่

เมื่อทำการสร้างฐานข้อมูล MyNewDatabase เสร็จสมบูรณ์ระบบจะทำการสร้างโนดย่อยต่อ จากโนดหลัก **MySQL Server at locationhost: 3306 [root]** ด้วยโนดชื่อ MyNewDatabase (ภาพที่ 10.2)



ภาพที่ 10.2 แสดงชื่อฐานข้อมูล mynewdatabase ที่ถูกสร้างขึ้น

จากนั้นให้ทำการสั่งให้ฐานข้อมูล mynewdatabase ได้ทำการเชื่อมต่อโดยผ่านทาง **Drivers** ด้วยการทำการคลิกขวาที่โนดชื่อ **mynewdatabase** ทำการเลือก **connect...** ซึ่งหลังจากทำการ เชื่อมต่อแล้วฐานข้อมูลจะปรากฏโนดย่อยชื่อ jdbc:mysql://localhost/mynewdatabase...เกิดขึ้น (ภาพที่ 10.3)



ภาพที่ 10.3 แสดงการเชื่อมต่อ mynewdatabase กับ Driver

สร้างตารางให้กับฐานข้อมูล (Creating Database Tables)

ขณะนี้ฐานข้อมูล mynewdatabase ได้ทำการเชื่อมต่อกับ MySQL Server แล้วลำดับถัดมา สามารถทำการสร้างตาราง แก้ไขตาราง ด้วย NetBeans IDE ซึ่งมีฟังก์ชัน (Function) ที่สนับสนุนแฟ้ม ชนิด SQL อย่างดียิ่ง ฐานข้อมูล mynewdatabase ยังเป็นฐานข้อมูลที่ว่างเปล่ายังไม่มีตารางในการใช้ เก็บข้อมูลใดๆ ในการสร้างตารางฐานข้อมูลสามารถทำได้สองวิธีคือ สร้างตารางได้จากกล่องข้อความ (Dialog) กับการสร้างตารางโดยใช้ตัว SQL Editor ซึ่งสามารถทำได้โดยตรงกับเมนูบน SQL Editor ดังนี้

การสร้างตารางด้วย SQL Editor (Using the SQL Editor)

การสร้างตารางจาก SQL Editor ของ NetBeans IDE มีขั้นตอนดังนี้

 จากตัว Explorer ของ Database เมื่อทำการขยายแล้วให้ทำการเชื่อมต่อโนด ตัว Explorer ของDatabase จะมีโฟลเดอร์ย่อยต่อจาก jbdc:mysql://localhost:3306/mynewdatabase... จำนวน 3 โฟลเดอร์ย่อย Tables Views และ Procedures (ภาพที่ 10.4)



ภาพที่ 10.4 แสดง Explorer ของฐานข้อมูล mynewdatabase

 2. ทำการคลิกขวาที่โฟลเดอร์ Tables แล้วทำการเลือก Execute Command ดังภาพที่ 10.5 จะได้พื้นที่ว่างของ main window ที่ใช้ในการสร้างโค๊ดของ SQL Editor (ภาพที่ 10.5)



ภาพที่ 10.5 แสดง Main Window ของ SQL Editor

 ในพื้นที่ของ SQL editor สามารถทำการสร้างตารางชื่อ Counselor โดยมีรายละเอียด ตารางดังนี้ (ทำการพิมพ์โค๊ดลงในส่วนของ Main Window) (ภาพที่ 10.6)

CREATE TABLE Counselor (

id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT,

firstName VARCHAR (50),

nickName VARCHAR (50),

lastName VARCHAR (50),

telephone VARCHAR (25),

email VARCHAR (50),

memberSince DATE DEFAULT '0000-00-00',

PRIMARY KEY (id)

);

หมายเหตุ รูปแบบคำสั่งของภาษา SQL ในการพิมพ์โค๊ดต้องให้ตรงตามข้อกำหนดของภาษา SQL ควร ศึกษาเพิ่มเติม



ภาพที่ 10.6 แสดงการสร้างตารางด้วยภาษา SQL

4. เมื่อพิมพ์โค๊ดตามภาพที่ 10.6 แล้วให้ทำการสั่งทำงาน (Run) ด้วยปุ่ม SQL Editor และ ด้านบนหรือสามารถการ Run ได้ด้วยการกดคีย์ Ctrl-Shift-E หรือทำการคลิกขวาใน SQL Editor และ เลือก Run ส่งผลการทำงานของ IDE สร้างตาราง Counselor ลงในฐานข้อมูล mynewdatabase สามารถตรวจสอบได้โดยการคลิกที่โฟลเดอร์ Tables ที่อยู่ใน Explorer ของ Database->mynewdatabase เพื่อทำการขยายโฟลเดอร์ย่อย (ภาพที่ 10.7)



ภาพที่ 10.7 แสดงการสั่ง Run และแสดงผลการ Run ของตาราง Counselor

การสร้างตารางด้วย Table Dialog (Using the Create Table Dialog)

ในการสร้างตารางด้วย SQL Editor ซึ่งเป็นวิธีหนึ่งในการสร้างตาราง ซึ่ง SQL Editor ยัง สามารถสร้างตารางได้ด้วยการทำงานของที่ละ Dialog ซึ่งในตัวอย่างนี้จะทำการสร้างตารางชื่อ Subject มีขั้นตอนดังนี้

 จากตัว Explorer ของ Database เมื่อทำการขยายแล้วให้ทำการเชื่อมต่อโนด ตัว Explorer ของ Database จะมีโฟลเดอร์ย่อยต่อจาก ibdc:mysql://localhost:3306/mynewdatabase... มีโฟลเดอร์ย่อย Tables ทำการคลิกขวาที่โฟลเดอร์ Tables แล้วทำการเลือก Create Table ตัวกล่อง ข้อความที่ใช้สร้างตาราง (Create Table Dialog) ถูกเปิดขึ้น ให้ทำการพิมพ์ชื่อตารางลงในช่อง Table Name: Subject และทำการคลิกปุ่ม Add column (ภาพที่ 10.8)

NetBeans IDE 8.0			
File Edit View Navigate Source Refactor Run Debug	Profile		
1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	v		
Projects Files Services %			
🖵 🗐 Databases			
MySQL Server at localhost: 3306 [root]	Create Table		×
	Table name Subject		Ŷ
phpmyadmin	Key Index Null Unique Column name	Data type Size	Add column
test			Edit
B- Drivers			Remove
jdbc:derby://localhost:1527/sample [app on APP]			Move Up
Jdbc:mysql://localhost:3306/mynewdatabase/zero	ite i i		Move Down
Tables Create Table			
Procec Recreate Table			
⊕-	avio		
Web Services Refresh			
Hudson Builders		OK Cancel	Help
🗄 🙀 Task Repositories			

ภาพที่ 10.8 แสดงขั้นตอนในการสร้างตารางด้วย Dialog

2. เมื่อได้ทำการกำหนดชื่อตารางเป็น Subject เพื่อทำการสร้างข้อมูลดังตารางที่ 10.1 ด้วย การพิมพ์ข้อมูลลงใน Dialog (ภาพที่ 10.9)

ตารางที่ 10.1 แสดงข้อมูลตาราง subject ของฐานข้อมูล mynewdatabase

Key	Index	Null	Unique	Column Name	Data Type	Size
checked]	[checked]		[checked]	id	SMALLINT	0
		[checked]		name	VARCHAR	50
		[checked]		description	VARCHAR	500
		[checked]		FK_counselorID	SMALLINT	0

able na	ame: Sub	ject						
Кеу	Index	Null	Unique	Column name		Data type	Size	Add column
	,							Edit
		C Ad	ld Columr	1		×		Remove
	- 1	Name	2:	id 🤇		П		Move Up
		Туре	:	SMALLINT		V		Move Down
		Size:	[Scale:			
•	-	Defa	ult:				P.	
1 Ado	d column	Co	nstraints					
	- 1		👿 श	mary key 📝 Unique	🔲 Null 🗸	Index		
			Charles [Cance	Help
L 2 ex	ecutior		Lneck:					
ed su col	ccessf umn 1		L					
ion f	inishe				Л			
						Cancel		

ภาพที่ 10.9 แสดงการเพิ่มคอลัมน์ให้กับตาราง Subject ของฐานข้อมูล mynewdatabase

 ทำการเพิ่มคอลัมน์ให้กับตาราง Subject ด้วยการคลิกปุ่ม Add column ทำการพิมพ์ข้อมูลดัง ตารางที่ 10.1 เป็นตัวกำหนดตารางโดยทำการสร้างให้ครบทั้ง 3 รายการ แสดงขั้นตอนในการสร้าง ตาราง (ภาพที่ 10.10)

lame:	name 🧲	C Add Column		×		
ype:	VARCHAR	Name: descrip	otion 🖕	Add Co	blumn	
ize:	50 Cale:	Type: VARCH	HAR 🧲	Name:	FK_counselor	3
)efault:		Size: 500	Cale:	Type:	SMALLINT	
Constraint	ts	Default:	•	Size:	4	Scale:
	Primary key 📄 Unique 💟 Null 📄 I	Constraints		Default:		
Check:		Primary k	key 📄 Unique 🛛 🕅 N	ull 🔄 In Constrai	ints	
		Check:			📄 Primary key 📄 Uniq	ue 🔽 Null 📄 Index
				Check	c	
	0 0					
	OK	1		0 I		
						—

ภาพที่ 10.10 แสดงการเพิ่มคอลัมน์ให้กับตาราง Subject ของฐานข้อมูล mynewdarabase

 เมื่อทำการสร้างตารางครบตามที่ได้ออกแบบข้อมูลในตารางที่ 10.1 แล้ว สามารถทำการตรวจ ข้อมูลได้ (ภาพที่ 10.11)

Projects	Files	Services %		-
🖃 🗐 Datal	oases			
📄 🖶 🍺 M	lySQL Serv	er at localhost:	3306 [ro	pot]
	informat	tion_schema		
	mynewd	latabase		
	mysql			
	🛛 phpmya	dmin		
	test			
1 🕀 📓 J	ava DB			
	rivers	// // // // // // // // // // // // //		[100]
ju 🕂 🖓 ju	ibc:derby:	//localhost:152/	//sample	e [app on APP]
je 💾 📜 je	ibc:mysql:/	//localhost:3306	/mynev	vdatabase?zeroDateTime
	mynew	database		
		es		
	÷	subject	_	
		id id		
		name		
		description		
		FK counsel	or	
	÷	Indexes		
	÷.	 Foreign Key	s	
t i i i i i i i i i i i i i i i i i i i	View	vs ,		
	-Proc	edures		
±	🛭 Other d	atabases		· · · · · · · · · · · · · · · · · · ·

ภาพที่ 10.11 แสดงผลการสร้างตาราง Subject ของฐานข้อมูล mynewdatabase

การสั่งงานตารางข้อมูล (Working with Table Data)

ในการสั่งให้การทำงานร่วมกันระหว่างตารางข้อมูลบน SQL Editor ของ NetBeans สามารถ ให้ทำงานในรูปแบบของ SQL queries บนฐานข้อมูลได้หลากหลาย การปรับปรุงแก้ไขโครงสร้างของ ฐานข้อมูล การเพิ่ม(add) การแก้ไข(modify) และการลบ(delete) รวมทั้งเพิ่มรายการข้อมูล (record หรือ row) ลงในตารางขอมูล Counselor สามารถทำได้ดังนี้

 ทำการเลือกตารางข้อมูลที่ต้องการสั่งงาน ในที่นี้ต้องการติดต่อกับตาราง Counselor เพื่อ ต้องการเพิ่มข้อมูลให้กับตารางที่ได้สร้างไว้แล้ว โดยคลิกขวาที่โฟลเดอร์ Tables->counselor แล้วเลือก Execute Command ตัว SQL Editor จะทำการเปิด main window เป็นพื้นที่ให้ทำการเขียนคำสั่งที่ ละบรรทัดได้ เป็นโค๊ดที่การต้องการเพิ่มรายการข้อมูล ได้ด้วยคำสั่งด้านล่าง (ภาพที่ 10.12)

INSERT INTO Counselor

VALUES (1, 'Ricky', "'The Dragon"', 'Steamboat','334 612-5678', 'r_steamboat@ifpwafcad .com', '2014-07-01')



ภาพที่ 10.12 แสดงขั้นตอนการสั่งเพิ่มรายการผ่าน Command กับตารางข้อมูล Counselor

2. เมื่อทำการเขียนโค๊ดด้วยภาษา SQL บน SQL Editor เรียบร้อยแล้ว สามารถทำการสั่ง Run ได้หลายวิธีตามได้กล่าวมาแล้วข้างต้น ในที่ให้ทำการกด Ctrl-Shift-E ผลการประมวลผลโค๊ดจะทำการ แสดงที่ Output window พบคำว่า successfully executed (ภาพที่ 10.13)



ภาพที่ 10.13 แสดงการสั่ง Run บน SQL Editor ของ NetBeans

3. หลังจากทำการสั่ง Run โค๊ดภาษา SQL แล้ว สามารถตรวจสอบผลการเพิ่มข้อมูลลงใน ตาราง Counselor ได้ด้วยการคลิกขวาที่โนดโฟลเดอร์ Counselor แล้วเลือก View Data ในส่วนของ main widow จะแสดงคำสั่งภาษา SQL และผลการทำงานจะแสดงในตาราง (ภาพที่ 10.14)

NetBeans IDE 8.0		_			_
File Edit View Navigate Source Refactor Run Debug Profile	Team Tools V	/indow Help			Q Search (
🕆 🎦 🔛 🖫 🏓 🥙 🤇 💷 🕞	TWP	• 🐘 • 🚯 •			
Projects Files Services #	GRAND SQL 1 [jdbc:r	nysql://localhost:33] 🛛 🗃 SQL 2 [jdbc:mj	ysql://localhost:33] 🛛		
🗐 🗐 Databases	Connection: jdb	c:mysql://localhost:3306/mynewdatabase?zero	DateTimeBehavior=con 👻 📑 📑	8 🚱 🖻 📴 🐻 • 🐻 • 💐 🖓 🖓	🖶 📭 🔗 🌜 🖣
MySQL Server at localhost: 3306 [root]	1 sele	ct * from counselor;	A		
	2		U U		
mynewdatabase	3				
mysql					
B tect	select * from co	unselor 18			
ava DB		📕 🔽 🚓 🛛 🖌 🖌 🔪 刘 Page Size:	20 Total Rows: 1 Page: 1 of 1		Matching
Drivers				1	
idbc:derby://localhost:1527/sample [app on APP]	# id	firstName	nickName	lastName	telephone
jdbc:mysql://localhost: 3306/mynewdatabase?zeroDateTime	1	1 Ricky	"The Dragon"	Steamboat	334 612-5678
🖶 🗐 mynewdatabase					
Tables					
subject View Data					
id Execute Command	·				
descr Add Column					
FK_ct Refresh					
Delete Delete					
I Fores					
Grab Structure					
Recreate Table	<pre> <</pre>		m	$ \rightarrow $	

ภาพที่ 10.14 แสดงการเพิ่มรายการด้วยภาษา SQL ผลการทำงานในตาราง Counselor

การสั่งการงานฐานข้อมูลด้วย Script (Running an SQL Script)

ในการจัดการเรื่องตารางข้อมูลของ NetBeans ยังสามารถสั่งให้ทำงานได้ด้วย SQL script โดยตรงได้ด้วย ดังตัวอย่างแฟ้มข้อมูลด้านล่าง เป็น script file ชื่อ ifpwafcad.sql หรือดาวน์โหลดได้ที่ https://netbeans.org/project_downloads/samples/Samples/Java%20Web/ifpwafcad.sql แล้วทำการบันทึกไว้ในตำแหน่งเดียวกับโปแกรม NetBeans IDE อยู่ ในที่เป็น C:\Program Files (x86)\NetBeans 8.0\bin สคริปต์จะทำการสร้างตาราง 2 ตารางที่คล้ายกับการสร้างตาราง Counselor และ Subject โดยสคริปต์ได้ทำการเปลี่ยนชื่อตารางเป็น Counselor2 และ Subject2 ตามลำดับ ในการสั่งสคริปต์ทำงานสามารถดำเนินการได้ดังนี้

 ทำการเปิดแฟ้มข้อมูลสคริปต์ด้วยการคลิกที่เมนูหลักFile>Open File แล้วเลือก ifpwafcad.sql ตามตำแหน่งที่ได้ทำการเก็บไว้ สคริปต์จะถูกทำการเปิดแสดงไว้ในส่วนของ SQL Editor (ภาพที่ 10.15) ด้านล่างนี้เป็นสคริปต์ที่ทำการบันทึกไว้ในแฟ้มข้อมูล ifpwafcad.sql

```
DROP TABLE IF EXISTS Counselor2;
DROP TABLE IF EXISTS Subject2;
```

```
CREATE TABLE Counselor2 (
```

counselor_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT, first_name VARCHAR (50), nick_name VARCHAR (50), last_name VARCHAR (50), telephone VARCHAR (25), email VARCHAR (50), member_since DATE DEFAULT '0000-00-00', PRIMARY KEY (counselor_id)

);

```
CREATE TABLE Subject2 (
```

```
subject_id SMALLINT UNSIGNED NOT NULL AUTO_INCREMENT, name VARCHAR (50),
```

description TEXT,

counselor idfk SMALLINT UNSIGNED,

PRIMARY KEY (subject_id)

);

INSERT INTO Counselor2 (first_name, nick_name, last_name, telephone, email, member_since)

VALUES ('Jake', "The Snake"', 'Roberts', '412 565-5656', 'snake@ifpwafcad.com', '2003-09-10'),

('Andre', "'The Giant"', ", '606 443-4567', 'bobo@ifpwafcad.com', '2001-01-12'),

('Brutus', ""The Barber"', 'Beefcake', '555 555-4432', 'beefcake@ifpwafcad.com', '2005-03-08'),

('Randy', "'Macho Man"', 'Savage', '555 317-4444', 'machoman@hotmail.com', '2000-11-10'),

('Ricky', '"The Dragon"', 'Steamboat','334 612-5678', 'r_steamboat@ifpwafcad.com', '1996-01-01'),

('George', "'The Animal"', 'Steele', '412 565-5656', 'george@ifpwafcad.com', Now()),

('Koko', "'The Birdman"', 'B. Ware', '553 499-8162', 'birdman@ifpwafcad.com', '1999-12-03'),

('Greg', "'The Hammer'', 'Valentine', '617 889-5545', 'valentino@ifpwafcad.com', '1998-05-07'),

('Bobby', "'The Brain"', 'Heenan', '777 513-3333', 'b_heenan@ifpwafcad.com', '2002-07-09');

INSERT INTO Subject2 (name, description, counselor_idfk)

VALUES ('Financial Consultancy', 'Investment advice and financial planning guidance, helping you to maximize your net worth through proper asset allocation. This includes the stocks, bonds, mutual funds, insurance products, and gambling strategies proven to work.', '9'),

('Existential Psychotherapy', 'Often wonder what the purpose of life is? After learning the basics of Existential Psychotherapy, you"ll understand why you"re happy when you"re feeling happy, and why you"re not feeling happy when you"re not happy, allowing you to transcend to a state of pure bliss.', '7'),

('Temper Management', 'Are your angry outbursts affecting your relationships with loved-ones? Do tantrums at work hinder your ability to perform? Temper management helps you to channel your anger into positive, life-changing productivity.', '4'),

('Past-Life Regression', 'Past-Life Regression is a journey of the soul, backward and forward through time, like a yo-yo.', '2'),

('Marriage Guidance', 'Even if you share a solid, caring and mutually beneficial relationship with your spouse, you may both still need urgent counseling. There's only one way to find out. Contact us now!', '1'), ('Crisis Management', 'Whether you"re a fireman, executive CEO, or housewife, applying crisis management techniques at the right moment can be lifesaving for you as well as all those around you.', '3'),

('Dream Analysis', 'Dream Analysis will allow you to delve into the depths of your subconcious. Your counselor will put you through a rigorous, disciplined training program, allowing you to remain in a waking state while dreaming. By the end, you''ll be able to analyse your dreams while you are having them!', '8'),

('Hypnosis', 'Contrary to popular belief, hypnosis can be a powerful and effective form of counseling.', '6'),

('Reiki', 'Need a massage but are afraid to let a stranger touch your body? Reiki could be the perfect solution for you.', '5');



ภาพที่ 10.15 แสดงขั้นตอนการเปิด script file : ifpwafcad.sql

 2. ทำการสั่ง Run โดยต้องทำการเลือกการสร้างตารางชื่อ Counselor2 และตาราง Subject2 ให้ทำการเชื่อมต่อกับฐานข้อมูลใหม่หรือฐานข้อมูลเดิมที่มีอยู่ในที่นี้เลือกฐานข้อมูล mynewdatabase (ภาพที่ 10.16)



ภาพที่ 10.16 แสดงการนำ script file ตารางมาเชื่อมต่อกับ mynewdatabase

3. สามารถตรวจสอบความถูกต้องได้โดยการ Explorer->Databases->mynewdatabase (ภาพที่ 10.17)



ภาพที่ 10.17 แสดงการเชื่อมต่อตาราง counselor2 กับ subject2 กับฐานข้อมูล mynewdatabase

4. สามารถตรวจสอบการบันทึกรายการของทั้งสองตาราง จากการกำหนดข้อมูลที่เขียนโดย สคริปต์มีการปรับปรุงข้อมูล ได้ด้วยการคลิกขวาที่โฟลเดอร์ชื่อตารางที่ต้องการตรวจสอบ และเลือก View Data ในส่วนของ SQL Editor จะทำการแสดงคำสั่งภาษา SQL ส่วนด้านล่างจะทำการแสดงค่า ข้อมูลที่ได้ทำการเพิ่มในตาราง ภาพที่ 10.18 แสดงข้อมูลตาราง Counselor2 และภาพที่ 10.19 แสดง ข้อมูลตาราง Subject2

e Edit View Navigate Source Refactor Run Debug Profile	Team Tools Window Help		Q . Search (Ctrl+I)
e car ner nangte source neretter nan besig riske			
2 🔁 🔛 🦏 🤊 (° 🗌 👘 🚽	T ₩ ▶·B·®·		
rojects Files Services 8	📑 ifpwafcad.sq/ 🖩 📄 SQL 1 [jdbc:mysql://localhost:33] 🛤		
Projects Files Services N Image: Database Image: Database	arnetion: jdx:mysd;/jdx:dhud:3305jmynewdalabas?feruDalcTmeEehaviar=can 💌 🛱 🕃 🐺	0 <u>- 9 9 - 1</u> - 1 <u>7</u> 7 5	9 8 4 9 8 8 9
mynewdatabase	select * from counselor 2 #		Matching Rows:
mynewdatabase Tables courselor formerical courselor formerical courselor formerical View Data	select * from counselor2 x Big III III III III III III IIII IIII II	last name	Matching Rows:
mynewdatabase Table roursel vorusel vorusel vorusel vorusel roursel roursel	select * from counselor 2 m select * from co	last_name Pohane	Matching Rows: telephone
mynewdatabase Tobes conneder conneder conneder conneder for conneder for conneder for conneder for conneder for conneder conneder	select" from ocuraelor 2 x Bill IIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIIII	last_name Roberts	Matching Rows telephone 412 565-5656 one 443-4547
mynewdatabase Tabes Tabes monowin counselor monowin monowin counselor monowin	select " from counselor 2 m counselor_d frst_name nok_name 1 1.3ke "The Snake" 2 2 Andre The Gant" 1 8 no m m	last_name Roberts Reefcalue	Matching Rows: telephone 412 565-5656 606 443-4587 555 555-432
mynewdatabase Tables Tables Tobs Tob	select." from counselor 2 w select." from counselor 2 w select." from counselor 3 first_name 1 1.Jake ntc_name 2 2.Andre The Gant" 3 3.Bhutus The Bother" 4 9.Radv Nacho Man"	last_name Roberts Beefcake Savoze	Matching Rows telephone 412 565-5656 606 443-557 555 555 - 4432 555 317-4442
mynewdatabase mynewdatabase moredatabase	select "from counselor2 # Bit in the select "from counselor2 # counselor_id first_name nick_name 1 alse "The Sinke" 2 2 Andre "The Sinke" 3 3 Shutus "The Select" 4 9 Randy 'Nacho Man' 5 5 8 Ricky "The Decor"	last_name Roberts Beefcake Savoge Staveboalt	Matching Rows: telephone 412 565-5656 606 443-4557 555 555 -4432 555 555 -4432 555 312 -4444 334 612 -557
mynewdatabase Takes Totos Tot	select " from counselor 2 m a counselor_d first_name not_name 1 1 1 1 Ale "The Sinke" 2 2 Andre "The Sinke" 3 3 Butus "The Sonke" 4 4 Randy "The Chapm" 5 5 RNdy "The Diagon" 6 6 6 George "The Arma"	last_name Roberts Beefolke Savage Steamboat Skenke	Matching Rows: telephone 412 955-9556 606 443-4567 555 555 4432 555 512 - 555 317-4444 334 612-5678 412 555 5563
mynewdatabase mynewdatabase consider con	select "from counselor 2 m Bit I all C Andre I Page Stet: 20 I Total Rows: 9 Page: 1 of 1 I a counselor d first_name nck_name 1 Jake "The Gast" 2 JAndre The Gast" 3 Bhutus The Barber" 4 Randy TheSham" 5 SRicky The Dragon" 6 6 6 corege The Areand" 7 7 Xiolo The Areand"	last_name Roberts Beefcake Stavoge Stamboat Skecke B. Ware	Matching Rows: telephone 412 565-5656 606 443-4867 555 555 443-4867 555 555 443-2 555 3127-4444 334 612-5678 412 565 555 553 949 6162
mynewdatabase Tobes Tob	select " from counselor 2 #	last_name Roberts Benfrake Swoge Steamboat Steele B. Ware Valenne	Matching Rows: telephone 412 505-5656 606 443-865 555 55-4422 555 327-484 334 622-6678 412 565 5666 553 499 842 617 999-545

ภาพที่ 10.18 แสดงข้อมูลตาราง Counselor2

Projects Files Services %	🗉 🧃 ifpwafcad.sql 🛛 🖬 SQL 1	[jdbc:mysql://localhost:33] 🛛 🗃 SQL 2 [jdbc:m	ysql://localhost:33] 18
Databases MySQL Server at localhost: 3306 [root] MySQL Server at localhost: 3306 [root] Drives Drives Jocrosoft Server at localhost: 1527[sample [app on APP] Si jdoc:ndorby://localhost: 3306/mynewdatabase?zerof Drives mynewdatabase Tables	Connection: jdbc:mssd://ocalhos 1 select * from s ateT	t:3305/mynewdatabase?zeroDateTmeBehavior =co ubject2;	<u>····</u> 8 ⊡ ⊋ © © 0 ⊫ · ■ ·
counselor counselor 2 counselor_id	select * from subject2 🗱	🕻 🔦 🔰 🗉 Page Size: 20 🗆 Total Ro	ws: 9 Page: 1 of 1
inick_name	# subject_id	name	description
	1 2 ≡ 3	1 Financial Consultancy 2 Existential Psychotherapy 3 Temper Management	Investment advice and financial planning guidance, helpin Often wonder what the purpose of life is? After learning t Are your angry outbursts affecting your relationships wit
member_since	4	4 Past-Life Regression	Past-Life Regression is a journey of the soul, backward a
	6	6 Crisis Management	Whether you're a fireman, executive CEO, or housewife,
Bubject	8	8 Hypnosis	Contrary to popular belief, hypnosis can be a powerful an
View Data V	9	9 Reiki	Need a massage but are afraid to let a stranger touch yo
Execute Command	Output %		
Add Column	Java DB Database Process 🕺 i	fpwafcad.sql execution 🕺 SQL 1 execution 🕺	SQL 2 execution 18
Kerresn Delete Del	Executed successfully Line 1, column 1	in 0.002 s.	

ภาพที่ 10.19 แสดงข้อมูลตาราง Subject2

แนะนำการโปรแกรมบนเครือข่ายด้วยจาวา

ในการศึกษาที่ผ่านมา ได้กล่าวถึงการจำลองเครือข่ายด้วยชุดโปรแกรม AppServ ที่มี Apache เป็นตัวจำลองการทำงานแม่ข่ายคอมพิวเตอร์ เป็นโลคอลโฮส (Localhost) หรือที่ URL Address เป็น 127.0.0.1 แล้วทำการเขียนโค๊ดที่ทำหน้าที่เป็นเครื่องแม่ข่าย (Server) ซึ่งทำการบันทึกโค๊ดต่างๆ ไว้ที่ C:\AppServ\www ดังนั้นจึงเป็นการจำลองสถานการณ์ว่าเครื่องคอมพิวเตอร์ที่ใช้เป็นเครื่องแม่ข่าย และในขณะที่ทำการเรียกข้อมูล ด้วยบราวเซอร์ ต่างๆ (Internet Explorer, Google Chrome) เครื่อง คอมพิวเตอร์ทำหน้าที่เป็นเครื่องลูกข่าย(Client) ในเวลาเดียวกัน

ดังนั้นถ้าการนำโค็ดที่ได้ทำการบันทึกไว้ที่ C:\AppServ\www ไปทำการเก็บไว้บนเครือข่าย คอมพิวเตอร์จริง หรือ IP Address จริง เป็นการทำงานจริงของการเขียนโปแกรมบนเครือข่าย คอมพิวเตอร์ได้ หรือศึกษาเพิ่มเติมได้ท<u>ี่https://netbeans.org/kb/docs/web/mysql-webapp.html</u> ซึ่งเป็นการสร้างเว็บไซต์บน MySQL database server ซึ่งครอบคลุมความคิดพื้นฐานเทคโนโลยีการ พัฒนาเว็บ JavaServer Pages (JSP), JavaServer Pages Standard Tag Library (JSTL), the Java Database Connectivity (JDBC) API และ สถาปัตยกรรม client-server เป็นต้น แนะนำการ ออกแบบสำหรับผู้เริ่มต้นมีความเข้าใจการพัฒนาเว็บไซต์ รวมทั้งมีความรู้ในการใช้ MySQL database

MySQL เป็นฐานข้อมูล ที่ใช้ในการจัดการกับข้อมูลที่ใช้ในการประยุกต์ใช้บนเว็บ ด้วยอาจเป็น เพราะ MySQL มีความโดดเด่น ด้านความเร็ว (speed) ด้านการยืดหยุ่น (flexibility) และด้านความ เป็นธรรมชาติ (reliability) MySQL ใช้ภาษา SQL ในการเขียนซึ่งมีโครงสร้างของภาษาเป็น Query Language สำหรับการเข้าถึง และการประมวลผลข้อมูลที่อยู่ในฐานข้อมูล จึงทำให้ MySQL เป็นระบบ มีความนิยมมากที่สุดในปัจจุบัน

สรุป

ในการทำการสร้างฐานข้อมูลด้วยจาวา ต้องทำการสร้างและทำการเชื่อมต่อฐานข้อมูลให้พร้อม ใช้งานก่อน ด้วยการกระทำผ่านทาง NetBeans IDE ที่ได้ติดตั้ง SQL Editor สามารถทำการสร้าง ฐานข้อมูลใหม่ที่ใช้ในการงานได้ **สร้างตารางให้กับฐานข้อมูล** สามารถทำได้สองวิธีคือ สร้างตารางได้ จากกล่องข้อความ (Dialog) กับการสร้างตารางโดยใช้ตัว SQL Editor **การสั่งงานตารางข้อมูล** ในการ สั่งให้การทำงานร่วมกันระหว่างตารางข้อมูลบน SQL Editor ของ NetBeans สามารถให้ทำงานใน รูปแบบของ SQL queries บนฐานข้อมูลได้ **การสั่งการงานฐานข้อมูลด้วย Script** ในการจัดการเรื่อง ตารางข้อมูลของ NetBeans สามารถสั่งให้ทำงานได้โดยตรง แล้วทำการบันทึกไว้ในตำแหน่งเดียวกับ โปแกรม NetBeans IDE สคริปต์จะทำการสร้างตาราง 2 ตารางที่คล้ายกับการสร้างตาราง Counselor และ Subject โดยสคริปต์ได้ทำการเปลี่ยนชื่อตารางเป็น Counselor2 และ Subject2 ตามลำดับ

แบบฝึกหัด

กำหนดให้ระบบเครื่องคอมพิวเตอร์ควรมีซอฟต์แวร์ดังนี้

- O NetBeans IDE เวอร์ชั่น 7.2, 7.3, 7.4, 8.0
- O Java Development Kit (JDK) เวอร์ชั่น 7 หรือ 8
- MySQL database server เวอร์ชั่น 5.x

ให้นักศึกษาสร้างโครงงานที่สามารถทำการติดต่อกับฐานข้อมูลอย่างน้อย 1 ฐานข้อมูล ประกอบด้วยตารางข้อมูลอย่างน้อย 2 ฐานข้อมูล โดยมีโครงสร้างการทำงาน (ภาพที่ 10.20) และมีหน้า แรกที่เป็นแฟ้มข้อมูล index.jsp (ภาพที่ 10.21)



ภาพที่ 10.20 แสดงโครงสร้างการติดต่อระหว่างเครื่องลูกข่ายกับเครื่องแม่ข่าย

Velcome to IFPWAFCAD,	
he Isteventional Ferrers	
ne international Former	
Professional Wrestlers'	
Acception for Counceling	
Association for Counseling	
and Development!	
IFPWAFCAD offers expert counseling in a wide range of fiel	ds.
To view the contact details of an IFPWAFCAD certified former	
To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:	
To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:	
To view the contact details of an IFPWAFCAD certified former professional wrestler in your area, select a subject below:	

ภาพที่ 10.21 แสดงหน้าแรกของโครงงาน

เอกสารอ้างอิง

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพา: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึ่งถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

- . (2545). Java Programming Volume II. กรุงเทพา: ซีเอ็ดยูเคชั่น บจก.
- สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถุ**. ค้นเมื่อ 23 สิงหาคม 2555, จาก

http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/ %BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf

- สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555, จาก <u>www.thaiglossary.org</u>
- Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Application Appserv network. ค้นเมื่อ Jun 27, 2012, จาก

http://www.appservnetwork.com/modules.php?name=Content&pa= showpage&pid=3

Armstrong, E. (2004). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Create Database. ค้นเมื่อ Jun 28, 2012, จาก

http://www.java2s.com/Code/Java/Database-SQL-JDBC/ CreatingaDatabaseinMySQL.htm

- MySQL. ค้นเมื่อ Jun 23, 2012, จาก http://www.mysql.com/why-mysql/white-papers/top-10-reasons-to-use-mysqlas-an-embedded-database/
- MySQL Database. ค้นเมื่อ Jun 28, 2012, จาก http://www.java2s.com/Code/Java/Database-SQL-JDBC/ <u>CreatingaDatabaseinMySQL.htm</u>
- MySQL Netbeans. ค้นเมื่อ Jun 28, 2012, จาก https://netbeans.org/kb/docs/ide/mysql.html
- Mysql Webapp. ค้นเมื่อ July 1, 2012, จาก <u>https://netbeans.org/kb/docs/web/mysql-webapp.html</u>

Project Netbeans. ค้นเมื่อ July 1, 2012, จาก <u>https://netbeans.org/project_downloads/samples/Samples/</u> <u>Java%20Web/ifpwafcad.sql</u>

Source Appserv. ค้นเมื่อ Jun 23, 2012, จาก <u>http://sourceforge.net/projects/appserv/files/</u>

บรรณานุกรม

- กิตติ ภักดีวัฒนะกุล, และศิริวรรณ อัมพรดนัย. (2544). Object-Oriented ฉบับพื้นฐาน. กรุงเทพา: สำนักพิมพ์ เคทีพี.
- รุ่งโรจน์ โพนคำ, และปราลี มณีรัตน์. (2545). Java Programming. กรุงเทพฯ: ซัคเซค มีเดีย บจก.
- วรรณิกา เนตรงาม. (2545). **คู่มือการเขียนโปรแกรมภาษาจาวา ฉบับผู้เริ่มต้น**. นนทบุรี: อินโฟเพรส บจก.

วีระศักดิ์ ซึงถาวร. (2543). Java Programming Volume I. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก. - . (2545). Java Programming Volume II. กรุงเทพฯ: ซีเอ็ดยูเคชั่น บจก.

สุดา เธียรมนตรี. (2556). **คู่มือเรียนเขียนโปรแกรมภาษา Java ฉบับสมบูรณ์**. นนทบุรี: ไอดีซี พรีเมียร์ บจก.

สุรางคนา ระวังยศ. (2555). **การเขียนโปรแกรมเชิงวัตถ**ุ. ค้นเมื่อ 23 สิงหาคม 2555, จาก <u>http://www.kmitl.ac.th/~s3010819/MyFile/My%20Ebook/JAVA/</u> <u>%BA%B7%B7%D5%E8%201%20%20%BA%B7%B9%D3.pdf</u>

สวทช. ค้นหาเมื่อ 2 มิถุนายน 2555, จาก <u>www.thaiglossary.org</u>

About the classes (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/java/javaOO/classes.html

About the java-Classes. (2012). Retrieved August 28, 2012, from:

http://pages.cs.wisc.edu/~hasti/cs368/JavaTutorial/NOTES/Java-Classes.html

About the java technology. (2012). Retrieved August 28, 2012, from:

http://docs.oracle.com/javase/tutorial/getStarted/intro/definition.html

Appserv. ค้นเมือ Jun 23, 2012, จาก

http://www.appservnetwork.com/modules.php?name=Content&pa =showpage&pid=3

Appserv files. ค้นเมื่อ Jun 23, 2012, จาก

http://sourceforge.net/projects/appserv/files/

Anban Pillay (2007). **Object Oriented Programming using Java**. University of KwaZulu-Natal.

Application Appserv network. ค้นเมื่อ Jun 27, 2012, จาก <u>http://www.appservnetwork.com/modules.php?name=Content&pa</u>= <u>showpage&pid=3</u>

Armstrong, E. (2004). **The J2EE™ 1.4 Tutorial**. Network Circle, CA: Sun Microsystem, Inc. Create Database. ค้นเมื่อ Jun 28, 2012, จาก

http://www.java2s.com/Code/Java/Database-SOL-JDBC/ CreatingaDatabaseinMySOL.htm

Java download (2012). Retrieved August 28, 2012, from:

http://java.sun.com/javase/downloads/index.jsp

บรรณานุกรม (ต่อ)

Java GUI. ค้นเมื่อ Jun 16, 2012, จาก
http://javacodeexam.blogspot.com/2014/02/java-gui-jpanel.html
Java Swing. ค้นเมื่อ Jun 16, 2012, จาก
http://en.wikipedia.org/wiki/Swing_(Java)
MySQL. ค้นเมื่อ Jun 23, 2012, จาก
http://www.mysql.com/why-mysql/white-papers/top-10-reasons-to-use-mysql-
as-an-embedded-database/
MySQL Database. ค้นเมื่อ Jun 28, 2012, จาก
http://www.java2s.com/Code/Java/Database-SQL-JDBC/
<u>CreatingaDatabaseinMySOL.htm</u>
MySQL Netbeans. ค้นเมื่อ Jun 28, 2012, จาก
https://netbeans.org/kb/docs/ide/mysql.html
Mysql Webapp. ค้นเมื่อ July 1, 2012, จาก
https://netbeans.org/kb/docs/web/mysql-webapp.html
NetBeans. ค้นเมื่อ Jun 23, 2012, จาก
https://netbeans.org/downloads/
NetBeans MySQL. ค้นเมื่อ Jun 28, 2012, จาก
https://netbeans.org/kb/docs/ide/mysql.html
Project Netbeans. ค้นเมื่อ July 1, 2012, จาก
https://netbeans.org/project_downloads/samples/Samples/
Java%20Web/ifpwafcad.sql
Source Appserv. ค้นเมื่อ Jun 23, 2012, จาก <u>http://sourceforge.net/projects/appserv/files/</u>
ผนวก การทดลอง

การทดลองที่ 1 หลักการเขียนโปรแกรมคอมพิวเตอร์

วัตถุประสงค์

- 1. เพื่อให้สามารถเข้าใจกระบวนการเขียนโปรแกรม
- 2. เพื่อให้สามารถเตรียมความพร้อมในการเขียนโปรแกรม
- 3. ศึกษาการทำงานของโปรแกรมภาษาจาวา

ความรู้พื้นฐาน

โปรแกรม Edit Plus ทำหน้าที่ในการสร้างซอร์สโค็ดของโปรแกรมจาวา โดยตัวซอร์สโค๊ด โปรแกรมต้องมีองค์ประกอบที่สำคัญ คือต้องมีประโยค public static void main(String args[){ } เป็นกำหนดที่เป็นเมธอดหลัก ซึ่งเมธอดหลักจะมีหน้าที่ในการทำงานอย่างไรบ้างขึ้นอยู่กับประโยคคำสั่ง ต่างๆ ที่อยู่ภายในเครื่องหมายปีกกาเปิด ({) และปีกกาปิด (}) และเมธอดหลัก public static void main(String args[){ } นี้ เป็นองค์ประกอบของคลาสหลัก ที่จะถูกสร้างขึ้นด้วยประโยคคำสั่ง public class FileNameClass { } โดยชื่อของคลาสหลัก (FileNameClass) ต้องเป็นชื่อเดียวกันกับแฟ้มข้อมูล ของซอร์สโค๊ดที่สร้างขึ้น โดยชื่อของคลาสหลักต้องเป็นซื่อของแฟ้มข้อมูลที่บันทึกด้วยชื่อ-นามสกุลคือ FileNameClass.java ด้วยเพราะเมื่อซอร์สโค๊ดถูกคอมไพล์ตัวคอมไพเลอร์จะทำการสร้างคลาสที่เป็น ไบท์โค๊ดขึ้นมา มาเป็นนามสกุล FileNameClass.class

การเขียนโปรแกรมจาวามีขั้นตอนในการพัฒนาคือ เมื่อทำการสร้างซอร์สโค็ดของโปรแกรมจา วาตามที่ได้กล่าวมาข้างต้นแล้ว ผู้พัฒนาโปรแกรมต้องทำการแปลภาษาจาวาด้วยตัวคำสั่งที่ใช้ในการ แปลภาษา javac.exe เป็นการสร้างแฟ้มข้อมูลที่เป็นคลาสมาหนึ่งแฟ้มข้อมูลที่มีนามสกุล .class เป็น ขั้นตอนที่สอง ซึ่งถ้ามองในมุมของระบบปฏิบัติการวินโดว์ (Window) แฟ้มข้อมูลที่ถูกสร้างขึ้นมา เป็นคลาสนั้น หมายความว่าในระบบระบบปฏิบัติการวินโดว์ (Window) แฟ้มข้อมูลที่ถูกสร้างขึ้นมา เป็นคลาสนั้น หมายความว่าในระบบระบบปฏิบัติการวินโดว์ จะทำการสร้างโฟลเดอร์ใหม่มาหนึ่ง โฟลเดอร์ (Folder) หรือหนึ่งไดเรกทอรี่ (Directory) ที่มีชื่อเดียวกันกับชื่อคลาส และหลังจากนั้นเป็นขั้น ที่สามเป็นการนำงานที่พัฒนาที่สร้างขึ้นมาสั่งให้ทำงานด้วยตัวคำสั่งที่ใช้ในการสั่งรันคือ java.exe ซึ่งใน การพัฒนางานด้วยโปรแกรมภาษาจาวาผู้พัฒนาสามารถทำงานได้ทั้งที่เป็นระบบดอส หรือเป็นระบบ ของวินโดว์ โดยใช้โปรแกรม Edit Plus เป็นเครื่องมือในการพัฒนาได้ทั้งสองระบบ แต่ด้วยโปรแกรม Edit Plus เป็นเครื่องมือที่มีความสะดวกต่อการสร้างและและการสั่งการจึงทำให้นักพัฒนานิยมใช้ตัว โปรแกรม Edit Plus เป็นเครื่องมือในการทำงาน

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติที่สามารถเข้าถึงระบบปฏิบัติการดอส (DOS Prompt)
- ระบบอินเทอร์เน็ต สำหรับดาวน์โหลด
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- ติดตั้งชุดพัฒนาโปรแกรมภาษาจาวา โปรแกรม Edit Plus ลงบนเครื่องคอมพิวเตอร์ของ ผู้เรียน
- 2. เรียกโปรแกรม Edit Plus มาใช้งาน
- เขียนโค๊ดโปรแกรมตามที่กำหนดด้านล่างลงบนโปรแกรม Edit Plus public class circle

```
{
    public static void main(String[] args)
    {
        int radius = 10;
        double pi = 3.14159;
        System.out.println("Circle Area");
        System.out.println("Radius = 10 ");
        System.out.println("Circle Area = "+ pi * radius * radius);
    }
}
```

 เมื่อพิมพ์โปรแกรมแล้วให้นักศึกษาทำการสั่งให้โปรแกรมภาษาจาวาทำการแปลภาษา โปรแกรมที่เขียนและตรวจความถูกต้องและทำการแก้ไขข้อผิดพลาดตามการแสดง Error โดยเลือกคำสั่งดังภาพด้านล่างแล้วสั่งทำการ แปลภาษา (Compiler) ที่เขียนและตรวจ ความถูกต้องและทำการแก้ไขข้อผิดพลาดตามการแสดง Error



5. สั่งทำการให้โปรแกรมทำงาน (Run) ได้ดังนี้



- 6. ทำการปรับปรุงโปรแกรมให้สามารถทำคำนวณพื้นที่สี่เหลี่ยมผืนผ้า
- 7. ให้ทำการพิมพ์โปรแกรมตามตัวอย่างทำการบันทึกชื่อแฟ้มข้อมูล FirstDialog.java

```
import javax.swing.JOptionPane;
public class FirstDialog
{
    public static void main(String[] args)
    {
        JOptionPane.showMessageDialog(null, "First Java dialog");
    }
}
8. ทำการแปลภาษาโปรแกรม (Compiler)
```

- 9. สั่งให้โปรแกรมทำงาน
- 10. เขียนผลการทำงานของโปรแกรมที่ได้

การทดลองที่ 2 หลักการเขียนโปรแกรมเชิงวัตถุ

วัตถุประสงค์

- 1. เพื่อให้สามารถเข้าใจกระบวนการเขียนโปรแกรมเชิงวัตถุ
- 2. เพื่อให้สามารถมีความพร้อมในการเขียนโปรแกรมเชิงวัตถุ
- 3. ศึกษาการทำงานของโปรแกรมภาษาจาวาตามรูปแบบคำสั่ง (Syntax)

ความรู้พื้นฐาน

การเขียนโปรแกรมจาวามีขั้นตอนในการพัฒนาด้วยการสร้างโปรแกรมจาวาแล้ว ผู้พัฒนาโปรแกรมต้องทำการแปลภาษาจาวา ที่การสร้างแฟ้มข้อมูลที่เป็นคลาสมาหนึ่งแฟ้มข้อมูล ขั้น ต่อมาเป็นการสั่งให้โปรแกรมทำงาน โดยใช้โปรแกรม Edit Plus เป็นเครื่องมือในการพัฒนา

้โดยการเขียนโปรแกรมทั้งทำการคำนึงการทำงานของภาษาจาเป็นการโปรแกรมเชิงวัตถุ ซึ่ง ้วัตถุ หรือ อ็อบเจกต์ ด้วยภาษาจาวามีสิ่งที่เป็นควรรู้พื้นฐานที่สำคัญในการเขียนโปรแกรมดังนี้ คลาส ้ คือเป็นการประกาศภาพรวมของกลุ่มวัตถุ **อินสแตนซ์** เป็นวัตถุที่ถูกสร้างขึ้นมาในหน่วยความจำด้วย ้ คำสั่ง new เป็นการอ้างถึงข้อมูลในหน่วยความจำที่เป็นวัตถุ ซึ่งทำงานคล้ายๆกับ พอยเตอร์ ในภาษาซี คืออ้างไปยังหน่วยความจำที่เก็บข้อมูลนั่นเอง **การปิดบังข้อมูล** เป็นวิธีการกำหนดสิทธิในการเข้าถึง ข้อมูล หรือกำหนดการกระทำต่อกับวัตถุของ คลาสนั้นๆ ทำให้แน่ใจได้ว่าข้อมูลของวัตถุนั้นจะถูก เปลี่ยนแปลงแก้ไขผ่านทางเมธอด หรือ คุณสมบัติ ที่ได้รับอนุญาตเท่านั้น การปิดบังข้อมูลนั้นเป็นการ ้ ป้องกันไม่ให้อ็อบเจกต์ภายนอกเข้าถึงข้อมูลได้อย่างอิสระ เป็นการสร้างความปลอดภัยให้กับข้อมูล **การ สืบทอดคุณสมบัติ** เป็นการสืบทอดคุณสมบัติจากคลาสแม่ไปยังคลาสลูก โดยที่คลาสลูกสามารถสืบทอด ้คุณสมบัติมาจากคลาสแม่ได้ ซึ่งเป็นคุณสมบัติของการนำอ็อบเจกต์ กลับมาใช้ใหม่ประหยัดเวลาในการ ้ทำงาน ซึ่งเป็นจุดเด่นสำคัญ ของการเขียนโปรแกรมเชิงวัตถุ **นามธรรม** เป็นการแสดงถึงคุณลักษณะ และพฤติกรรมของอ็อบเจกต์ เท่าที่จำเป็นต้องรับรู้และใช้งาน โดยซ่อนส่วนที่เหลือเอาไว้เพื่อไม่ให้เกิด ความสับสน และสามารถนำมาเพิ่มเติมคุณสมบัติเพิ่มเต็มภายหลังได้ **โพลิมอฟิซึม** เป็นการกำหนด หน้าที่การทำงานที่มีหลายรูปแบบ หรือกล่าวได้ว่า หนึ่งเมธอด สามารถทำงานได้หลายๆรูปแบบ โดยมี ชื่อเมธอดเดียวกัน แต่สามารถทำงานแตกต่างกันได้ ซึ่งโพลิมอฟิซึมสามารถแบ่งเป็น 2 แบบ คือ การโอ เวอร์โหลดดิ้ง (Overloading) และการโอเวอร์ไรดิ้ง (Overriding) เป็นต้น

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์ที่ใช้ระบบปฏิบัติที่สามารถเข้าถึงระบบปฏิบัติการดอส (DOS Prompt)
- 2. ระบบอินเทอร์เน็ต สำหรับดาวน์โหลด
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- 3. เขียนโปรแกรมตาม ตัวอย่าง 2.1 โปรแกรมในการกำหนดค่าคงที่
- เมื่อพิมพ์โปรแกรมแล้วให้นักศึกษาทำการสั่งให้โปรแกรมภาษาจาวาทำการแปลภาษา โปรแกรมที่เขียนและตรวจความถูกต้องและทำการแก้ไขข้อผิดพลาดตามการแสดง สั่งทำ การแปลภาษา ที่เขียนและตรวจความถูกต้องและทำการแก้ไขข้อผิดพลาดตามการแสดง Error
- สั่งทำการให้โปรแกรมทำงาน
- ให้นักศึกษาเขียนผลการทำงานโปรแกรม
- 7. ทำการปรับปรุงโปรแกรมตาม ตัวอย่าง 2.2 โปรแกรมในการแสดงค่าที่กำหนด
- 8. ทำการแปลภาษาโปรแกรม
- 9. สั่งให้โปรแกรมทำงาน
- 10. ให้นักศึกษาเขียนผลการทำงานโปรแกรม
- 11. เขียนโปรแกรม ตัวอย่าง 2.3 โปรแกรมในการกำหนดค่าในขอบเขต
- 12. ทำการแปลภาษาโปรแกรม และสั่งให้โปรแกรมทำงาน
- 13. ให้นักศึกษาเขียนผลการทำงานโปรแกรม
- 14. เขียนโปรแกรม ตัวอย่าง 2.4 การเขียนโปรแกรม ด้วยการจบคำสั่งของแต่ละบรรทัดด้วย คำสั่ง ; หรือเขียนโค๊ดในบรรทัดเดียวกันมากกว่าหนึ่งคำสั่ง
- 15. ทำการแปลภาษาโปรแกรม และสั่งให้โปรแกรมทำงาน
- 16. เขียนผลการทำงานของโปรแกรมที่ได้
- 17. ให้นักศึกษาเขียนผลการทำงานโปรแกรม พร้อมวิเคราะห์การทำงานของโปรแกรม
- 18. ให้นักศึกษาเขียนโปรแกรมที่ยังไม่สมบูรณ์ ตัวอย่าง 2.5
- 19. ให้นักศึกษาแก้ไข้ให้สมบูรณ์
- 20. ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน และเขียนผลการทำงานโปรแกรม

การทดลองที่ 3 ตัวแปรและตัวดำเนินการ

วัตถุประสงค์

- เพื่อให้สามารถเขียนโปรแกรมโดยการนำตัวแปรและตัวดำเนินการมาเป็นเครื่องมือใน การแก้ปัญหาโจทย์ที่กำหนดได้
- เพื่อให้สามารถมีความสามารถในการวิเคราะห์ตัวแปรและตัวดำเนินการบนตัว โปรแกรมจาวา
- เพื่อให้เกิดทักษะในการโปรแกรมภาษาจาวา ด้วยการประยุกต์ตัวแปรและตัว ดำเนินการมาประกอบเป็นงาน

ความรู้พื้นฐาน

ตัวแปร เป็นส่วนประกอบที่สำคัญอย่างมากในการโปรแกรมทุกโปรแกรมต้องมีการเรียกใช้ตัว แปรทุกครั้ง เพื่อให้เก็บข้อมูลได้เหมาะสมกับขนาดของค่าที่ต้องทำการเก็บ ตัวแปรเลขที่นำมาใช้งาน สามารถแบ่งได้ดังนี้ 1) ชนิด Boolean ขนาด 1 บิท 2) ชนิด byte ขนาด 8 บิท 3) ชนิด char ขนาด 16 บิท 4) ชนิด short ขนาด 16 บิท 5) ชนิด int ขนาด 8 บิท 6) ชนิด float ขนาด 32 บิท 7) ชนิด long ขนาด 64 บิท และ 8)ชนิด double ขนาด 64 บิท ดังนั้นนักศึกษาต้องพิจารณาในการกำหนด หรือประกาศตัวแปรว่าต้องนำมาใช้ในการเก็บข้อมูลเป็นชนิดใด เพื่อให้การประมวลผลของโปรแกรม ทำงานได้อย่างมีประสิทธิ หมายถึงโปรแกรมทำงานได้ถูกต้องแล้ว โปรแกรมต้องสามารถทำงานหรือ ประมวลผลได้อย่างรวดเร็วตามมาด้วย

ตัวดำเนินการ สามารถทำการแบ่งตามลักษณะการดำเนินการกับตัวถูกกระทำได้ 6 ลักษณะ ดังนี้ **1) ตัวดำเนินการการกำหนดค่าโดยใช้สัญญาลักษณ์** = (เครื่องหมายเท่ากับ) เป็นตัวดำเนินการ ในการทำหน้าที่ในการกำหนดค่า ต่างๆ **2) ตัวดำเนินการทางคณิตศาสตร์** ซึ่งมีส่วนย่อยดังนี้ 2.1) ตัว ดำเนินการทางคณิตศาสตร์เลขจำนวนเต็ม มีเครื่องหมาย + (บวก) เครื่องหมาย – (ลบ) เครื่องหมายคูณ (*) เครื่องหมายหาร (/) และเครื่องหมายหารคิดเศษ (modulus: %) 2.2) ตัวดำเนินการทาง คณิตศาสตร์เลขทศนิยม มีเครื่องหมาย + (บวก) เครื่องหมาย – (ลบ) เครื่องหมายคูณ (*) และเครื่องหมายหาร (/) และเครื่องหมาย + (บวก) เครื่องหมาย – (ลบ) เครื่องหมายคูณ (*) และเครื่องหมายหารคิดเศษ (modulus: %) 2.3) ตัวดำเนินการทางคณิตศาสตร์ร่วมกับการ กำหนดค่า เป็นตัวดำเนินการที่นำผลการกระทำของคณิตศาสตร์มาเก็บยังตัวแปรเดิมที่ถูกกระทำ เช่น a = a+1; 2.4) ตัวดำเนินการที่นำผลการกระทำของคณิตศาสตร์มาเก็บยังตัวแปรเดิมที่ถูกกระทำ เช่น a = a+1; สามารถเขียนลดรูปได้เป็น a++; **3) ตัวดำเนินการระดับบิท** ในการกระทำระดับบิทสามารถ กระทำข้อมูลของตัวแปรที่เป็นชนิดตัวเลขขนาดต่างๆ ได้แก่ byte, short, integer, long และ char ตัว ดำเนินการที่นำมาดำเนินการระดับบิท มีดังนี้ อินเวอร์เตอร์ แอนด์ ออร์ เอ็กคูชีพออร์ การเลื่อนขวา และเลื่อนซ้าย **4) ตัวดำเนินการความสัมพันธ์เชิงเปรียบเทียบ** ได้แก่ ตัวดำเนินการเท่ากัน == ตัว ดำเนินการไม่เท่ากัน != ตัวดำเนินการมากกว่า > ตัวดำเนินการน้อยกว่า > ตัวดำเนินการมากกว่าและ เท่ากับ >= ตัวดำเนินการน้อยกว่าและเท่ากับ >= เป็นต้น

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- เขียนโปรแกรมตาม ตัวอย่าง 3.1 โปรแกรมการกำหนดตัวแปร และการเปรียบเทียบตัว ดำเนินการหารกับโมดูลัส
- เมื่อพิมพ์โปรแกรมแล้้วให้นักศึกษาทำการสั่งให้โปรแกรมภาษาจาวาทำการ สั่งทำการ แปลภาษา และสั่งทำการให้โปรแกรมทำงาน
- ให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่า ทำงานอย่างไรโดยอ้างอิง ตัวแปร และตัวดำเนินการ
- 6. ทำการปรับปรุงโปรแกรมตาม ตัวอย่าง 3.2
- 7. ทำการแปลภาษาโปรแกรม และสั่งให้โปรแกรมทำงาน
- 8. ให้นักศึกษาเขียนผลการทำงานโปรแกรม
- ให้นักศึกษาทำการเขียนโปรแกรมทำการใช้ตัวดำเนินการออร์(OR) กระทำ 2 ตัวแปร แล้ว นำผลมาผ่านการตัวดำเนินการน็อต(NOT) โดยทำการดัดแปลงจากตัวอย่างโปรแกรมภาพ ที่ 3.17 ทำการแปลภาษาโปรแกรม และสั่งให้โปรแกรมทำงาน
- 10. ให้นักศึกษาเขียนผลการทำงานโปรแกรม
- ให้นักศึกษาทำการเขียนโปรแกรมทำการใช้ตัวดำเนินการเอ็กคลูซีพออร์(EXCLUSIVE OR) กระทำ 2 ตัวแปร โดยทำการดัดแปลงจากตัวอย่างโปรแกรมภาพที่ 3.18
- 12. ทำการแปลภาษาโปรแกรม และสั่งให้โปรแกรมทำงาน
- 13. เขียนผลการทำงานของโปรแกรมที่ได้
- ให้นักศึกษาทำการเขียนโปรแกรมทำการใช้ตัวดำเนินการความสัมพันธ์เชิงเปรียบเทียบ (Relational Operator) ตรวจสอบรหัสผ่าน (pass word)
- 15. ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน และเขียนผลการทำงานโปรแกรม

การทดลองที่ 4 ประโยคควบคุมการทำงาน

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมในการควบคุมการทำงานของงานที่กำหนดได้
- เพื่อให้สามารถมีความสามารถในการวิเคราะห์ สังเคราะห์โปรแกรมให้สามารถทำการ ควบคุมได้ทั้งประโยคการทำงานวนรอบ ประโยคการควบคุมการตัดสินใจ
- 3. เพื่อให้เกิดทักษะในการแก้ปัญหาโปรแกรมภาษาจาวาได้

ความรู้พื้นฐาน

ประโยค if-else

มีโครงสร้างของประโยค if-else ดังนี้

เมื่อ

<boolean expression="">คือ ประโยคเงื่อนไขที่กำหนดตรวจสอบถ้าถูกต้องให้ค่าบูลีนเป็น</boolean>				
	true	และให้ทำงานใน statements ในบล็อกถัดไป และถ้า		
	ตรวจสล	อบเงื่อนไขว่าไม่ถูกต้องค่าบูลีนเป็น false ให้ทำบล็อกถัดไป		
	หรือทำเ	ทำเนินการหลัง else ถ้ามี		
<statements></statements>	คือ บล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม			
	ซึ่งอาจเ	ป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง		
	คำสั่งต้อ	งเขียนภายใต้เครื่องหมาย { }		
[]	คือ เป็นเครื่องหมายที่แสดงว่าผู้เขียนโปรแกรม จะกำ			
	ในเครื่อ	งหมาย[] หรือไม่ก็ได้ เช่น [else <statements>;]</statements>		
	หมายถึง	ง ถ้ามีการกำหนด else เป็นการกำหนดบล็อกให้ดำเนินการ		
	งานเมื่อ	เงื่อนไขไม่ถูกต้อง		

ประโยค switch-case

	มีโครงสร้างของบ switch (<variat< th=""><th>lระโยค ดังนี้ ble >)</th></variat<>	lระโยค ดังนี้ ble >)			
	{				
	case <	value> : <statements>; break;</statements>			
	case <	case <value> : <statements>; break;</statements></value>			
	default	default : <statements>;</statements>			
เมื่อ	}				
	<variable></variable>	เป็น ค่า ที่ใช้กำหนดทางเลือก โดยการเทียบค่าของ variable ว่าตรง กับ value ที่ case ใด ตัวคอมไพเลอร์จะตัดสินใจเลือกงานนั้น ค่า ของ variable สามารถเป็นชนิดข้อมูล int, short, long, char, หรือ byte แต่ห้ามเป็น boolean, float, หรือ double			
	case <value></value>	เป็นตัวงานที่ถูกเลือกจาก ค่าของ variable ตรงกับ value ซึ่งค่าของ value เป็นตัวแยกแยะงาน ดังนั้นค่าของ value ห้ามมีค่าซ้ำกัน			
	<statements></statements>	เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง คำสั่งต้องเขียนภายใต้เครื่องหมาย { }			
	Default	เป็นบล็อกให้ดำเนินการก่อนออกจาก switch-case ในกรณีที่ค่าใน variable มีค่าไม่ตรงกับvalue ของ case ใดๆ			
	break	ทำหน้าที่ปิดงานของแต่ละ case ยกเว้นในส่วนของ default ไม่ต้อง ใช้ break			

ประโยค for

มีโครงสร้างประโยคดังนี้

for (<initial exp> ; <condition exp> ; <update exp>)

<statements>;

เมื่อ

- <initial exp> เป็น ค่าที่ใช้กำหนดค่าเริ่มต้นของรอบที่ต้องการให้ทำงานซ้ำ ซึ่ง สามารถกำหนดให้เป็นค่าผ่านตัวแปร หรือ กำหนดให้อยู่ในรูปของ สมการ สามารถกำหนดได้
- <condition exp>เป็น เงื่อนไขในการตรวจสอบจำนวนรอบว่าครบตามจำนวนที่ กำหนดหรือไม่ ซึ่งถ้าผลการทดสอบเป็น true โปรแกรมจะไป ดำเนินการใน statement
- <statements> เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง คำสั่งต้องเขียนภายใต้เครื่องหมาย { }

262

<update exp> จะถูกเรียกให้ประมวลผลต่อจากการทำงานของ statement และนำ ค่าตัวแปรเริ่มต้นที่ปรับปรุงค่า (update) แล้วไปทำการทดสอบ เงื่อนไขใน <condition exp> อีก และทำงานซ้ำเช่นนี้จนกว่าค่าใน <condition exp> มีค่าเป็น false โปรแกรมก็จะหลุดจากลูปไป ทำงาน ในประโยคที่ต่อจาก statement

ประโยค while (while statement)

มีโครงสร้างประโยคดังนี้ while (<boolean expression>) <statements>;

เมื่อ

<boolean expression> เป็น เงื่อนไขในการตรวจสอบว่าถูกต้องหรือไม่ ซึ่งถ้าผลการ ตรวจสอบเป็น true โปรแกรมจะไปดำเนินการใน statement แต่ ถ้าผลการตรวจสอบเป็น false โปรแกรมจะไปดำเนินการใน นอก บล็อกของ statements <statements> เป็นบล็อกที่กำหนดงานให้ดำเนินงานในการแก้ปัญหาของโปรแกรม ซึ่งอาจเป็นคำสั่งหรือเมธอดเดียว หรือเป็นกลุ่มคำสั่งมากกว่าหนึ่ง คำสั่งต้องเขียนภายใต้เครื่องหมาย { }

ประโยค do...while (do...while statement)

มีโครงสร้างประโยค do...while ดังนี้

do {

<statements>; } while (<boolean expression>);

เมื่อ

 do เป็นการสั่งให้ทำงานเริ่มต้นก่อนหนึ่งรอบ ตามบล็อก statement
 <statements> เป็นบล็อกที่ดำเนินงานของลูปในการทำงานซ้ำ เป็นกลุ่มคำสั่งในการ แก้ปัญหาของโจทย์ต้องเขียนภายใต้เครื่องหมาย { }
 while <boolean expression> เป็น เงื่อนไขในการตรวจสอบว่าถูกต้องหรือไม่ ซึ่ง ถ้าผลการตรวจสอบเป็น true โปรแกรมจะไปดำเนินการใน do <statement> ไป เรื่อยๆ จนตรวจสอบแล้วผลเป็น false โปรแกรมจะไปดำเนินการนอกบล็อกของ statements

ประโยคการควบคุมอื่นๆเพิ่มเติม (other control statements)

ประโยคในการควบคุมทิศทางกำหนดให้โปรแกรมทำงานของผู้เขียนโปรแกรม นอกเหนือจากวิธีในการกำหนดผ่านทางเงื่อนไข ผ่านทางการวนลูป เป็นหลักแล้วยังมีประโยคที่ น่าสนใจเพิ่มเติม ที่สามารถนำมาช่วยในการแก้ปัญหาในการเขียนโปรแกรมที่ไม่ปกติได้ เมื่อ ต้องการกระโดด (jump) ออกจากบล็อกของลูปการทำงานปกติ ไปยังจุดการทำงานจุดอื่นได้ ประโยคดังกล่าวประกอบด้วย ประกอบด้วย break, continue และ return

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- เขียนโปรแกรมตาม ตัวอย่าง 4.1 เป็นตัวอย่างโปรแกรมการกำหนดแสดงข้อความ GoodMorning เมื่อรับค่า 0-11 แสดงข้อความ GoodAfterNoon เมื่อรับค่า 12-18 และ แสดงข้อความ GoodEverning เมื่อรับค่า 19-24
- เมื่อพิมพ์โปรแกรมแล้วให้นักศึกษาทำการสั่งให้โปรแกรมภาษาจาวาทำการแปลภาษา สั่ง run และเขียนผลการทำงานของโปรแกรม ตั้งแต่เริ่มทำการรับค่าจากคีย์บอร์ด และผลการ ประมวลผลโปรแกรม
- เขียนโปรแกรมตาม ตัวอย่าง 4.2 โปรแกรมการใช้ for loop ให้นักศึกษาทำการวิเคราะห์ โปรแกรม ว่าสามารถนำไปใช้งานได้หรือไม่ ให้นักศึกษาทำการปรับปรุงโค๊ดให้สามารถ run ได้ ตามที่นักศึกษาต้องการ
- ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน และให้นักศึกษาเขียนผลการทำงาน โปรแกรม
- ให้นักศึกษาทำการเขียนโปรแกรมตัวอย่าง 4.3 โปรแกรมการใช้ do...while loop ให้ นักศึกษาทำการวิเคราะห์โปรแกรม ว่าสามารถนำไปใช้งานได้ด้านใด ให้นักศึกษาทำการ ปรับปรุงโค๊ดให้สามารถ run ได้ ตามที่นักศึกษาต้องการ
- จงเขียนโปรแกรม Java ให้รับค่าชื่อของผู้เขียนโปรแกรม ทำการพิมพ์ 5 ครั้ง เพื่อทำการ คำนวณ

การบวกจาก 1 ถึง 100 โดยที่บวกเฉพาะเลขคี่ (1, 3, 5, . . .)

การทดลองที่ 5 คลาสและอ็อบเจกต์

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมในงานด้วยการใช้โครงสร้างคลาส
- 2. เพื่อให้สามารถสร้างอ็อบเจกต์ประกอบการเขียนโปรแกรมได้
- เพื่อให้สามารถมีความสามารถในการวิเคราะห์แยกแยะการทำงานของแต่ละอ็บเจกต์ และ สังเคราะห์โปรแกรมให้สามารถนำคลาสที่สร้างขึ้น หรือคลาสจากไลบรารีมาใช้งานได้
- 4. เพื่อให้เกิดทักษะในการแก้ปัญหาโปรแกรมภาษาจาวาได้

ความรู้พื้นฐาน

การเขียนโปรแกรมเชิงวัตถุ จึงเป็นการเขียนโปรแกรมที่สามารถเข้าถึงในการแก้ปัญหาได้ โดยมี แนวคิดจากธรรมชาติมากำหนดเป็นหลักในการเขียนโปรแกรม มีองค์ประกอบสำคัญคือ คลาส แพ็กเกจ และ อ็อบเจกต์ มีรายละเอียดดังนี้

คลาส เป็นเปรียบเสมือนกับภาชนะที่รวบรวมวัตถุ(object) โดยในแต่ละวัตถุประกอบด้วย ตัว แปร และสองเมธอด ความหมายของ**คลาส** คือเป็นที่รวบรวมอ็อบเจกต์หนึ่งอ็อบเจกต์ หรือมากกว่า หนึ่งอ็อบเจกต์ หรือเรียกคลาสว่า เป็นพิมพ์เขียวในการสร้างอ็อบเจกต์ โดยข้อมูลที่อยู่ในคลาสในแต่ละ คลาส สามารถเรียก หรือนำมาใช้มาใช้งานภายนอกคลาสได้

แพ็กเกจ การเขียนโปรแกรมภาษาเชิงวัตถุ ด้วยภาษาจาวามีการเรียกใช้งานคลาสที่เป็น แพ็กเกจอย่างสม่ำเสมอ เพื่ออำนวยความสะดวกแก่ผู้พัฒนาโปรแกรมได้อย่างต่อเนื่อง แพ็กเกจที่เป็น ส่วนสำคัญสำหรับผู้เริ่มต้นในการพัฒนาโปรแกรมภาษาจาวา ควรรู้และนำมาใช้งานบางส่วนดังนี้

java.applet เป็นแพ็กเกจที่เก็บคลาสด้าน Applet ส่วน java.awtเป็นแพ็กเกจที่เก็บคลาส ด้าน GUI(Graphic User Interface) แยกจาก swing และ java.beans เป็นแพ็กเกจที่เก็บคลาส ด้าน beans เป็นต้น

อ็อบเจกต์ มีรูปแบบในการสร้าง ประกอบด้วย 3 แบบ คือที่

1. ประกาศ instance class ด้วย class_name Object_ identifier ;

โดยการเตรียมพื้นที่บนหน่วยความจำสำหรับ instance class ที่สร้างขึ้นด้วย

Object_identifier = new class_name ([class_parameter]);

- ประกาศ instance พร้อมเตรียมพื้นที่บนหน่วยความจำสำหรับ instance ที่สร้างใหม่ class name Object_ identifier = new class_name();
- สร้างอ็อบเจกต์โดยไม่ประกาศตัวแปรอ็อบเจกต์เพื่อใช้อ้างอิงอ็อบเจกต์ได้ new class_name();

ส่วนคอนสตรัคเตอร์ เป็นการสร้างอ็อบเจกต์พร้อมกับการกำหนดค่าเริ่มต้นให้กับแอทตริบิวต์ ต่าง ๆ ของอ็อบเจกต์ที่ได้สร้างขึ้นด้วยคีย์เวร์ด new โดยคอนสตรัคเตอร์จะถูกเรียกใช้ให้ทำงาน อัตโนมัติ เมื่อทำการสร้างออบเจ็คขึ้น ในขณะที่สร้างแต่ละอินสแตนซ์ขึ้นมาที่มีชื่อเดียวกันกับ คลาส ซึ่งสามารถกำหนดเป็น public ไว้หน้า modify ได้เท่านั้น

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- เขียนโปรแกรมตาม ตัวอย่าง 5.1 เป็นตัวอย่างโปรแกรมการสร้างคลาสลักษณะคอนสตรัค เตอร์ และการส่งค่าเมื่อมีการสร้างอ็อบเจกต์ ทำการบันทึกคลาสเป็น Lab5_1 ทำการ ทดสอบโปรแกรม
- เมื่อพิมพ์โปรแกรมแล้วให้นักศึกษาทำการสั่งให้โปรแกรมภาษาจาวาทำการแปลภาษา สั่ง run และเขียนผลการทำงานของโปรแกรม ตั้งแต่เริ่มทำการรับค่าจากคีย์บอร์ด และผลการ ประมวลผลโปรแกรม
- เขียนโปรแกรมตาม ตัวอย่าง 5.2 โปรแกรมการใช้คลาส ให้นักศึกษาทำการวิเคราะห์ โปรแกรม ว่าสามารถนำไปใช้งานได้หรือไม่ ให้นักศึกษาทำการปรับปรุงโค๊ดให้ได้ผลตามที่ นักศึกษาต้องการ
- ให้นักศึกษาทำการศึกษาแพ็กเกจของโปรแกรมภาษาจาวา เพิ่มเติมจาก ตัวอย่างโปรแกรม
 5.3
- ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน และให้นักศึกษาเขียนผลการทำงาน โปรแกรม
- ให้นักศึกษาทำการเขียนโปรแกรมตามตัวอย่างโปรแกรมที่ 5.4 ให้สามารถทำงานรับจาก ผู้ใช้งาน จำนวน 2 ค่า แล้วนำมาคำนวณ พร้อมแสดงผลการคำนวณด้วย Dialog ของ Packet

266

การทดลองที่ 6 การใช้งานแอพเพ็ต

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมจาวาด้วยเทคโนโลยีแอพเพ็ตได้
- 2. เพื่อให้สามารถนำโค๊ดแอพเพ็ตมาทำงานร่วมกับเว็บเพจได้
- 3. เพื่อให้สามารถมีความสามารถในการวิเคราะห์แยกแยะการทำงานของแต่ละเทรดได้
- 4. เพื่อให้เกิดทักษะในการแก้ปัญหาโปรแกรมภาษาจาวาได้

ความรู้พื้นฐาน

ในการสร้างโค๊ดจาวาแอพเพ็ตต้องการสร้างอ็อบเจกต์ที่ได้ทำการขยายมาจากคลาส ต้นแบบชื่อ java.applet.Applet ด้วยคีย์เวิร์ด extends ดังตัวอย่างการสร้างอ็อบเจกต์ ดังนี้ public class FirstApplet extends java.applet.Applet

เมื่อ FirstApplet.java ถูกคอมไพล์แล้วจะได้ FirstApplet.java.class จากนั้นจะต้องมี HTML page ที่มี APPLET tag ที่เรียก FirstApplet.java.class ให้ทำงานอยู่ใน ไฟล์ JavaApplet.html และทำการเรียก Applet มาใช้งานด้วย appletviewer (ภาพด้านล่าง)



ภาพแสดงการ APPLET tag อยู่ในไฟล์ JavaApplet.html

ในการทำงานร่วมกันระหว่างแท็กแอพเพ็ต กับ HTML สามารถทำการส่งพารามิเตอร์ที่อยู่ ภายใต้แท็ก <APPLET> กับ </APPLET> โดยมีรูปแบบ

```
<APPLET CODE = "AppletFile.class" width = 300 height =100>
```

```
PARAM NAME = "ชื่อของพารามิเตอร์ 1" VALUE = "ค่าข้อมูลของพารามิเตอร์ 1">
```

<PARAM NAME = "ชื่อของพารามิเตอร์ N" VALUE = "ค่าข้อมูลของพารามิเตอร์ N"> </APPLET>

โดยพารามิเตอร์ เป็นลักษณะ Case-Sensitive กล่าวคือ ถ้าเป็นชื่อเดียวกันแต่คนละลักษณะ ตัวพิมพ์ ถือว่าเป็นชื่อพารามิเตอร์ คนละตัวกันจึงสามารถใช้ ในการเก็บค่าข้อมูลคนพารามิเตอร์ละตัว

การสร้างกราฟิกแสดงผลสู่หน้าจอภาพ สามารถทำได้โดยใช้เมธอด ที่อยู่ในแพ็คเก็ต java.awt เช่น setColor() และ setFont() ส่วนเมธอดที่ทำการสร้างลักษณะรูปร่างต่างๆ ได้แก่เมธอด drawString(), drawLine(), drawRect(), fillRect(), drawRoundRect(), fillRoundRect(), drawOval(), fillOval(), drawArc(), fillArc(), drawPolygon(), fillPolygon() และเมธอด drawPolyline() ดังตัวอย่าง การกำหนดสีให้ไปใช้กับอ็อบเจกต์ที่ต้องการแสดงกราฟิก มีรูปแบบการ โค๊ด

setColor(Object สี);

การกำหนดเสียงของแอพเพ็ต ทำได้โดยการเรียกเมธอดสำหรับใช้งานการนำเสียงมาเก็บไว้ ที่ตัวแปรได้ด้วยเมธอด getAudioClip() และเมธอดสั่งการควบคุมเสียง เมธอดสั่งให้เปิดเสียง AudioClip.play() เมธอดสั่งวนรอบเสียง AudioClip.loop() และเมธอดปิดเสียง AudioClip.stop() เป็นต้น ซึ่งไฟล์เสียงที่นำมาใช้งานควรอยู่ที่โฟลเดอร์เดียวกับโค๊ดจาวา และชนิดของไฟล์เสียงที่สนับสนุน ได้แก่ .wav .au และ .midi เป็นต้น

การกำหนดภาพของแอพเพ็ตสามารถเรียกใช้งานได้อย่างสะดวก ได้ด้วยถ้าต้องการนำ ภาพมาเก็บไว้ที่ตัวแปรได้ด้วยเมธอด getImage() และเมธอดสั่งแสดงภาพได้ด้วยเมธอด g.drawImage("ไฟล์ภาพ", 0, 0, null) ซึ่งไฟล์ภาพที่นำมาใช้งานควรอยู่ที่โฟลเดอร์เดียวกับโค๊ดจาวา และชนิดของไฟล์ภาพที่แอพเพ็ตสนับสนุนได้แก่ .gif และ .jpg เป็นต้น

แอพเพ็ตกับเทรดสำหรับภาพเคลื่อนไหว ภาพเคลื่อนไหวสามารถสร้างขึ้นได้ โดยการนำ ภาพนิ่งหลายๆ ภาพมาฉายต่อเนื่องกันด้วยความเร็วสูง ทำให้เกิดภาพลวงตาของการเคลื่อนไหว ซึ่งใน การดำเนินการต้องอาศัยการทำงานของ เทรด ซึ่งเทรด ก็คือ ส่วนประกอบย่อยของการโปรเซส (ประมวลผล) ดังนั้น เทรด จึงเป็นงานย่อยๆ ของการประมวลผลในขบวนการทำงานของโปรเซสเซอร์ ถ้าในการทำงานของซีพียู 1 โปรเซส ที่มีการทำงานหลายๆ เทรด (งานย่อย) การสร้างเทรดในแพ็กเกจ ของแอพเพ็ต เป็นการเรียกเมธอด run() ซึ่งระบบก็ต้องมีส่วนประกอบการทำงานเทรดของเมธอด paint() ประกอบทุกครั้งอย่างน้อย 1 เทรด ตัวอย่าง การสร้างคลาสที่มีการอินเตอร์เฟส (interface) เท รดเป็นส่วนย่อยนั่นเอง เช่น public class ClassName implements Runnable {...} หรือสร้างคลาส ด้วย public class ClassName extends Thread {...} ในการสร้างภาพเคลื่อนไหว ก่อนทำการสร้าง ต้องเตรียมภาพที่ต้องการนำมาแสดง และพื้นหลังที่ต้องการให้ภาพนิ่ง จำนวนตามที่ต้องการสร้าง ภาพเคลื่อนไหวในที่นี้ต้องทำการเตรียมแฟ้มข้อมูลที่เป็นนามสกุล .gif มาร่วมใช้งานการสร้างเป็นต้น

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- เขียนโปรแกรมตามตัวอย่าง 6.1 โปรแกรมในการกำหนดสี สร้างรูปสี่เหลี่ยม และ เติมสีใน รูปเหลี่ยม ที่ในการใช้เมธอดการสร้างกราฟิกด้วยเมธอด setColor()และเมธอด drawRect() ซึ่งมีรูปแบบการโค๊ด คือ drawRect(x, y, w, h) โดย x, y เป็นจุดเริ่มต้นของ รูปสี่เหลี่ยม ส่วน w เป็น ความกว้างของรูปสี่เหลี่ยม และ h เป็นความสูงของรูปสี่เหลี่ยม และการเติมสีลงในรูปเหลี่ยม
- ให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่า ทำงานอย่างไรโดยนำผลการทดลองมาประกอบการอธิบาย

- 5. ให้นักศึกษาเขียนโปรแกรม ตามตัวอย่างที่ในการใช้เมธอดการสร้างกราฟิกด้วยเมธอด setColor()และเมธอด drawRoundRect(20,10,100,100,50,50) ซึ่งมีรูปแบบการโค๊ด คือ drawRoundRect(x, y, w1, h1, w2, h2) โดย x, y เป็นจุดเริ่มต้นของรูปสี่เหลี่ยม ส่วน w1 เป็นความกว้างของรูปสี่เหลี่ยม และ h1 เป็นความสูงของรูปสี่เหลี่ยม ส่วน w2 เป็นความกว้างของรูปสี่เหลี่ยมของที่ใช้กำหนดความโค้งของมุมสี่เหลี่ยม และ h2 เป็น ความสูงของรูปสี่เหลี่ยมของที่ใช้กำหนดความโค้งของมุมสี่เหลี่ยม แสดงภาพที่ 6.17 และ การเติมสีลงในรูปเหลี่ยมเมธอด fillRoundRect()
- 6. ให้นักศึกษาเขียนโปรแกรมตาม ตัวอย่าง 6.2 โปรแกรมในการแสดงค่าของเมธอด drawRoundRect() กับเมธอด fillRoundRect()
- 7. ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน ให้ผลการทำงานแสดงภาพที่ 6.18
- ให้นักศึกษาเขียนให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ใน แต่ละบรรทัดว่าทำงานอย่างไรโดยนำผลการทดลองมาประกอบการอธิบาย
- เขียนโปรแกรมตาม ตัวอย่าง ตัวอย่าง 6.3 โปรแกรมในการแสดงค่าของเมธอด drawOval() กับเมธอด fillOval()
- ให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่า ทำงานอย่างไร และให้เขียนผลการทำงานของโปรแกรม
- 11. ให้นักศึกษาทำการค้นคว้าในการเขียนโค๊ด โดยใช้ g .setColor(), drawRect(), fillRect(), fillOval() ให้สามารถ แสดงผลการทำงานโปรแกรม (ภาพที่ 6.19)
- 12. ให้นักศึกษาทำการค้นคว้าในการเขียนเมธอด drawArc(), fillArc(), drawPolygon(), fillPolygon() และเมธอด drawPolyline()
- 13. จงเขียนโค๊ดให้มีภาพเคลื่อนไหวของแอพเพ็ตอย่างน้อย 2 ภาพ และเขียนเหตุผลการแสดง ค่าของโค๊ดว่าได้ผลที่แสดงการทำงานเป็นเพาะเหตุใด
- 14. จากโค๊ดด้านล่างพร้อมเขียนเหตุผลการแสดงค่าของโค๊ดว่าได้ผลที่แสดงเพราะเหตุใด

```
import java.awt.*;
import java.applet.*;
public class ImageExApplet extends java.applet.Applet
{
    Image imgEx;
    public void init()
        {
        imgEx = getImage(getCodeBase(), "bravo.gif"); }
    public void paint(Graphics g)
        {
        g.drawImage(imgEx, 100, 100, null); }
}
```

การทดลองที่ 7 ส่วนติดต่อกับผู้ใช้แบบกราฟิก

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมจาวาสร้างส่วนติดต่อกับผู้ใช้แบบกราฟิกได้
- 2. เพื่อให้สามารถนำส่วนติดต่อกับผู้ใช้แบบกราฟิกมาทำงานร่วมกับเสียงและภาพได้
- เพื่อให้สามารถมีความสามารถในการวิเคราะห์แยกแยะการทำงานของส่วนติดต่อกับผู้ใช้ แบบกราฟิกได้
- 4. เพื่อให้เกิดทักษะในการแก้ปัญหาโปรแกรมภาษาจาวาได้

ความรู้พื้นฐาน

GUI มีส่วนประกอบหลักดังนี้

Containers เป็นส่วนที่ทำหน้าที่ในเก็บ Components ต่างๆ ที่ต้องแสดงหน้าเว็บ ตามที่ กำหนด ซึ่งอาจเป็นโครงร่างในการออกแบบการวาง layout (เช่น flow หรือ grid) และส่วนของ container สามารถเก็บส่วนที่เป็น sub-containers ได้การโปรแกรม GUI ใน Container จะต้อง มี frame, Dialog และ Applet (Top-Level Containers) ซึ่งในส่วนของ frame ได้ จัดเตรียมส่วนที่เป็น main window

Components เป็นส่วนประกอบเป็นส่วนที่แสดงข้อมูล บนหน้าจอเว็บ เช่น Button, Label, and TextField เป็นต้น ในการโปรแกรม GUI ทุก component ต้องถูกวางใน container โดยทุก component สามารถวางด้วยเมธอด add (Component c) มีผลทำให้ Container สามารถเข้ามาทำงานได้

การโปรแกรม GUI ด้วย Swing และ AWT

ขั้นตอนที่ 1 สร้าง JFrame ด้วย JFrame *fName* = new JFrame("*Title*");

ขั้นตอนที่ 2 สร้าง JPanel ซึ่งเป็น component และต้องนำ JPanel ไปวางบนในส่วนของ container ในที่นี้

> ก็คือ JFrame ซึ่งมีรูปแบบของการสร้าง JPanel คือ JPanel pName = new JPanel(manager);

ขั้นตอนที่ 3 สร้างคลาสที่เป็น Component 3.1 ส่วนที่เป็น ปุ่ม (Button) สามารถสร้างโดยมีรูปแบบการสร้าง ดังนี้ **JButton** *bName* = new **JButton**(text, icon); 3.2 ป้ายแสดงข้อความ (Label) มีรูปแบบการสร้าง ดังนี้ JLabel *lName* = new JLabel(text, icon, alignment); 3.3 ช่องรับข้อความ (TextField) มีรูปแบบการสร้าง ดังนี้ JTextField *tName* = new JTextField (text, length); 3.4 ช่องรับรหัสผ่าน (PasswordField) มีรูปแบบการสร้าง ดังนี้ JPasswordField pwName = new JPasswordField (text, length); 3.5 ตัวเลือกหนึ่งรายการ (RadioButton) มีรูปแบบการสร้าง ดังนี้ JRadioButton rbName = new JRadioButton (text, icon, boolValue); 3.6 รายการข้อมูล (List) มีรูปแบบการสร้างรายการข้อมูล ดังนี้ JList *ltName* = new JList (item); 3.7 ตัวเลือกรายการมากกว่าหนึ่งตัวเลือก (CheckBox) มีรูปแบบการสร้างรายการข้อมูล ดังนี้ JCheckBox ckName = new JCheckBox (text, icon); 3.8 ตัวเลือกรายการหนึ่งตัวเลือก (ComboBox) มีรูปแบบการสร้างรายการข้อมูล ดังนี้ JComboBox cbName = new JComboBox (item); 3.9 เมนู (Menu) มีรูปแบบและขั้นตอนในการสร้างเมนู ดังนี้ 3.9.1 สร้างเมนูบาร์ด้วย JMenuBar MyMenuBar = new JMenuBar(); 3.9.2 สร้างเมนูด้วย JMenu MyMenu = new JMenu(StringName); รายการเมนูย่อยJMenultem 3.9.3 MyMenultem = new JMenultem(iTemName); 3.9.4 การเพิ่มรายการเมนูย่อยลงในเมนูด้วย MyMenu.add(MyMenuItem); 3.9.5 การเพิ่มเมนูลงในเมนูบาร์ด้วย MyMenuBar.add(MyMenu); 3.9.6 การสั่งให้เมนูบาร์ลงในเฟรมด้วย FrameName.setJMenuBar(MyMenuBar);

ขั้นตอนที่ 4 การจัดวางองค์ประกอบ (Layout Manager)

FlowLayout เป็นการจัดวางองค์ประกอบให้ทำการแสดงที่เรียงลำดับตามบรรทัดแรกให้เต็ม บรรทัดก่อนจึงทำการให้แสดงในบรรทัดถัดไป จนครบองค์ประกอบที่ถูกสั่งให้ทำการแสดง ซึ่งมีรูปแบบ ดังนี้

JPanel *flName* = new JPanel();

flName.setLayout(new FlowLayout());

BorderLayout เป็นการจัดวางองค์ประกอบให้แสดงตามทิศทั้ง 5 ตำแหน่ง คือทิศเหนือ (North) ทิศใต้ (South) ทิศตะวันออก (East) ทิศตะวันตก (West) และจัดให้อยู่กึ่งกลาง ซึ่งมีรูปแบบ ดังนี้

JPanel bflName = new JPanel();

flName.setLayout(new BorderLayout());

GridLayout เป็นการจัดวางองค์ประกอบให้ทำการแสดงในช่อง (grid) เป็นลักษณะตารางซึ่งมี รูปแบบดังนี้

JPanel glName = new JPanel();

glName.setLayout(new GridLayout(Row, Column));

สร้าง Basic Event Handling

โดยมีละขั้นตอนในการสร้าง ดังนี้

1. สร้างออบเจกต์ JRadioButton ด้วย

JRadioButton rB = new JRadioButton ("Single Choice");

 กำหนด Event ให้กับ ออบเจกต์ JRadioButton และ addActionListener (listener) เป็นคำสั่งที่ใช้ในการดักฟังเหตุการณ์เมื่อมีการคลิกที่คอมโพเนนต์ ด้วย

rB.addActionListener (new ActionListener ()

3. การคืนค่าคอมโพเนนต์ เมื่อมีการเกิดเหตุการณ์ (Event) ขึ้น ด้วย

JRadioButton rB = (JRadioButton) event.getSource ();

4. การตรวจสอบสถานะของ RadioButton ว่าถูกเลือกอยู่หรือไม่ถูกเลือกอยู่ ด้วย

boolean isSelected = radioButton.isSelected ();

5. การแสดง Message Dialog ด้วย

JOptionPane.showMessageDialog (null, "Click to selected.");

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- ให้นักศึกษาเขี้ยนโปรแกรม ตามตัวอย่างที่ 7.1 ในการใช้ Frame และ Panel พร้อมการ กำหนดสี
- ให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่า ทำงานอย่างไรโดยนำผลการทดลองมาประกอบการอธิบาย
- ให้นักศึกษาเขียนโปรแกรม โดยใช้เครื่องมือ ComboBox ให้ได้ผลการทำงานโปรแกรม ดัง ภาพที่ 7.19
- ให้นักศึกษาเขียนโปรแกรมตาม ตัวอย่าง 7.3 โปรแกรมในการแสดงค่าของเมธอด drawOval() กับเมธอด fillOval()
- 7. ทำการแปลภาษาโปรแกรม สั่งให้โปรแกรมทำงาน ให้ผลการทำงาน
- ให้นักศึกษาทำการค้นคว้าทำการเขียน ActionListener รองรับการคลิกเมาส์ และการรัน ข้อความลงใน TextBox จำนวน 20 ตัวอักษร
- ให้นักศึกษาเขียน GUI ให้สามารถรับค่ามาทำการคำนวณคณิตศาสตร์อย่างมีระบบ 1 โปรแกรม

การทดลองที่ 8 การติดต่อกับกับแฟ้มข้อมูล

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมจาวาสร้างการติดต่อกับแฟ้มข้อมูลได้
- 2. เพื่อให้สามารถนำส่วนการตรวจจับด้วย การใช้คำสั่ง try ... catch / finally
- 3. เพื่อให้สามารถเขียนโปรแกรมจาวาสร้างการติดต่อด้วยคาร์แร็คเตอร์สตรีมได้
- 4. เพื่อให้สามารถมีความสามารถในการวิเคราะห์แยกแยะการทำงานแฟ้มข้อมูลได้
- 5. เพื่อให้เกิดทักษะในการแก้ปัญหาโปรแกรมภาษาจาวาได้

ความรู้พื้นฐาน

ในการส่งข้อมูลระหว่างตัวโปรแกรม กับแหล่งที่เก็บข้อมูลจะเป็นกระบวนการที่เกิดขึ้นได้ 2 ลักษณะคือ การอ่าน (Read) กับ การเขียน (Write)

การอ่านไฟล์สตรีม (File Stream Reading) ต้องมีการตรวจสอบความถูกต้องหรือ การดัก จับ เออเรอร์ (Error) ทั้งด้านอินพุตและเอาต์พุต (IOException) ว่าแฟ้มข้อมูลที่ได้ทำการติดต่อนั้นมี อยู่จริง หรือมีข้อมูลอยู่หรือไม่ ถ้าไม่มีการทำงานของ IOException จะทำการฟ้องชนิดของการ Error ให้ผู้ใช้โปรแกรมทราบข้อผิดพลาด มีรูปแบบดังนี้

public static void main(String[] args) throws IOException สามารถสั่งงาน ดังนี้

File filename	= new File ("path");
FileInputStream fInput	= new FileInputStream(filename);
<pre>fInput.read();</pre>	

การเขียนไฟล์สตรีม (File Stream Writing) มีรูปแบบการสั่งงาน ดังนี้

File filename	= new File ("path");
FileOutputStream fOutput	= new FileOutputStream(filename,

append);

fOutput.write(data);

การใช้คำสั่ง try ... catch / finally เป็นการจัดการ Exception มีโครงสร้างการทำงาน คล้ายกับกลุ่มคำสั่ง if / else กล่าวคือถ้ามีการตรวจสอบเงื่อนไข ในที่นี้คือ Exception ที่อยู่หลัง try ว่า มีเหตุการณ์เกิดขึ้น ถือว่าถูกต้อง ก็จะให้โปรแกรมมาประมวลผลหลังคำสั่ง catch ส่วน finally เป็น ชุดคำสั่งให้ทำทุกเหตุการณ์ก่อนออกจาก try

คาร์แร็คเตอร์สตรีม เป็นสตรีมที่มีการจัดการข้อมูลที่ได้ทำการติดต่อกับแฟ้มข้อมูล เมื่อทำการ เรียกใช้แฟ้มข้อมูลที่ไม่มีอยู่ในหน่วยความจำ ก็จะทำการสร้างแฟ้มข้อมูลขึ้นมาตอบสนองการติดต่อของ Stream ได้ทันที

การอ่านข้อมูลคาร์แร็คเตอร์สตรีม มีรูปแบบดังนี้			
File filename	= new File ("path");		
FileReader fInput	= new FileReader(filename);		
data	= fInput .write();		
การเขียนข้อมูลคาร์แร็คเตอร์สตรีม มีรูปแบบดังนี้			
Filo filonomo	- new File ("nath"):		

File filename= new File("path");FileWriter fOutput= new FileWriter(filename [, append]);fOutput.write(data);

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม Edit Plus
- 4. ตัวโปรแกรมภาษาจาวา
- 5. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. เปิดโปรแกรมภาษาจาวา โปรแกรม Edit Plus บนเครื่องคอมพิวเตอร์ของนักศึกษา
- 2. ตรวจสอบที่อยู่ของคำสั่งต่างในโปรแกรม Edit Plus ก่อนใช้งาน
- ให้นักศึกษาเขียนโปรแกรม ตามตัวอย่างที่ 8.1 การใช้ Byte Stream และทำการจัดเตรียม แฟ้มข้อมูลที่ใช้ในการติดต่อกับโปรแกรมแกม ชื่อ D:/Java/DataForLaB8.txt และ LaB8Out.txt
- ให้ได้ผลการทำงาน ข้อมูลในแฟ้ม D:/Java/DataForLaB8.txt แสดงภาพที่ 8.13ม 8.14 และ 8.15
- ให้นักศึกษาเขียนอธิบาย โดยทำการวิเคราะห์การทำงานของแต่คำสั่ง ในแต่ละบรรทัดว่า ทำงานอย่างไรโดยนำผลการทดลองมาประกอบการอธิบาย
- 6. ให้นักศึกษาเขียนโปรแกรม เรื่อง try... catch ให้ได้ผลการทำงานโปรแกรม (ภาพที่ 8.16)
- ให้นักศึกษาเขียนโปรแกรมตาม ตัวอย่าง 8.3 โปรแกรมในการแสดง character stream โดยใช้ try... catch ร่วมกับการใช้คำสั่งการติดต่อแฟ้มข้อมูลระดับไฟล์ ด้วยคำสั่ง FileReader และ FileWriter
- 8. ให้นักศึกษาเขียนผลการทดลองของโปรแกรมและอธิบาย การทำงานมาพอเข้าใจ
- ให้นักศึกษาทำการค้นคว้าทำการเขียน ให้นักศึกษาทำการค้นคว้าทำการเขียน Buffer Stream รองรับการป้อนข้อมูล จำนวน 10 ตัวอักษร
- 10. ให้นักศึกษาทำการค้นคว้าทำการเขียนข้อมูลด้วยคลาส PrinterWriter

การทดลองที่ 9 การประยุกต์การติดต่อฐานข้อมูลกับเครือข่าย

วัตถุประสงค์

- 1. เพื่อให้สามารถเขียนโปรแกรมจาวาบนเน็ตบีนส์ได้
- 2. เพื่อให้สามารถเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอลได้
- 3. มีทักษะการโปรแกรมใช้มายเอ็สคิวแอลบนเครือข่ายได้

ความรู้พื้นฐาน

การติดตั้งเน็ตบีนส์

สามารถดาวน์โหลดได้ที่ <u>https://netbeans.org/downloads/</u>

การใช้งานเน็ตบีนส์

เรียกโปรแกรมเน็ตบีนส์ จาก Start->All Programs ->NetBeans->NetBeans IDE 8.0

สร้างโปรเจ็ค (Project)

ด้วยการเลือกรายการเมนู File->New Project

สั่งให้ทำงาน

้สั่งให้ทำการทดสอบและการทำงานของโค๊ดด้วยการคลิกปุ่ม Play 🕨

การติดตั้ง AppServ

การติดตั้งระบบเครือข่ายคอมพิวเตอร์ สำหรับการเขียนการควบคุมหน้าเว็บไซต์ โปรแกรมฐานข้อมูลมายเอ็สคิวแอล (MySQL) และ phpMyAdmin ซึ่งชุดพัฒนาสามารถ ดาวน์โหลด ได้ที่ <u>http://sourceforge.net/projects/appserv/files/</u>

การเชื่อมต่อเน็ตบีนส์กับมายเอ็สคิวแอล

เปิดโปรแกรม NetBeans เพื่อทำการเปิด Services ของ Database ผ่านทาง Driver ให้สามารถเชื่อมต่อ MySQL

ทำการเปิด Services ก็สามารถนำ NetBeans ทำการเชื่อมต่อกับ MySQL ผ่านทาง Localhost

ทำการกำหนด Configuration ให้กับ Host ที่เป็น localhost คือ User Name : คือ root ส่วน Password : คือ อย่างน้อย 4 ตัวอักษร

การเพิ่ม Library ของ NetBeans กับ MySQL jdbc Driver

การกำหนดให้มีการทำงานร่วมกันระหว่าง NetBeans กับ MySQL มีความจำเป็นต้อง อาศัย Driver ของฐานข้อมูล MySQL อย่างต่อเนื่อง ในการแปลงข้อมูลระหว่างกัน สามารถ การเพิ่ม Library ของ NetBeans กับ MySQL jdbc Driver โดยการคลิกขวาที่ Libraries แล้ว ทำการเลือก Add Library ทำการเลือก MySQL JDBC Driver

การกำหนดคุณลักษณะมายเอ็สคิวแอล

การกำหนดสามารถทำได้โดยการคลิกขวาที่ Databases->MySQL Server at localalhost: 3306 [root] (disconnection) แล้วทำการเลือก Properties

ทำการกำหนด Administration Password เป็น **1234** (สามารถเว้นว่างไว้ได้) ตามที่ กำหนดขณะทำการติดตั้ง และเมื่อต้องการให้มีการจำรหัส สามารถทำได้โดยการทำ เครื่องหมายในบล็อก Remember Password แล้วคลิก OK

ทำการกำหนดคุณสมบัติของผู้บริหารระบบได้โดยการคลิกแท็บ Admin Properties โดยทำการกำหนดในส่วนของ Path/URL to admin tool: ส่วนของ Path to start command: และส่วนของ Path to stop command: ซึ่งมีปุ่ม **Browse**

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. ตัวโปรแกรม NetBeans
- 4. ตัวโปรแกรม Appserv
- 5. ตัวโปรแกรม MySQL
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. ให้นักศึกษาทำการติดตั้ง NetBeans ไว้ที่ Dive D:
- 2. ให้นักศึกษาทำการสร้าง Project ชื่อ HelloBeans ไว้ที่ D:\Breans
- 3. ให้โค้ดให้แสดง ชื่อ รหัสนักศึกษา และสาขาวิชาของนักศึกษา
- 4. ให้ทำการทดสอบการทำงานของโค๊ด และให้บันทึกผลการทำงานโปรแกรม
- 5. ให้นักศึกษาทำการติดตั้ง MySQL ด้วย Appserv ลงในไว้ที่ Drive D:
- 6. ให้นักศึกษาทำการทดสอบการทำงานของ NetBeans กับ MySQL ให้สมบูรณ์

การทดลองที่ 10 การประยุกต์การติดต่อฐานข้อมูลกับเครือข่าย

วัตถุประสงค์

- 1. เพื่อให้สามารถให้เน็ตบีนส์เชื่อมต่อกับ MySQL ได้
- 2. เพื่อให้สามารถสร้างฐานข้อมูลด้วยเน็ตบีนส์ บนมายเอ็สคิวแอลได้
- 3. มีทักษะการสร้างตารางบนมายเอ็สคิวแอลของเครือข่ายได้

ความรู้พื้นฐาน

สร้างและทำการเชื่อมต่อฐานข้อมูลให้พร้อมใช้งาน

การติดต่อกับฐานข้อมูลสามารถกระทำได้โดยการกระทำผ่านทางตัวอิดิเตอร์ของมายเอ็สคิว แอล (SQL editor) ซึ่งตัว NetBeans IDE ได้ติดตั้ง SQL Editor สามารถทำการสร้างฐานข้อมูลใหม่ที่ใช้ ในการงานได้ ด้วยทำการสร้างฐานข้อมูลโดยการคลิกขวาที่โนด MySQL Server at locationhost: 3306 [root] และเลือกรายการเมนูย่อย Create Database ระบบจะมีกล่องข้อความ Create MySQL Database ให้ทำการกรอกชื่อฐานข้อมูลลงในช่อง New Database Name: แล้วทำการคลิก OK

สร้างตารางให้กับฐานข้อมูล (Creating Database Tables)

การสร้างตารางฐานข้อมูลสามารถทำได้สองวิธีคือ สร้างตารางได้จากกล่องข้อความ (Dialog) กับการสร้างตารางโดยใช้ตัว SQL Editor ดังนี้

การสร้างตารางด้วย SQL Editor (Using the SQL Editor)

การสร้างตารางจาก SQL Editor ของ NetBeans IDE มีขั้นตอนดังนี้

- จากตัว Explorer ของ Database เมื่อทำการขยายแล้วให้ทำการเชื่อมต่อโนด จะมี โฟลเดอร์ย่อยต่อจาน jbdc:mysql://localhost:3306/mynewdatabase... จำนวน 3 โฟลเดอร์ย่อย Tables Views และ Procedures
- จากนั้นให้ทำการคลิกขวาที่โฟลเดอร์ Tables แล้วทำการเลือก Execute Command จะได้พื้นที่ว่างของ main window ที่ใช้ในการสร้างโค๊ดของ SQL Editor
- ทำให้มีพื้นที่ของ SQL editor สามารถทำการสร้างตารางได้ตามต้องการ

การสร้างตารางด้วย Table Dialog (Using the Create Table Dialog)

สร้างตารางได้ด้วยการทำงานของที่ละ Dialog ซึ่งในตัวอย่างนี้จะทำการสร้างตารางชื่อ Subject มีขั้นตอน ดังนี้

- ตัว Explorer เมื่อทำการขยายแล้วตัวExploreของDatabaseจะมีโฟลเดอร์ ย่อยต่อจาก กรี jbdc:mysql://localhost:3306/mynewdatabase... มี โฟลเดอร์ย่อย Tables ทำการคลิกขวาที่โฟลเดอร์ Tables แล้วทำการเลือก Create Table ตัวกล่องข้อความที่ใช้สร้างตาราง (Create Table Dialog) ถูกเปิดขึ้น ให้ทำการพิมพ์ชื่อตารางลงในช่อง Table Name: Subject และ ทำการคลิกปุ่ม Add column
- เมื่อได้ทำการกำหนดชื่อตารางเป็น Subject เพื่อทำการสร้างข้อมูล ด้วยการ พิมพ์ข้อมูลลงใน Dialog
- ทำการเพิ่มคอลัมน์ให้กับตาราง Subject ด้วยการคลิกปุ่ม Add column ทำ การพิมพ์ข้อมูล เป็นตัวกำหนดตารางโดยทำการสร้างให้ครบทั้ง 3 รายการ
- เมื่อทำการสร้างตารางครบตามที่ได้ออกแบบข้อมูลในตารางที่ 10.1 แล้ว สามารถทำการตรวจข้อมูลได้

การสั่งงานตารางข้อมูล (Working with Table Data)

ในการสั่งให้การทำงานร่วมกันระหว่างตารางข้อมูลบน SQL Editor ของ NetBeans สามารถ ให้ทำงานในรูปแบบของ SQL queries บนฐานข้อมูลได้หลากหลาย การปรับปรุงแก้ไขโครงสร้างของ ฐานข้อมูล การเพิ่ม(add) การแก้ไข(modify) และการลบ(delete) รวมทั้งเพิ่มรายการข้อมูล (record หรือ row) ลงในตารางขอมูล Counselor

การสั่งการงานฐานข้อมูลด้วย Script (Running an SQL Script)

ในการจัดการเรื่องตารางข้อมูลของ NetBeans สามารถสั่งให้ทำงานได้โดยตรงด้วยการดาวน์ โหลดที่ <u>https://netbeans.org/project_downloads/samples/Samples/Java%20Web/ifpwafcad.sql</u> แล้ว ทำการบันทึกไว้ในตำแหน่งเดียวกับโปแกรม NetBeans IDE อยู่ ในที่เป็น C:\Program Files (x86)\NetBeans 8.0\bin สคริปต์จะทำการสร้างตาราง 2 ตารางที่คล้ายกับการสร้างตาราง Counselor และ Subject โดยสคริปต์ได้ทำการเปลี่ยนชื่อตารางเป็น Counselor2 และ Subject2 ตามลำดับ ในการสั่งสคริปต์ทำงานสามารถดำเนินการได้

แนะนำการโปรแกรมบนเครือข่ายด้วยจาวา

การนำโค็ดที่ได้ทำการบันทึกไว้ที่ C:\AppServ\www ไปทำการเก็บไว้บนเครือข่าย คอมพิวเตอร์จริง หรือ IP Address จริง เป็นการทำงานจริงของการเขียนโปแกรมบนเครือข่าย คอมพิวเตอร์ได้ ซึ่งเป็นการสร้างเว็บไซต์บน MySQL database server ซึ่ง MySQL เป็นฐานข้อมูล ที่ใช้ในการจัดการกับข้อมูลที่ใช้ในการประยุกต์ใช้บนเว็บ ด้วยอาจเป็นเพราะ MySQL มีความโดดเด่น ด้านความเร็ว (speed) ด้านการยืดหยุ่น (flexibility) และด้านความเป็นธรรมชาติ (reliability) MySQL ใช้ภาษา SQL ในการเขียนซึ่งมีโครงสร้างของภาษาเป็น Query Language สำหรับการเข้าถึง และการ ประมวลผลข้อมูลที่อยู่ในฐานข้อมูล จึงทำให้ MySQL เป็นระบบมีความนิยมมากที่สุดในปัจจุบัน

อุปกรณ์

- 1. เครื่องคอมพิวเตอร์
- 2. ระบบอินเทอร์เน็ต สำหรับการค้นคว้า
- 3. เครื่องคอมพิวเตอร์ควรมีซอฟต์แวร์ NetBeans IDE เวอร์ชั่น 7.2, 7.3, 7.4, 8.0
- 4. เครื่องคอมพิวเตอร์ควรมีซอฟต์แวร์ Java Development Kit (JDK) เวอร์ชั่น 7 หรือ 8
- 5. เครื่องคอมพิวเตอร์ควรมีซอฟต์แวร์ MySQL database server เวอร์ชั่น 5.x
- 6. หน่วยความจำเคลื่อนที่ (Handy Drive หรือ External Hard disk)

วิธีทดลอง

- 1. ให้นักศึกษาทำการทดสอบการทำงานของ NetBeans กับ MySQL ให้สมบูรณ์
- 2. ให้นักศึกษาทำการสร้างฐานข้อมูลที่นักศึกษาต้องการ 1 ฐานข้อมูล
- ให้นักศึกษาสร้างตารางที่ใช้ในการเก็บรายชื่อเพื่อนในห้องเรียนอย่างน้อย 20 รายชื่อ
- 4. เมื่อสร้างตารางสำเร็จตามต้องการแล้ว
 - a. ฐานข้อมูลต้องสามารถเพิ่มชื่อเพื่อนได้
 - b. ฐานข้อมูลต้องสามารถลบชื่อเพื่อนได้
 - c. ฐานข้อมูลต้องสามารถแทรกชื่อเพื่อนได้
- 5. ให้นักศึกษาน้ำฐานข้อมูลวางที่เครือข่ายจำลองหรือ IP จริงตามความเหมาะสม

ดัชนี

A

Abstraction · 31 Address · 49, 230, 232, 239, 259 Apache · 228 Applet · 111, 133 Applet Life Cycle · 133 Applet with Graphic · 140 Applet with Image · 144 Applet with Sound · 143 Applet with Thread \cdot 146 Appserv · 228 AppServ · 228 argument · 38 arithmetic · 59, 60, 61, 62 ASCII · 52, 53, 54, 68 assignment · 58, 59, 61 AWT · 159, 160, 161, 162, 163, 164, 165, 166, 167, 190, 191

B

bitwise · 64 BorderLayout · 180 braces · 38 brackets · 38 break · 93 Button · 169 byte · 37, 50, 64, 65 Byte Stream · 197

C

char · 30, 37, 50, 51, 52, 53, 64, 65, 79, 98, 125 Character Stream · 203 CheckBox · 174 class · 105 Class · 12, 30, 105, 107, 108, 185, 224 ComboBox · 174 comma · 38 Command Prompt · 18 comment \cdot 36, 37 compiler · 11 Component · 169 Components · 163 Computer Programming · 9 Configure User Tools · 21 Configuring MySQL Server Properties · 236 constructor · 122 Constructor · 121 Containers · 163 continue · 94 Control statements · 72 CPU · 9, 146 Creating and Connecting to the Database Instance · 243

D

data type · 50 Decision · 72, 73, 101 decrement · 62 destroy() · 134 dir · 19 do...while · 89 DOS · 11, 18, 19, 29 double · 37, 39, 40, 43, 50, 52, 56, 57, 65, 79

E

Edit Plus · 20 Encapsulation · 31 Enterprise Edition · 13 Environment Variables · 17 Event · 184 expressions · 38, 58

F

File Stream · 196, 198 File Stream Writing · 200 floating · 39, 60, 108 FlowLayout · 177 for · 83 full stop · 38

G

GridLayout · 182 GUI · 161, 162

H

Handling · 159, 183, 184, 190 HTML · 137

I

identifiers · 36, 37 if-else · 73 increment · 62 Inheritance · 31 init() · 134 instance · 23, 30, 38, 103, 106, 107, 115, 116, 124, 125 int · 30, 37, 39, 40, 49, 50, 55, 65, 67, 68, 77, 79, 81, 82, 97, 98, 99, 100, 115, 120, 121, 125, 126, 127, 128, 147, 148, 149, 150, 151, 205, 207, 209, 212 integer · 38, 39, 55, 59, 64, 82

J

J2SDK · 13 Java 2 Software Developer Kit · 13 Java Runtime Environment · 12 Java Virtual Machine · 12, 133 javac · 20 JFrame · 167 JPanel · 168 JRadioButton · 184

K

keywords · 36, 37

L

Label · 169 Layout Manager · 177 List · 173 literals · 36, 38 long · 37, 50, 57, 64, 65, 79, 149 Loop · 72, 83, 94, 95, 101

M

Memory Unit · 9 Menu · 175 Micro Edition: Java · 13 MySQL · 226, 228 MySQL jdbc Driver · 235 MySQL Server Configuration · 231

N

Netbeans · 218 NetBeans · 233, 235

0

object · 30, 106, 114 Operating System · 9 operator · 58, 59, 60, 61, 62, 64 Operator · 47 other control · 93 Output Unit · 9 overloading · 122

P

package · 112 paint() · 134 Panel · 164 parentheses · 38 Pass by reference · 120 Pass by value · 119 Path · 16 phpMyAdmin · 228 Platform of Java · 13 Polymorphism · 31 postfix · 63 prefix · 63

R

RadioButton · 172 Read · 196 Relational · 65, 66, 228 return · 95 root · 230, 233, 234, 236, 239, 243, 244

<mark>S</mark>

Script · 218, 241, 254, 260 ScrollPane · 164 semicolon · 38 separators · 36, 37 Sequence \cdot 72, 101 short · 37, 50, 54, 55, 64, 65, 67, 79 source code \cdot 10, 11 source file \cdot 11 source program \cdot 10 Standard Edition · 13 start() · 134 Static · 123 Static method member · 124 stop() · 134 stream · 197 Stream · 196 sun microsystem co., ltd · 11 Swing · 159, 160, 161, 162, 166, 167, 190, 193, 214 switch-case · 79

Т

Tables · 245, 246, 248, 249, 252 TextField · 170 Thread · 146 try ... catch · 202

V

variable · 39, 41, 48, 77, 79, 80, 82, 106, 39, 47, 68

W

while · 86

white space · 36, 42 Write · 196