

# ใบงานการทดลองที่ 13

## เรื่อง การใช้งาน Inner Class และการใช้งาน Thread

### 1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจการโปรแกรมเชิงวัตถุ การกำหนดวัตถุ การใช้วัตถุ
- 1.2. รู้และเข้าใจการทำงานหลายงานพร้อมกัน

### 2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

### 3. ทฤษฎีการทดลอง

- 3.1. Nest Class คืออะไร? มีวัตถุประสงค์เพื่ออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

เป็น Class ที่ประกาศภายใน body ของ Class หรือ Interface อื่นๆ

จุดประสงค์หลักของการสร้าง Nested Classes คือการ group Class และ Interface ที่เกี่ยวข้องกันให้อยู่ภายใน File เดียวกัน

ถึงแม้ว่าการทำ Package ก็ช่วยในเรื่องดังกล่าวแล้วแต่การทำ Nested Classes ทำให้การ group แข็งแรงมากขึ้นอีกชั้น

- 3.2. จงยกตัวอย่างการสร้าง Inner Class

- 3.3. จงยกตัวอย่างการเรียกใช้งาน Instance ที่มีการเรียกใช้งาน Properties ภายใน Inner Class

```
Public static void main( String[] args ){
```

```
OuterClass outerClass = new OuterClass.InnerClass();
```

```
outerClass.test += 10 ;
```

- 3.4. จงยกตัวอย่างการเรียกใช้งาน Instance ที่มีการเรียกใช้งาน Method ภายใน Inner Class

```
Public static void main( String[] args ){
```

```
OuterClass outerClass = new OuterClass.InnerClass();
```

```
outerClass.printData();
```

```
}
```

- 3.5. Thread คืออะไร? มีประโยชน์อย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

Thread คือระบบของจาวาสำหรับการสนับสนุนการทำงานแบบ multi-tasking แบบที่ในระบบปฏิบัติการก็จะให้โปรแกรมสามารถทำงาน

พร้อมกันได้ เช่น ฟังเพลงไปด้วยพิมพ์งานไปด้วยก็ได้ นอกจากนี้เรายังสามารถทำงานพร้อมกันได้ด้วยเรียกว่า multi-thread

ประโยชน์จาก Thread นั้นโปรแกรมจะต้องเป็นแบบ Multithreading ซึ่งจะมีข้อได้เปรียบ เช่น มีการตอบสนองของโปรแกรมที่ดีกว่า

การประมวลผลเร็วกว่า ใช้ทรัพยากรน้อยกว่า การใช้ประโยชน์จากระบบมากกว่า และการทำงานแบบขนาน

3.6. การเริ่มต้นใช้งาน Thread มีขั้นตอนอย่างไรบ้าง?

---

---

---

---

3.7. ระหว่าง Thread และ Runnable มีรูปแบบการใช้งานที่เหมือนหรือแตกต่างกันอย่างไร?

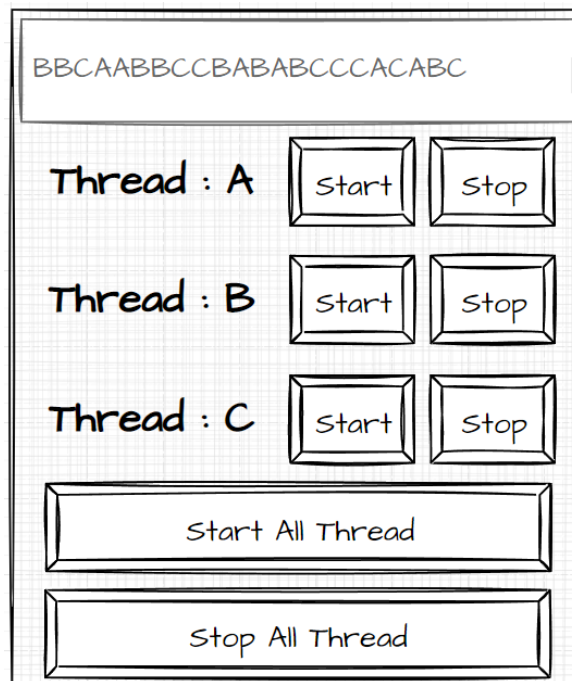
Thread เป็นคลาสในแพ็คเกจ java.lang คลาสที่ขยายคลาสของ วัตถุ และใช้อินเตอร์เฟซ Runnable คลาส Thread มีตัวสร้างและวิธีการในการสร้างและดำเนินการกับเธรด Runnable เป็นอินเตอร์เฟซในแพ็คเกจ java.lang การใช้อินเตอร์เฟซที่เรียกใช้งานได้นั้นเราสามารถกำหนดเธรดได้ ส่วนต่อประสานที่ Runnable ได้มีวิธีการเดียว run () ซึ่งนำมาใช้โดยคลาสที่ใช้ส่วนต่อประสาน Runnable มันเป็นที่ต้องการที่จะใช้อินเตอร์เฟซที่เรียกใช้แทนการขยายชั้นเรียนด้วย เนื่องจากการใช้ Runnable ทำให้โค้ดของคุณเชื่อมโยงกันอย่างหลวม ๆ เนื่องจากโค้ดของเธรดต่างจากคลาสที่กำหนดงานให้กับเธรด มันต้องใช้หน่วยความจำ น้อยลงและยังช่วยให้ชั้นเรียนที่จะรับช่วงอื่น ๆ

3.8. สถานะ Deadlock มีลักษณะเป็นอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

ซึ่งเป็นสถานการณ์ที่ 2 thread หรือมากกว่าถูกล็อก (LOCKED) ตลอดกาล ซึ่งรอกันและกันให้ทำงานให้เสร็จก่อน ซึ่งในบทความนี้จะมาคุยกันเรื่องนี้โดยใช้ปัญหาอาหารเย็นของนักปราชญ์ (Dining Philosophers) ที่เป็นปัญหาคลาสสิกที่กล่าวถึงปัญหาการ synchronization ในสภาวะแวดล้อม multi-thread และให้เห็นภาพทางเทคนิคของการแก้ไขปัญหาของปัญหานี้

4. ลำดับขั้นการปฏิบัติการ

- 4.1. จงสร้างหน้าต่าง GUI เพื่อทำการทดสอบสร้าง Thread ที่มีส่วนประกอบดังต่อไปนี้
  - 4.1.1. สร้าง Thread A ที่สร้างจาก Inner Class
  - 4.1.2. สร้าง Thread B และ C จาก Class ปกติ
  - 4.1.3. แต่ละ Thread จะมีปุ่ม Start เพื่อเริ่มต้นพิมพ์ตัวอักษรของ Thread ลงในช่อง Textbox และ Stop เพื่อหยุดการพิมพ์ตัวอักษรของ Thread ในช่อง Textbox
  - 4.1.4. สร้างปุ่ม Start All Thread เพื่อทำให้ Thread แต่ละตัวทำงานพร้อมกัน
  - 4.1.5. สร้างปุ่ม Stop All Thread เพื่อให้ Thread แต่ละตัวหยุดทำงานพร้อมกัน



โค้ดโปรแกรมของปุ่ม Start และ Stop ของ Thread A

โค้ดโปรแกรมของปุ่ม Start และ Stop ของ Thread B
โค้ดโปรแกรมของปุ่ม Start และ Stop ของ Thread C

--

โค้ดโปรแกรมของปุ่ม Start All Thread

โค้ดโปรแกรมของปุ่ม Stop All Thread

## 5. สรุปผลการปฏิบัติการ

การใช้งาน thread นั้นเป็นการทำ งานแบบขนานที่ทำ งานหลายๆคำ สั่งพร้อมๆกันโดยที่ไม่ตนเองทำ งานเป็นลำดับ งานใดทำ เสร็จก่อนก็ทำการ return ก่อน

---

---

---

---

---

## 6. คำถามท้ายการทดลอง

6.1. Inner Class แตกต่างจาก Class แบบปกติอย่างไร?

---

---

---

---

6.2. เมื่อใดจึงเป็นช่วงเวลาที่ดีที่สุดในการใช้งาน Inner Class  
หาก code เริ่มที่จะซับซ้อนและจำเป้นที่จะต้องสร้างอีก class แต่ไม่ยากทำไฟล์แยก

---

---

---

---

6.3. ข้อควรระวังในการใช้งาน Thread คืออะไร?

---

---

---

---