

บทที่ 11 : โปรแกรมแบบเธรด

วัตถุประสงค์

- เพื่อให้เข้าใจขั้นตอนการพัฒนาโปรแกรมแบบเธรดโดยการสืบทอดมาจากคลาสที่ชื่อ Thread หรือ implements อินเตอร์เฟสที่ชื่อ Runnable
- เพื่อให้สามารถเรียกใช้เธรดต่างๆ ที่เกี่ยวข้องกับการทำงานของคลาส Thread

แบบฝึกหัดเชิงปฏิบัติการ

แบบฝึกหัดที่ 1 โปรแกรมนาฬิกาแบบดิจิทัล

วัตถุประสงค์

- 1) เพื่อให้เข้าใจการเขียนโปรแกรมแบบเธรด
- 2) เพื่อให้เข้าใจและสามารถเรียกใช้เธรดที่สำคัญของคลาสที่ชื่อ Thread

คำสั่ง

ทดลองเขียนโปรแกรมนาฬิกาแบบดิจิทัล โดยให้มีการทำงานแบบเธรด

ขั้นตอนการปฏิบัติการ

- 1) เขียนคลาสที่ชื่อ MyClock ซึ่งสืบทอดมาจากคลาส JLabel และกำหนดให้เป็นคลาสที่ implements อินเตอร์เฟส Runnable โดยมีคำสั่งประกาศดังนี้

```
public class MyClock extends JLabel implements Runnable
```
- 2) เขียนเมธอด run() โดยมีคำสั่งต่างๆดังนี้
 - คำสั่ง while เพื่อให้โปรแกรมวนอยู่ในบล็อกคำสั่งโดยไม่มีที่สิ้นสุดดังนี้

```
while (true) {  
    .....  
}
```
 - ภายในบล็อกคำสั่ง while ให้สร้างออปเจกต์ของคลาส Calendar โดยใช้คำสั่ง

```
Calendar d = Calendar.getInstance()
```
 - เรียกคำสั่ง get() ในคลาส Calendar เพื่อหาค่าชั่วโมง นาทีและวินาทีของเวลาปัจจุบัน โดยใช้คำสั่ง

```
int sec = d.get(Calendar.SECOND);  
int min = d.get(Calendar.MINUTE)  
int hour = d.get(Calendar.HOUR_OF_DAY);
```
 - เรียกใช้เมธอด setText() เพื่อกำหนดข้อความของ JLabel ให้เป็น hour:min:sec
 - เรียกใช้เมธอด Thread.sleep() เพื่อให้ออปเจกต์แบบ Thread หยุดทำงาน 1 วินาที แล้ววนกลับมาทำคำสั่งสร้างออปเจกต์ของคลาส Calendar ใหม่
- 3) เขียนคลาสที่ชื่อ MyFrame ที่มีเมธอด main() อยู่ภายในและมีคำสั่งต่างๆ ดังนี้
 - ประกาศและสร้างออปเจกต์ของคลาส JFrame
 - ประกาศและสร้างออปเจกต์ที่ชื่อ clock ของคลาส MyClock
 - ประกาศและสร้างออปเจกต์ที่ชื่อ t ของคลาส Thread โดยมี argument เป็นออปเจกต์ clock
 - เรียกใช้เมธอด start() ของออปเจกต์ t
 - ใส่ออปเจกต์ clock ลงในออปเจกต์ของคลาส JFrame
 - กำหนดขนาดของออปเจกต์ของคลาส JFrame และเรียกใช้เมธอด setVisible(true)

4) คอมไพล์และรันโปรแกรม `MyFrame.java`

แบบฝึกหัดทบทวน

จงเขียนโปรแกรมที่สร้างออปเจกแบบเรตขึ้นมา 100 ตัว แต่ละออปเจกจะทำการบวกเลขให้กับตัวแปร `sum` ทีละ 1 โดยเราจะต้องส่งตำแหน่งอ้างอิงของตัวแปร `sum` ให้กับออปเจกแบบเรตแต่ละตัว ทั้งนี้กำหนดให้ `sum` เป็นออปเจกของคลาสประเภท `Wrapper`