



Thread ในจาวา (2)

အိပ်.....

รหัสนักศึกษา.....สาขา.....

ผู้สอน ผู้ช่วยศาสตราจารย์ ดร. ชูพันธุ์ รัตนโกศา

ในงานที่

4

วัตถุประสงค์

- ฝึกการเขียนโปรแกรมภาษาจาวาเพื่อสร้างและใช้งาน Thread

1. จงเขียนโปรแกรมเพื่อแสดงการแก้ไข Dining Philosophers โดยกำหนดให้รับจำนวนเต็ม n (int) จากผู้ใช้ เพื่อสร้าง Philosopher n คน และส้อม n อัน นั่งกินอาหารอยู่บนโต๊ะกลม โดยกำหนดให้ต้องเขียนทั้งหมด 3 คลาส : Philosopher, Fork และ DiningPhilosophers (เป็น main class)

Fork.java

```
public class Fork {
```

```
volatile boolean taked = false; // เก็บสถานะว่าส้อมนี้ถูกใช้งานอยู่หรือไม่
```

```
String holderName = " "; // เก็บชื่อ Philosopher ที่ใช้งานส้อมนี้อยู่
```

```
public String getHolderName() {
```

```
return holderName;
```

}

// Philosopher เรียกใช้เมื่อต้องการใช้ส้อม ถ้าส้อมไม่ว่างให้คืนค่า false

// ถ้าส้อมว่างจะต้องตักค่า `taked` และใส่ที่คอนที่จะให้ส้อม และคืนค่า `true` ว่าสามารถหยิบส้อมมาใช้งานได้

```
public synchronized boolean take(String holderName) {
```

}

//Philosopher จะเรียกใช้เมื่อต้องการเลิกใช้งานส้อมนี้

```
public synchronized void putDown() {
```

}

}

Philosopher.java

```
import java.util.*;

public class Philosopher extends Thread {
    Fork left, right;          // อ้างอิงถึง object ของส้อมด้านซ้ายและด้านขวาของ Philosopher
    String name;                // ชื่อของ Philosopher
    String status = "---";      // สถานะของ Philosopher
    int numEat = 0; boolean done = false; Random r = new Random();

    public Philosopher(String name, Fork left, Fork right) {    //รับชื่อ Philosopher และ object ของส้อมซ้าย-ขวา
        this.left = left; this.right = right; this.name = name;
    }

    public String getStatus() { return status; }                //เรียกเพื่อคืนค่าสถานะของ Philosopher
    public int getNumEat() { return numEat; }                   //เรียกเพื่อคืนค่าจำนวนครั้งที่ Philosopher ได้กินอาหาร
    public void done() { done = true; }                        //เรียกเพื่อหยุดการทำงานของ Philosopher
    public void think() {                                       //เรียกเพื่อให้ Philosopher อยู่ในสถานะ คิด (THK)
        status = "THK";
        try { Thread.sleep(r.nextInt(150) + 150); } catch(Exception e) {}
    }

    public void eat() {                                         //เรียกเพื่อให้ Philosopher อยู่ในสถานะ กิน (EAT)
        status = "EAT";
        try { Thread.sleep(r.nextInt(150) + 150); } catch(Exception e) {}
    }

    // Philosopher จะเริ่มต้นที่คิด เมื่อคิดเสร็จจะหยิบส้อม เมื่อได้ส้อม 2 อันถึงจะสามารถกินได้ เมื่อกินสำเร็จให้เพิ่มค่า numEat
    public void run() {
        while(!done) {

        }
    }
}
```

