# ใบงานการทดลองที่ 11 เรื่อง การใช้งาน Abstract และ Interface

## จุดประสงค์ทั่วไป

### รู้และเข้าใจการกำหนดวัตถุ การใช้วัตถุ การซ่อนวัตถุ และการสืบทอดประเภทของวัตถุ

### รู้และเข้าใจโครงสร้างของโปรแกรมเชิงวัตถุ

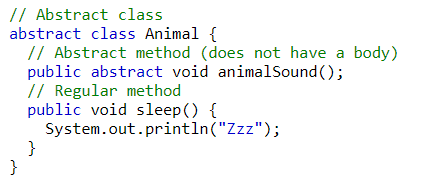
## เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์ 1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

## ทฤษฎีการทดลอง

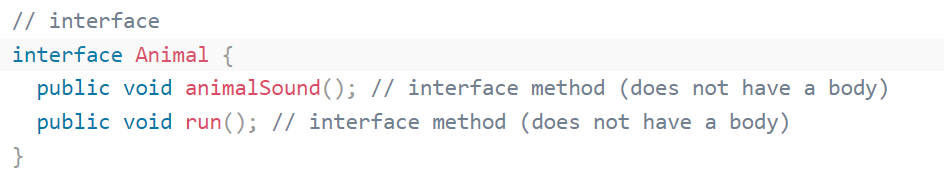
### Abstract Class คืออะไร? มีลักษณะการทำงานอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

คลาสที่ซ้อนรายละเอียดไว้เพื่อแสดงผล บางส่วน



### Interfaces คืออะไร? มีลักษณะการทำงานอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

คือ abstract class ที่สมบรูณ์แบบ เมื่อ method class ที่มีใน interface ไม่มีรายละเอียด



### คำสั่ง extends และ implements มีการใช้งานที่แตกต่างกันอย่างไร?

Extends ใช้สำหรับสืบทอด class

Implement ใช้สำหรับสือทอด interface ไปใช้

### ภายใน Abstract Class มี Constructor หรือไม่? เพราะเหตุใด?

ไม่มีเพราะเป็นการสร้างที่อยู่ตัวแปรเท่านั้น ส่วนกำหนดค่าจะไปทำในส่วนของ class ลูกที่สืบทอด

### ภายใน Interface มี Constructor หรือไม่? เพราะเหตุใด?

ไม่มี เพราะ คล้ายๆกับ abstract class แต่จะสมบูรณ์มากกว่าเลยจะไม่มีการกำหนดค่าภายในเลย

## ลำดับขั้นการปฏิบัติการ

### ให้ผู้เรียนสร้าง Abstract Class ของรถถัง(ClassicTank) โดยจะต้องมีรายละเอียดดังต่อไปนี้

#### Properties : HP เพื่อกำหนดค่าพลังให้กับรถถัง

#### Properties : Str เพื่อกำหนดค่าความแรงในการยิงของรถถัง

#### Properties : Vit เพื่อกำหนดค่าพลังป้องกันของรถถัง

#### Properties : BaseDamage เพื่อการกำหนดค่าพลังการโจมตีพื้นฐาน

#### Method : SetHP( ) ; เพื่อทำการกำหนดค่าพลังเริ่มต้น

#### Method : GetHP( ) ; เพื่อตรวจสอบค่าพลัง ณ เวลาปัจจุบัน

#### Method : Attack( Tank Enemy ) ; เพื่อทำการยิงปืนใหญ่โจมตีศัตรู โดยการโจมตี จะเป็นการลดค่าพลังของรถถังฝั่งตรงกันข้าม (Enemy คือรถถังของศัตรู, Points คือค่าพลังโจมตีของเรา)

### ให้ผู้เรียนสร้างคลาส NormalTank เพื่อสืบทอด ClassicTank เพื่อเขียนรายละเอียดของ Method ทั้งหมดอันได้แก่ SetHP() , GetHP() , Attack( Tank Enemy )

### ในคลาสหลัก ให้สร้าง Instance จาก NormalTank อยู่จำนวน 2 คัน เพื่อทำการต่อสู้กัน โดยควรต้องมีบทบาทดังนี้

#### สร้างรถถัง A และ B ให้มีค่าพลังเบื้องต้นดังต่อไปนี้

|  |  |  |
| --- | --- | --- |
| ค่าสถานะ | รถถัง A | รถถัง B |
| HP | 200 | 250 |
| Str | 12 | 8 |
| Vit | 9 | 10 |
| BaseDamage | 11 | 10 |

#### รถถังทั้ง A และ B ผลัดกันโจมตีซึ่งกันและกัน เพื่อมุ่งหวังให้ค่าพลังของฝั่งตรงกันข้ามลดลงจนค่า HP = 0

#### รายละเอียดของพลังการโจมตีสามารถคำนวณได้ตามสมการดังต่อไปนี้

DamagePoint = MyTank\_BaseDamage \* Floor( MyTank\_Str / Enermy\_Vit ) \* Random( 0.7, 0.9 )

#### แสดงผลการทำงานผ่าน Console เพื่อให้เห็นรายละเอียดค่าพลังปัจจุบันของรถถังแต่ละคัน พลังการโจมต่อ ณ ขณะนั้น จนกว่าจะมีรถถังคันใดคันหนึ่งมีค่า HP = 0

|  |
| --- |
| โค้ดโปรแกรมภายใน Abstract Class |
|  |

|  |
| --- |
| โค้ดโปรแกรมภายใน NormalTank |
|  |

|  |
| --- |
| โค้ดโปรแกรมภายในฟังก์ชันการทำงานหลัก |
|  |

|  |
| --- |
| ผลลัพธ์การทำงานของโปรแกรม |
|  |

### เปลี่ยน Abstract Class ให้กลายเป็น Interfaces และเปรียบเทียบผลลัพธ์การทำงานของโปรแกรม

|  |
| --- |
| หลังจากเปลี่ยน Abstract Class เป็น Interface แล้ว เกิดอะไรขึ้นอย่าง? อธิบายพร้อมยกตัวอย่างประกอบให้ชัดเจน |
|  |

## สรุปผลการปฏิบัติการ

จากการทดลองพบว่าเมื่อเราเปลี่ยนมาเป็นแบบใช้ interface จะยุ่งยากและสบสัน

## คำถามท้ายการทดลอง

### เมื่อใดจึงควรเลือกใช้งาน Abstract Class

เมื่อเวลาที่เราต้องการจะใช้ตัวแปรนั้นหลายๆรอบแต่ค่าจะไม่เหมือนกัน

### เมื่อใดจึงควรเลือกใช้งาน Interface

เมื่อเวลาที่เราต้องการจะใช้ตัวแปรนั้นหลายๆรอบแต่ค่าจะไม่เหมือนกัน และ เราต้องมากำหนดค่าอีกรอบนึง