

## แบบฝึกปฏิบัติครั้งที่ 2

### การสร้างคลาสและการใช้งาน

#### โครงสร้างคลาส

โครงสร้างของคลาส ประกอบด้วย

1. แอททริบิวต์ : เป็นตัวแปรของคลาส เป็นตัวแปรที่ทั้งคลาสจะมองเห็นและสามารถใช้งานได้ ไม่ว่าจะอยู่ที่เมธอดใดภายในคลาส
2. คอนสตรัคเตอร์ : เมธอดที่มีชื่อเดียวกับชื่อคลาส ทำหน้าที่ในการกำหนดค่าให้กับแอททริบิวต์ของอ็อบเจ็กต์ที่สร้างขึ้น
3. เมธอด : ความสามารถหรือการกระทำต่าง ๆ ที่เกิดขึ้นกับอ็อบเจ็กต์ของคลาส

#### การเรียกใช้งานแอททริบิวต์และเมธอด

การเรียกใช้งาน จะมีอยู่ 2 ลักษณะ คือการเรียกใช้งานผ่านอ็อบเจ็กต์และการเรียกใช้งานผ่านคลาส ทั้งนี้ขึ้นอยู่กับรูปแบบในการสร้างเมธอดและแอททริบิวต์ของคลาสนั้น ๆ

การเรียกใช้งานผ่านอ็อบเจ็กต์ จะต้องทำการสร้างอ็อบเจ็กต์ของคลาสนั้น ๆ ก่อน หลังจากนั้นหากต้องการค่าแอททริบิวต์หรือเรียกใช้เมธอดของคลาส จะทำการเรียกโดยอ้างอิงจากชื่ออ็อบเจ็กต์ที่สร้างขึ้น เช่น การเรียกใช้ Scanner จะต้องทำการสร้างอ็อบเจ็กต์ของคลาส Scanner ก่อน แล้วจึงเรียกใช้คำสั่งในการรับค่าประเภทต่าง ๆ ได้จากอ็อบเจ็กต์ที่สร้างขึ้น

การเรียกใช้งานผ่านคลาส หากต้องการค่าแอททริบิวต์หรือเรียกใช้เมธอดของคลาส จะทำการเรียกใช้โดยอ้างอิงจากชื่อคลาส เช่น การใช้งานคลาส Math หากต้องการค่า PI จะเรียกโดยใช้คำสั่ง Math.PI หรือหากต้องการเรียกเมธอด pow จะใช้คำสั่ง Math.pow(a,n)

#### โครงสร้างแอททริบิวต์และเมธอด

รูปแบบคำสั่งในการสร้างแอททริบิวต์

```
[modifier] attributeType attributeName [= value] ;
```

รูปแบบคำสั่งในการสร้างเมธอด

```
[modifier] returnType methodName ([parameterList]) {  
    // statements  
}
```

modifier : คีย์เวิร์ดต่าง ๆ ที่ใช้ในการกำหนดรูปแบบการเรียกใช้งานและขอบเขตของการเรียกใช้

attributeType : ชนิดของแอตทริบิวต์ที่สร้างขึ้น

attributeName: ชื่อของแอตทริบิวต์

returnType : ชนิดของข้อมูลที่จะได้รับเมื่อทำการเรียกใช้เมธอดนี้

methodName : ชื่อของเมธอด

parameterList : ตัวแปรที่ต้องส่งมาเมื่อมีการเรียกใช้เมธอด (จะมีหรือไม่ก็ได้)

**modifier** จะมีอยู่ 2 กลุ่ม คือ

กลุ่มที่ 1 ระบุขอบเขตของการเรียกใช้ จะมีคีย์เวิร์ดที่ใช้งานกันอยู่ 3 คำ คือ

	ภายในคลาส	แพ็คเกจเดียวกัน	subclass	ข้ามแพ็คเกจ
public	x	x	x	x
protected	x	x	x	
private	x			
ไม่ระบุ	x	x		

กลุ่มที่ 2 ระบุรูปแบบของการเรียกใช้ จะใช้คีย์เวิร์ด static เพื่อระบุรูปแบบการเรียกใช้เป็นการเรียกใช้งานผ่านอ็อบเจกต์ หรือ เรียกใช้งานผ่านคลาส

	เรียกใช้ผ่านอ็อบเจกต์	เรียกใช้ผ่านคลาส
ระบุคำว่า static	x	x
ไม่ระบุคำว่า static	x	

เช่น

```
public static int max (int a, int b)
```

```
public int max (int a, int b)
```

จาก 2 คำสั่งที่เห็นจะทำหน้าที่ในการหาค่า max ระหว่าง a และ b เหมือนกัน แต่รูปแบบการเรียกใช้จะแตกต่างกัน คือ คำสั่งล่างจะต้องทำการสร้างอ็อบเจกต์ขึ้นก่อน จึงจะสามารถเรียกใช้ได้ โดยผ่านอ็อบเจกต์ดังกล่าว

คำสั่ง ให้นักศึกษาทำการเขียนโปรแกรมภาษาจาวา โดยมีรายละเอียดดังนี้

1. ทำการสร้างคลาส Circle ซึ่งประกอบด้วยเมธอดต่าง ๆ ดังนี้
  - 1.1. area : หาพื้นที่วงกลม
  - 1.2. circumference : หาเส้นรอบวง
  - 1.3. volume : หาปริมาตรทรงกลม
  - 1.4. getRadius : เรียกค่า radius ของวงกลม
  - 1.5. setRadius : กำหนดค่า radius ของวงกลม
2. ทำการสร้างคลาส testCircle
  - 2.1. ทำการรับค่า r จากผู้ใช้ แล้วทำการสร้างอ็อบเจกต์ของคลาส Circle
  - 2.2. คำนวณหาค่าพื้นที่วงกลมพร้อมแสดงผล
  - 2.3. คำนวณหาค่าเส้นวงกลมพร้อมแสดงผล
  - 2.4. คำนวณหาค่าปริมาตรทรงกลมพร้อมแสดงผล
  - 2.5. แสดงค่า r
  - 2.6. ทำการรับค่า r เพื่อ set ค่าใหม่
  - 2.7. คำนวณหาค่าพื้นที่วงกลมพร้อมแสดงผล
  - 2.8. คำนวณหาค่าเส้นวงกลมพร้อมแสดงผล
  - 2.9. คำนวณหาค่าปริมาตรทรงกลมพร้อมแสดงผล
  - 2.10. แสดงค่า r

---

### วิธีการส่งงาน

1. ให้นักศึกษาทำการ export project โดยตั้งชื่อเป็น **รหัสนักศึกษา**
2. ทำการอัปโหลดไฟล์ส่งทาง google classroom โดยนักศึกษาต้อง **ส่งให้ถูก section** ไม่เช่นนั้นจะไม่ได้คะแนน
3. เขียนคอมเมนต์ แจ้งชื่อและรหัสนักศึกษา
4. กรณีที่นักศึกษาส่งงานไม่ทันกำหนดเวลา นักศึกษาจะโดนหักคะแนนตามระยะเวลาที่ส่งช้า โดยการส่งงานเลท จะรับไม่เกิน 2 วันหลังจากครบกำหนด
5. กรณีที่นักศึกษาล่า จะต้องรีบติดต่อส่งงานก่อนคาบเรียนถัดไป โดยนักศึกษาจะได้รับการตรวจให้คะแนนเมื่อนักศึกษาทำการส่งใบลาเรียบร้อยแล้ว สำหรับการส่งใบลา นักศึกษาจะต้องนำมาส่งภายในคาบเรียนถัดไปที่นักศึกษากลับมาเรียน