

ใบงานการทดลองที่ 4

เรื่อง การกำหนดวัตถุ การใช้วัตถุ การสืบทอด และการห่อหุ้ม

1. จุดประสงค์ทั่วไป

- 1.1. รู้และเข้าใจหลักการเขียนโปรแกรมเชิงวัตถุ คลาส การกำหนด และการใช้วัตถุ
- 1.2. รู้และเข้าใจหลักการสืบทอด และการห่อหุ้มวัตถุ

2. เครื่องมือและอุปกรณ์

เครื่องคอมพิวเตอร์1 เครื่อง ที่ติดตั้งโปรแกรม Eclipse

3. ทฤษฎีการทดลอง

3.1. คลาสคืออะไร? มีลักษณะเด่นเป็นอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= Class คือ หมวดหมู่ เป็นแผนผังหรือเป็นตัวกำหนดส่วนประกอบต่างๆ มีจุดเด่นคือสามารถที่จะนำไปสร้างและแยกประเภท Object โดยใช้กับสิ่งใดก็ได้ ซึ่งภายในจะประกอบไปด้วย Field (เมื่อกำหนดค่าเกินกว่าที่ตัวแปรจะรับได้) , attribute และ method

เช่น สินค้า(class) มีส่วนประกอบคือ ชื่อสินค้า(object/attribute), ราคา(object/attribute), จำนวนสินค้า(object/attribute) เป็นต้น

3.2. วัตถุคืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

= Object คือ วัตถุที่สร้างขึ้นมาจาก Class หรือ class สำเร็จรูป เรียกได้อีกอย่างว่า instance ซึ่งภายในobjectจะประกอบด้วย Attribute (คุณสมบัติ) กับ Method (การตอบสนอง) โดยสามารถสืบทอดตัวแปรและฟังก์ชันจาก class ได้ด้วย

เช่น รถยนต์ มี attribute คือ สี, รุ่น, ขนาด

มี method คือ บ่นได้, เบรคได้

3.3. คุณสมบัติ(Properties/Attributes) ควรมีลักษณะการประกาศอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= การประกาศโดยการกำหนด type data ขึ้นมาแล้วตามด้วยชื่อของ attribute นั้น โดยการกำหนดค่าจะใส่ไว้
หลังเครื่องหมาย “=”

```
int x = 5 ;
```

-data type : integer

-attribute name : x

-initial value : 5

3.4. การกระทำ/ฟังก์ชัน/เมธอด(Method) ควรมีลักษณะการประกาศอย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= ประกาศโดย กำหนดสถานะการเข้าถึงการใช้งาน(public/private) ตามด้วยชื่อ และparameters ซึ่งเป็นlist ของ
ตัวแปรที่จะนำไปใช้ในmethod

เช่น type name (parameter1, parameter2, ...) {

statements

}

access_modifier type name (parameter1, parameter2, ...) {

statements

}

3.5. เพราะเหตุใดจึงควรสร้าง 1 คลาสต่อ 1 ไฟล์?

= เพราะในการ อ้างอิงหรือการหา class จะต้องการ class main เพียง 1 อัน หากต้องการอ้างอิงหรือเรียกใช้จะทำให้การคอมไพล์ผิดพลาด

3.6. เมื่อสร้างวัตถุขึ้นมาแล้ว วัตถุจะสามารถอ้างอิง Properties หรือ Method ได้ด้วยวิธีการใด ?

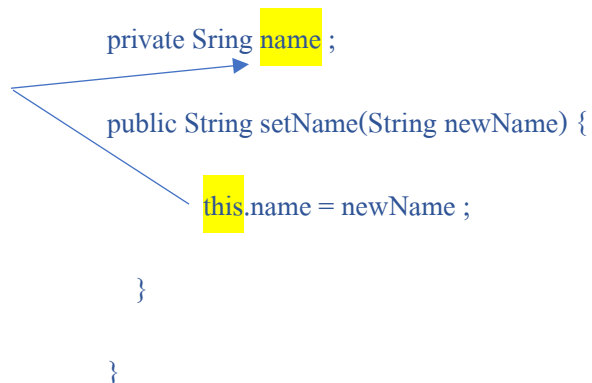
= อ้างอิงถึงได้โดยการ ระบุชื่อคลาสพร้อมกับระบุ object ที่ต้องการเข้าถึง โดยจะใช้ new ในการรับค่าด้วย

3.7. คำสั่ง this มีหน้าที่อย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= this. มีหน้าที่ในการกล่าวถึงclassนั้นๆ หรือก็คือ attribute name ที่อยู่ในคลาสนั้นๆ

เช่น public class person {

```
    private String name ;  
    public String setName(String newName) {  
        this.name = newName ;  
    }  
}
```



นั่นคือ นำค่า newname ที่ได้รับมาส่งไปที่ Class นี้ซึ่งก็คือ name

3.8. Constructor Method มีหน้าที่อย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= โดย Constructor Method จะเป็น Method ที่มักจะใช้เซตค่าเริ่มต้นเมื่อเราทำการเรียก Class นั้นๆ

```
Static void Main( string[] args )
```

```
{
```

```
    Product book, cup ;
```

```
    book = new Product();
```

จะเป็นการเรียกใช้งาน Constructor

```
}
```

3.9. Destructor Method มีหน้าที่อย่างไร? อธิบายพร้อมยกตัวอย่างประกอบ

= Destructor Method นั้นเป็น Method ที่ให้ Class ทำอะไรบางอย่างก่อนที่ Class นั้นจะถูกทำลายลง

```
Public Class Car
```

```
    Protected overrides Sub Finalize()
```

```
        MessageBox.Show("ยกเลิกการใช้รถจักรยานยนต์เรียบร้อยแล้ว")
```

```
    End Sub
```

```
End Class
```

3.10. การสืบทอด(Inheritance) คืออะไร? มีข้อดีและข้อเสียอย่างไร?

= การสืบทอด เราสร้างคลาสใหม่จากคลาสเดิมที่มีอยู่แล้ว โดยมีการถ่ายทอดคุณสมบัติของคลาสหลักมา เหมือนการถ่ายทอด DNA จากพ่อแม่สู่ลูก ใน java นั้น จะสามารถสืบทอด attribute และ methods จาก Class หนึ่ง ไปสู่อีกคลาสหนึ่งได้โดยจะแบ่งเป็น subclass (child : คลาสที่สืบทอด) และ superclass (parent : คลาสที่ถูกสืบทอด) โดยมีข้อยกเว้นว่า สมาชิกที่มีการเข้าถึงเป็น private และ constructor จะไม่ถูกสืบทอดไปด้วย

ข้อดี

-Class ลูกสามารถเรียกใช้งานคำสั่งต่าง ๆ จาก Class พ่อแม่ ได้โดยทันทีโดยไม่ต้องเขียนโปรแกรมใหม่

- ทำให้ Class ต่าง ๆ ที่มีลักษณะเหมือน ๆ กันทำการสืบทอดคุณสมบัติไปได้ และใช้งานได้โดยทันที
- เพื่อจัดกลุ่ม และทำให้การออกแบบ Class มีประสิทธิภาพมากยิ่งขึ้น

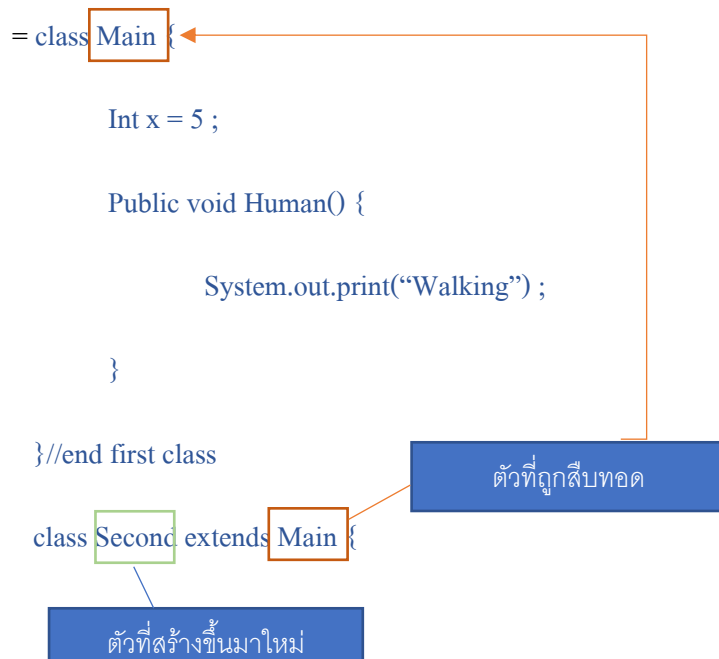
ข้อเสีย

-การทำ Inheritance ที่ถูกหลัก มันทำยาก เพราะต้องเข้าใจและแยกความสัมพันธ์แบบ IS A กับ HAS A ได้อย่างถูกต้อง ถึงจะมีโอกาสออกแบบได้ถูก ซึ่งในโจทย์จริงบางทีมันไม่ได้แยกกันได้ง่ายเหมือนในตัวอย่าง

-เปลี่ยนแปลงแก้ไขยาก ถ้าโค้ดเราใช้ Inheritance กับคลาสหลายๆตัว แต่พบว่า ไม่ควรทำโครงสร้างแบบนั้น หรือ อยากแก้ไขโครงสร้างพวกนั้นล่ะ มันอาจจะต้องไปไล่แก้ไขเป็นร้อยๆคลาสเลยก็เป็นได้ แล้วถ้าเกิดเราทำเป็น Library ไว้ แล้วคนที่เอาไปใช้ต่อนี้ต้องมากอยไล่แก้ตามโครงสร้างใหม่

-คนเอาไปใช้ต่อเอาไปใช้ยาก ถ้าเราจะต้องไปใช้คลาสที่คนอื่นเขียนไว้เราใช้ได้เลยไหม? ซึ่งคำตอบส่วนใหญ่ก็คือต้องไปทำความเข้าใจมันก่อน แต่มันจะยิ่งยากขึ้นหลายเท่าเมื่อมันเป็น โครงสร้างแบบ Inheritance

3.11. จงยกตัวอย่างการสร้างคลาสรองเพื่อทำการสืบทอดจากคลาสหลัก



3.12. จงยกตัวอย่างการสร้างวัตถุของคลาสหลักและคลาสรอง พร้อมกับยกตัวอย่างการเรียกใช้งานวัตถุในแต่ละคลาส เพื่อให้เห็น ภาพการสืบทอดการทำงานซึ่งกันและกัน

```
= Public class Main {  
  
    Public void fullThrottle (){  
        System.out.println("The car is goin as fast as it can!");  
    }  
  
    Public void speed(int maxSpeed){  
        System.out.println("Max Speed is : " + maxSpeed );  
    }  
}  
  
Class Second  
  
    Public static void main (String[] args){  
        Main myCar = new Main();  
        myCar.fullThorttle();  
        myCar.Speed(200);  
    }  
}
```

3.13. การควบคุมระดับการเข้าถึง(Access Modifier) ของตัวแปรแบบ Public, Protected และ Private คืออะไร ?

= Public คือ class นั้นจะถูกเรียกใช้งานได้จากทุกส่วนในโปรแกรม

Protected คือ คลาสหรือสมาชิกสามารถเข้าถึงได้ภายใน package เดียวกันและ Sub class เท่านั้น

Private คือ คลาสหรือสมาชิกสามารถเข้าถึงได้ภายในคลาสเดียวกันเท่านั้น

3.14. การห่อหุ้ม(Encapsulation) คืออะไร? อธิบายพร้อมยกตัวอย่างประกอบ

= คือการปกปิดหรือควบคุมการเข้าถึงข้อมูลของออบเจกต์จากภายนอก ซึ่งวิธีการที่ใช้ในการควบคุมคือการทำงานผ่าน Method เพื่อดำเนินการกับข้อมูลนั้น

4. ลำดับขั้นการปฏิบัติการ

4.1. จงเขียนโปรแกรมสร้างคลาสในการจัดการอาเรย์ดังต่อไปนี้

4.1.1. สร้างคลาสชื่อว่า MyClassicalArray

มี Properties ชื่อว่า MyArray[] พร้อมกับสุ่มค่าไว้ภายในตัวแปรทั้งหมด 10 ค่า

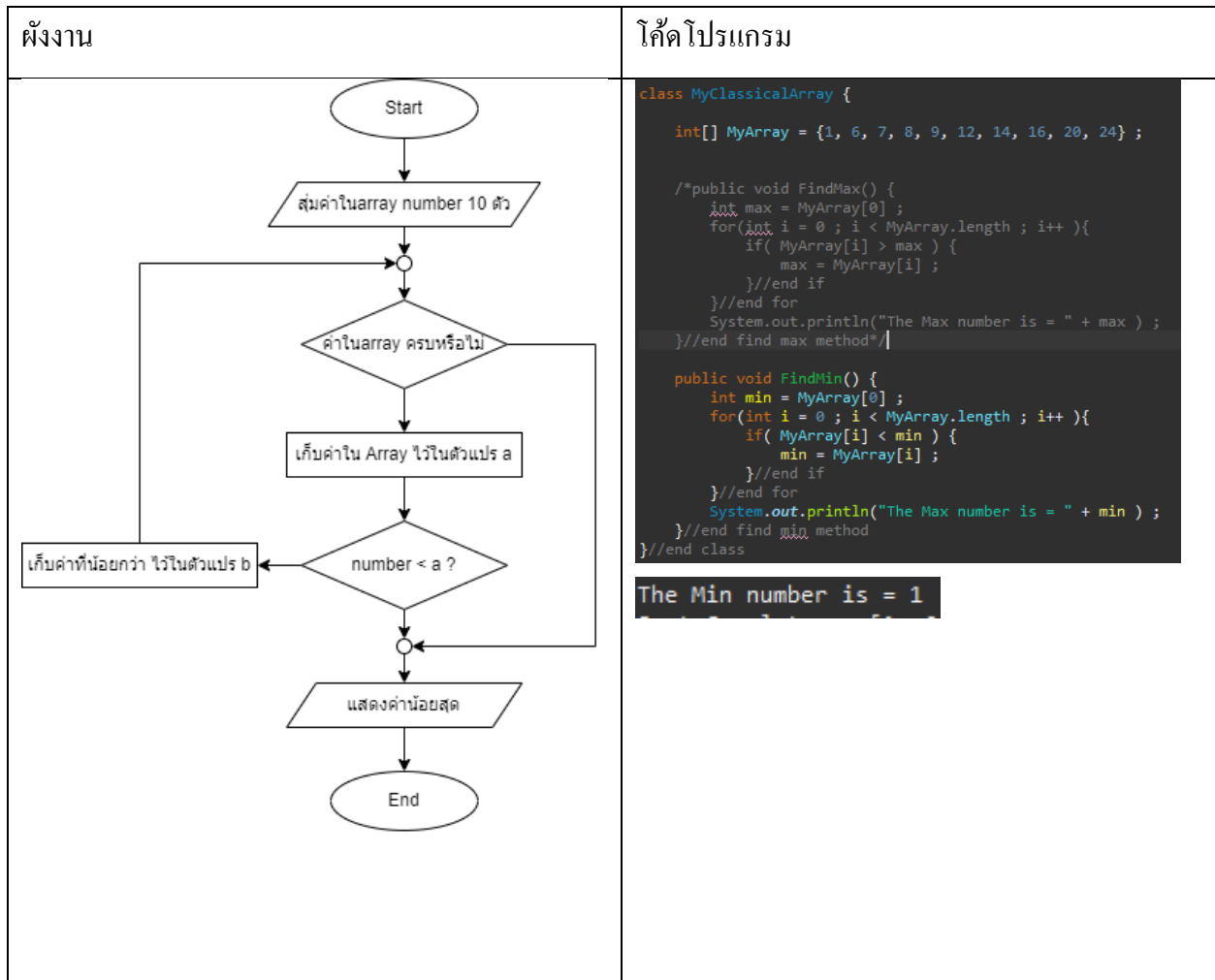
มี Method ชื่อว่า FindMax() ; เพื่อหาค่าที่มากที่สุดที่อยู่ในตัวแปร MyArray

มี Method ชื่อว่า FindMin() ; เพื่อหาค่าที่น้อยที่สุดที่อยู่ในตัวแปร MyArray

Method : FindMax() ;

ผังงาน	โค้ดโปรแกรม
	<pre>class MyClassicalArray { int[] MyArray = {1, 6, 7, 8, 9, 12, 14, 16, 20, 24} ; public void FindMax() { int max = MyArray[0] ; for(int i = 0 ; i < MyArray.length ; i++){ if(MyArray[i] > max) { max = MyArray[i] ; } } System.out.println("The Max number is = " + max) ; } }</pre> <p>The Max number is = 24</p>

Method : FindMin() ;

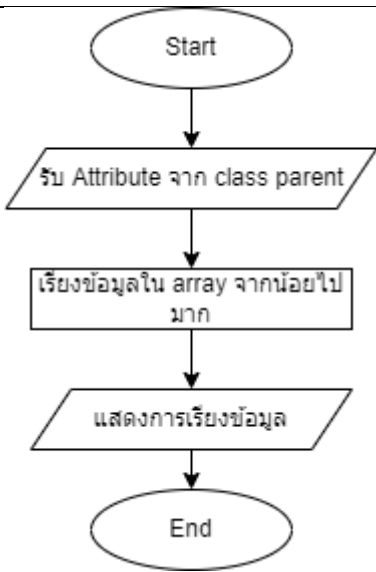


4.1.2. สร้างคลาสชื่อว่า MyCurrentArray ที่สืบทอดคลาส MyClassicalArray

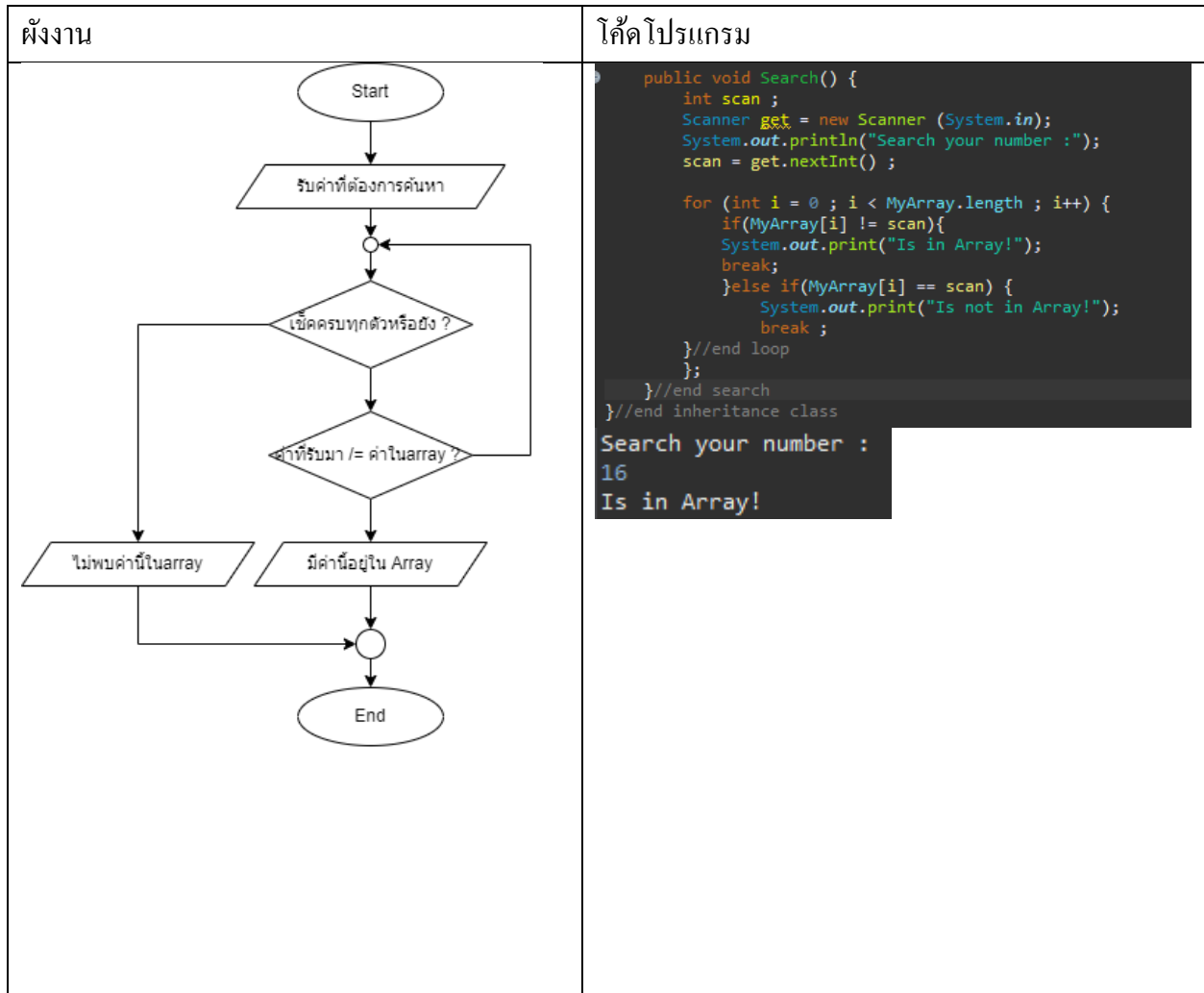
มี Method ชื่อว่า Sort() ; เพื่อเรียงค่าภายในตัวแปร MyArray จากน้อยไปมาก

มี Method ชื่อว่า Search(Find) ; เพื่อค้นหาค่าที่อยู่ในตัวแปร MyArray

Method : Sort() ;

ผังงาน	โค้ดโปรแกรม
 <pre>graph TD; Start([Start]) --> Receive[/รับ Attribute จาก class parent/]; Receive --> Sort[เรียงข้อมูลใน array จากน้อยไปมาก]; Sort --> Display[/แสดงการเรียงข้อมูล/]; Display --> End([End]);</pre>	<pre>class MyCurrentArray extends MyClassicalArray{ public void Sort () { Arrays.sort(MyArray); System.out.println("Sort Complete : " + Arrays.toString(MyArray)); } //end sort }</pre> <pre>Sort Complete : [1, 6, 7, 8, 9, 12, 14, 16, 20, 24]</pre>

Method : Search(Find) ;



4.1.3. ในฟังก์ชันหลัก สร้างวัตถุจากคลาส MyClassicalArray ขึ้นมา และทดสอบการใช้งานคำสั่ง FindMax() ;
และคำสั่ง FindMin() ;

4.1.4. ในฟังก์ชันหลัก สร้างวัตถุจากคลาส MyCurrentArray ขึ้นมา และทำการทดสอบการใช้งานคำสั่ง
FindMax() ; คำสั่ง

FindMin() ; คำสั่ง Sort() ; และคำสั่ง Search(Find) ;

โค้ดโปรแกรมภายในฟังก์ชันหลัก

```
public static void main(String[] args) {  
    // TODO Auto-generated method stub  
    MyClassicalArray getvalue = new MyClassicalArray();  
    MyCurrentArray sort = new MyCurrentArray();  
    MyCurrentArray search = new MyCurrentArray();  
  
    getvalue.FindMax() ;  
    getvalue.FindMin() ;  
    sort.Sort() ;  
    search.Search() ;  
  
} //end main  
} //end class
```

ผลลัพธ์การทำงานของโปรแกรม

```
The Max number is = 24  
The Min number is = 1  
Sort Complete : [1, 6, 7, 8, 9, 12, 14, 16, 20, 24]  
Search your number :  
16  
Is in Array!
```

5. สรุปผลการปฏิบัติการ

= ในการใช้งาน หลายๆคลาสนั้นจะต้องเรียนรู้ในส่วนของสิทธิ์การเข้าถึง private public protected ในการจะสร้างคลาสเพิ่ม หากใช้เป็นการสืบทอด จะได้รับข้อมูลในส่วน of method และ attribute ไปด้วย

6. คำถามท้ายการทดลอง

6.1. การสืบทอดในภาษาจาวาสามารถทำได้โดยใช้คำสั่งใด ?

= ประกาศ Class ใหม่ แล้วใช้ extend ตามด้วยชื่อของ class ที่จะสืบทอด

Class Study extends Student

6.2. จงอธิบายข้อควรระวังในการใช้งาน public, private และ protected

= ต้องดูการใช้งานร่วมให้ดี การเลือกใช้สิทธิ์การเข้าถึง หากใช้ไม่ถูกต้องกับการทำงานของข้อมูลจะส่งผลให้เกิดการผิดพลาด

6.3. วัตถุ และ คลาส มีความเหมือนหรือแตกต่างกันอย่างไร ?

= เหมือนกัน เพียงแต่ว่า object นั้น เป็นองค์ประกอบหรือรายละเอียดที่สืบทอดไปของ class

6.4. ในฐานะที่เป็นผู้พัฒนาระบบ คุณจะเลือกใช้การสืบทอดคลาสเมื่อใด? เพราะเหตุใด ?

= เมื่อต้องการใช้งานข้อมูลหรือนำข้อมูลเดิมมาพัฒนาเพิ่มเติม เพราะการสืบทอดจะนำข้อมูลการทำงานใน class ที่ถูกสร้างไว้ก่อนมาใช้งานได้ด้วย