

Software Construction

Midterm Lecture

Ed. 1

20 / 09 / 2016

Contents

- ~~- Type Conversion~~
- Data type (Primitive, Type wrapper, Reference)
- all parameter
- Constructor
- Pre/Post Condition
- Passing Parameter
- **Access modify**
- class Object : equal, toString
- Immutator class (Accessor, Mutator)
- Side Effect (Call by Value, Call by Reference)
- ~~- Overloaded method~~
- **abstract class**
- Inherit
- Associate
- Encapsulation
- Polymorphism

Part I : Data type

Data type – ทุกๆข้อมูลใน java จะต้องมี type เป็นของตัวเอง ซึ่ง type จะบอกรูปแบบการเก็บข้อมูล และความสามารถของข้อมูลนั้นๆ แบ่งออกเป็น 2 ประเภท

Primitive Data Type – เก็บเฉพาะข้อมูล ไม่สามารถใช้งาน method ได้

- มีทั้งหมด 8 แบบ 4จำนวนเต็ม 2จำนวนจริง 1อักขระ 1ค่าความจริง
- **Type Wrapper** – เนื่องจาก primitive ไม่สามารถเรียกใช้ method ได้ java จึงมีคลาสพิเศษซึ่งจะคอยทำงานกับ primitive type เรียกคลาสนี้ว่า Type Wrapper

Data type	Size	Type Wrapper	เว้นไว้เพื่อจุด
byte	1 byte	Byte	
short	2 byte	Short	
int	4 byte	Integer	
long	8 byte	Long	
float	4 byte	Float	
double	8 byte	Double	
char	2 byte	Character	
boolean	1 bit	Boolean	

Reference Data Type (Object Data Type)

- เก็บทั้งข้อมูล และ เมธอด
- type ของข้อมูล คือ class

** String ถือเป็น Reference Type

Software Construction - Lecture
Access Modifier

Part II : Access Modifier

Access Modifier – key word เพื่อกำหนดคุณสมบัติของ variable method หรือ class

access modifier	Can use?			Option			
	class	method	variable	Class	Package	Subclass	World
public	○	○	○	○	○	○	○
default	○	○	○	○	○	X	X
protected	○	○	○	○	○	○	X
private	○	○	○	○	X	X	X
final	○	○	○	class : ไม่สามารถสืบทอดได้ (มีคลาสลูกไม่ได้) method : คลาสลูกไม่สามารถ override ได้ variable : ประกาศเป็น constant ไม่สามารถเปลี่ยนค่าได้			
abstract	○	○	X	class : ไม่สามารถสร้าง obj. ได้ method : เมธอดที่ไม่มีคำสั่งภายใน และจะบังคับให้คลาสลูก override			
static	X	○	○	method : สามารถเรียกใช้ผ่าน class ได้ variable : สามารถเรียกใช้ผ่าน class ได้			

Part III : Parameter

```
class A{
    public int a;
    public methodA(int y){
        int z = 2;
        a += z*y;
    }
}

class Start{
    public start(){
        int x = 3;
        A a1 = new A();
        a1.methodA(x);
    }
}
```

Formal Parameter

- ตัวแปรที่ประกาศไว้รอรับค่าเวลาถูกเรียก
- จากตัวอย่างคือ y

Actual Parameter

- ตัวแปรหรือค่าที่ส่งไปเมื่อเรียกเมธอด
- จากตัวอย่างคือ x

Locals Parameter

- ตัวแปรที่ถูกประกาศขึ้นภายในเมธอด
- จากตัวอย่าง x จะเป็น locals ของ start, z เป็น locals ของ method

Implicit Parameter

- ตัวแปรที่เป็น Instance variable ของคลาส ของเมธอดนั้น
- จากตัวอย่าง method อยู่ใน class A ตัวแปร a ที่เมธอดนี้เรียกมาใช้จะเป็น Implicit parameter

Explicit Parameter

- ตัวแปรที่เป็น Formal และ Actual (ไม่ค่อยชัวร์)

Part IV : Abstract Class

Abstract Class

- เป็นคลาสที่ไม่สามารถสร้าง obj. ได้
- มีไว้เพื่อเป็นแม่แบบของคลาสอื่นๆ
- เช่น คลาส Vehicle ที่เป็นคลาสแม่ของ Car, AirPlane ฯลฯ
- ถ้าเราสร้าง obj. ของ Car นั้นคือ เราจะได้รถ1คัน อันนี้คือเรื่องปกติ
- แต่ถ้าเราสร้าง obj. ของ Vehicle นั้นคือเราจะได้ยานพาหนะ แล้วมันคืออะไร?
- ที่จะบอกก็คือว่าคลาสที่ทำเป็น abstract คือคลาสที่ยังไม่สมบูรณ์ ขาดmethodหรือattribute บางอย่างที่จะทำให้ทำงานได้

Abstract Method

- เป็น method ที่สร้างไว้เฉยๆ ไม่มีคำสั่งภายใน ไม่มีปีกกา
- ex. `public abstract void method(int a);`
- คลาสที่เป็นคลาสลูก จะถูกบังคับให้ override abstract method ทุกเมธอดของคลาสแม่
- ยกเว้นถ้าคลาสลูกก็เป็น abstract class เหมือนกัน

