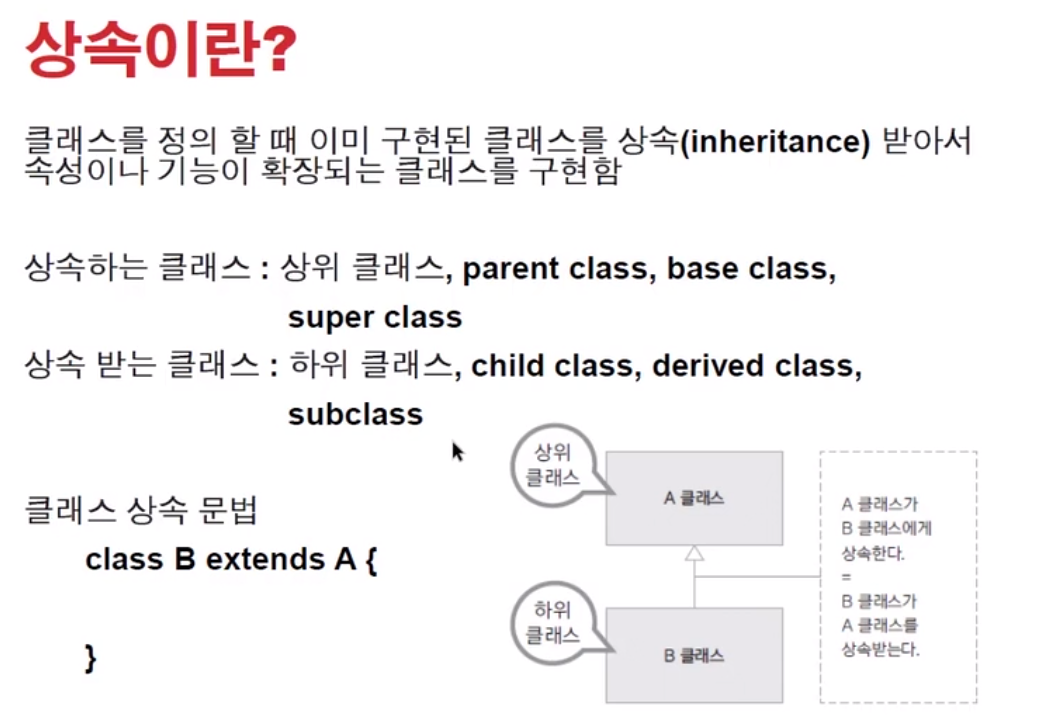
Java 상속과 다형성(Polymorphism)



**public** **class** Point {

**private** **int** x;

**private** **int** y;

**public** **int** getX() {

**return** x;

}

**public** **void** setX(**int** x) {

**this**.x = x;

}

**public** **int** getY() {

**return** y;

}

**public** **void** setY(**int** y) {

**this**.y = y;

}

}

**package** com.ajnet.inheritance;

**public** **class** Circle {

**private** **int** x;

**private** **int** y;

**private** **int** radius;

}

Point와 Circle Class의 관계는 일반화, 구체화인 상속의 관계가 아님.

따라서

**public** **class** Circle extends {

**private** **int** radius;

}

가 아니라

**public** **class** Circle {

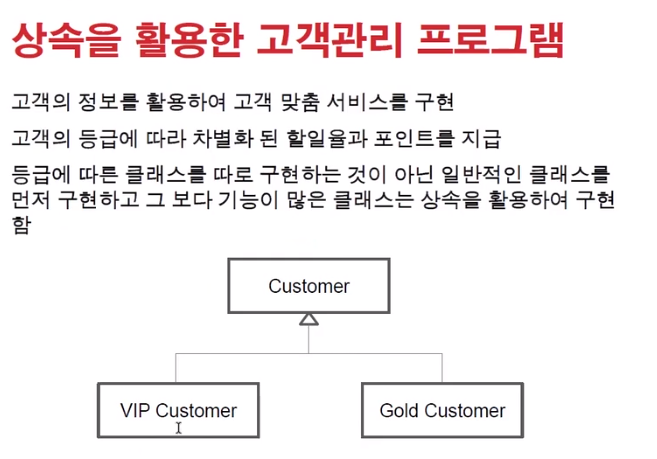
Point point;

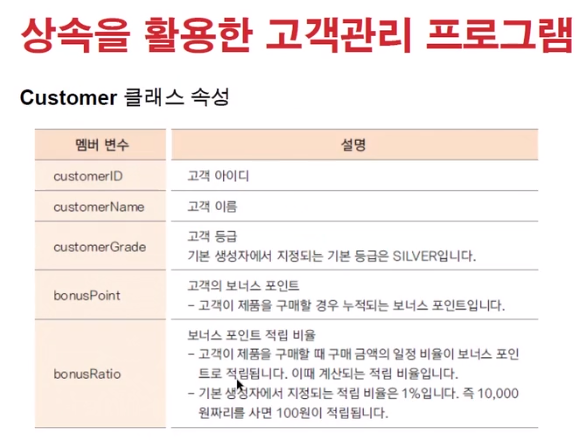
**private** **int** radius;

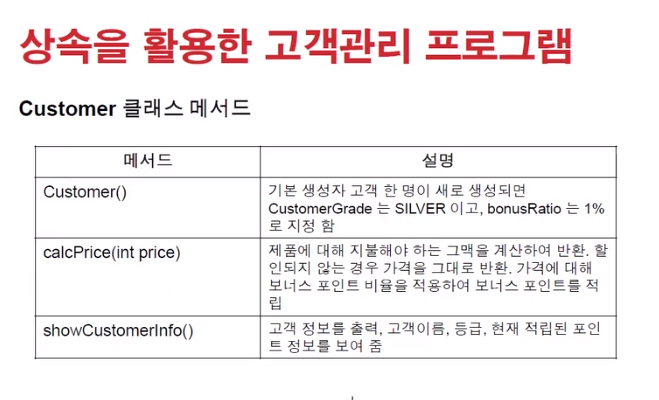
}

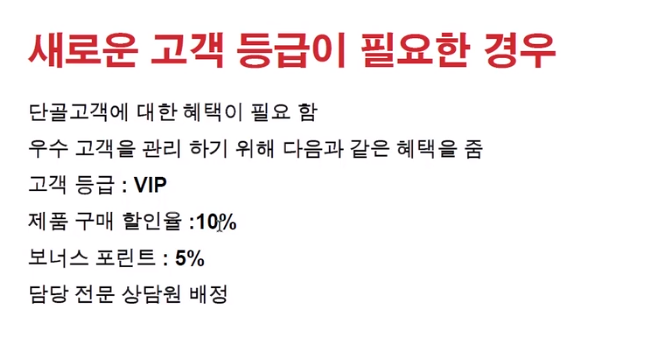
로 작성되는 것이 맞다.

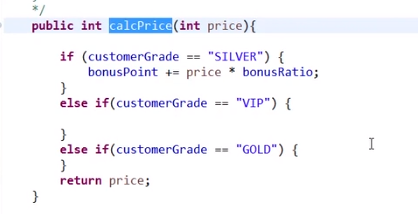
즉, Circle과 Point의 관계는 Has a 관계이며, 상속은 is a 의 관계이다











등급에 따라 if else 가 늘어나게 되어 유지보수가 어려워진다.

그래서 VIPCustomer라는 상속 class를 만든다.

public class VIPCustomer extends Customer{

private int agentID;

double saleRatio;

public VIPCustomer()

{

customerGrade = "VIP";

bonusRatio = 0.05;

saleRatio = 0.1;

// System.out.println("VIPCusomer() 생성자 호출");

}

public VIPCustomer(int customerID, String customerName, int agentID){

super(customerID, customerName);

customerGrade = "VIP";

bonusRatio = 0.05;

saleRatio = 0.1;

this.agentID = agentID;

// System.out.println("VIPCusomer(int, String) 생성자 호출");

}

public int calcPrice(int price){

bonusPoint += price \* bonusRatio;

return price - (int)(price \* saleRatio);

}

public int getAgentID(){

return agentID;

}

Variable default -> 같은 package에서만 가시성이 있음.

Scope : public private protected default