상속성

#01. 클래스 상속의 이해

클래스 간에 부모, 자식 관계를 설정하는 코딩 기법

부모 클래스 A를 자식 클래스 B가 상속받을 때, B는 **A의 모든 멤버변수와 메서드를 자신의 것으로 상속**받게 된다. (단, private으로 설정된 기능은 상속되지 않음)

B는 A의 public, protected 기능들을 직접적으로 코딩하지 않더라도 자신의 것으로 사용할 수 있게 된다.

1) 상속의 정의 방법

extends 키워드를 사용하여 부모 클래스의 이름을 명시한다.

```
public class 현재클래스이름 extends 부모클래스이름 { ... }
```

2) 기능의 확장을 위한 상속

원래 extends 라는 단어의 의미는 연장하다, 확장하다 라는 의미이다.

어떤 클래스 A가 다른 클래스 B를 상속받은 후 자신만의 기능을 구현했다면 **A의 기능을 B를 통해 확장했다** 고 할 수 있다.

계산 기능을 갖는 클래스간의 상속 실험

아래 예제에서는 CalcChild가 CalcParent의 원래 기능에 times()와 divide()를 확장했다고 볼 수 있다.

> CalcParent.java, CalcChild.java, Ex01_CalcTest.java

3) 공통 기능을 표현하기 위한 상속

두 개 이상의 클래스를 작성하면서 중복되는 기능이 있다면 하나의 부모 클래스를 통해 중복 코드를 최소화 할 수 있다.

아래의 예시에서 명시한 항목들을 포함하는 JavaBeans 클래스를 작성할 경우 모든 클래스가 num, subject, content, writer라는 멤버변수와 이에 대한 getter, setter를 포함해야 한다.

항목	공지사항 게시판	질문과 답변 게시판	자유 게시판
글 번호	num	num	num
제목	subject	subject	subject
내용	content	content	content
작성자	writer	writer	writer
첨부파일	file	(사용안함)	(사용안함)

항목	공지사항 게시판	질문과 답변 게시판	자유 게시판
추천수	(사용안함)	vote	(사용안함)
스크랩 수	(사용안함)	(사용안함)	scrapCount

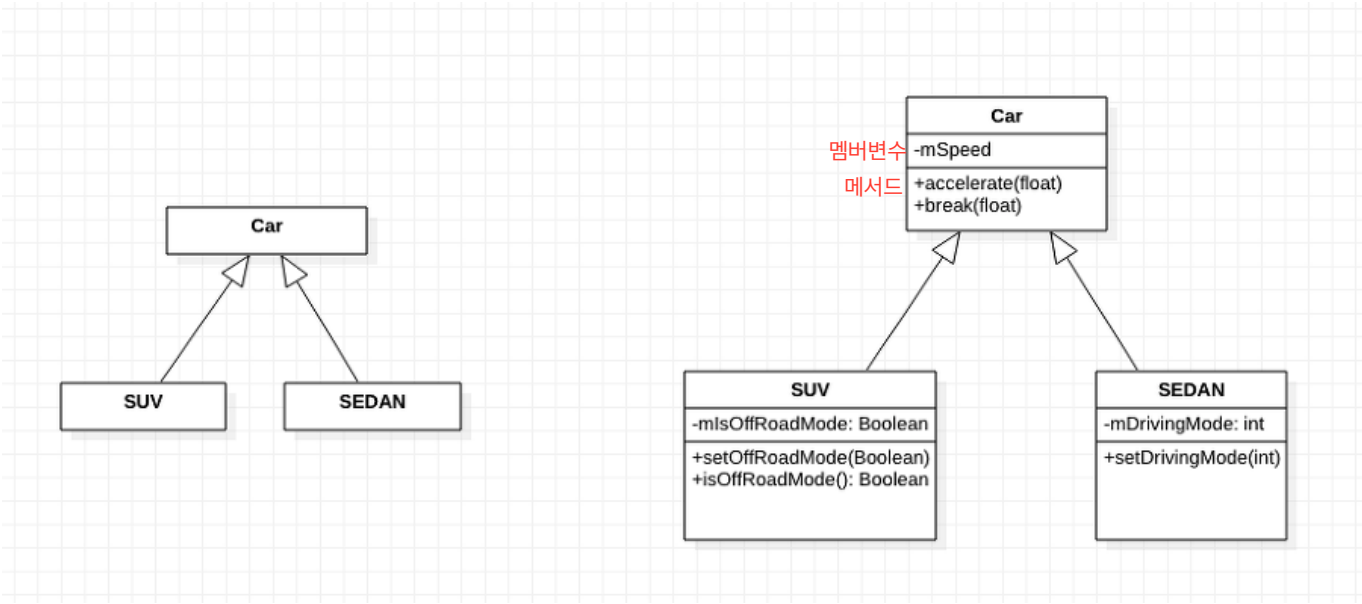
하지만 아래와 같이 "게시판"이라는 클래스를 하나 미리 정의하고 다른 클래스들이 이를 상속 받는다면 각각의 클래스는 자신만의 특성을 표현할 수 있는 고유한 값들만 추가하는 것으로 **중복 코드를 최소화** 할 수 있다.

항목	게시판(부모)	공지사항 게시판	질문과 답변 게시판	자유 게시판
글 번호	num	(게시판 상속)	(게시판 상속)	(게시판 상속)
제목	subject	(게시판 상속)	(게시판 상속)	(게시판 상속)
내용	content	(게시판 상속)	(게시판 상속)	(게시판 상속)
작성자	writer	(게시판 상속)	(게시판 상속)	(게시판 상속)
첨부파일		file	(사용안함)	(사용안함)
추천수		(사용안함)	vote	(사용안함)
스크랩 수		(사용안함)	(사용안함)	scrapCount

> Board.java, NoticeBoard.java, QnaBoard.java, FreeBoard.java, Ex02_BoardTest.java

#02. 클래스 다이어그램 (Class Diagram)

클래스 내부 구성 요소 및 클래스 간의 관계를 도식화 하여 시스템의 특정 모듈이나 일부 또는 전체 구조를 나타내는 설계도.



1) 클래스 다이어그램 작성 방법

접근 한정자

public은 +, private은 -, protected는 #으로 표현한다.

멤버변수

```
접근한정자기호_변수이름 : 데이터타입 = "기본값"
```

기본값이 필요하지 않을 경우 생략 할 수 있다. 예를 들어 private 형식의 기본값을 갖지 않는 문자열 변수 myName은 아래와 같다.

```
-myName : String
```

메서드

```
접근한정자기호_메서드이름(파라미터타입) : 리턴타입
```

파라미터와 리턴타입은 변수이름 없이 데이터 타입만 명시한다.

생성자

1. 생성자임을 의미하는 기호 ⓒ를 메서드 이름 앞에 적용

```
ⓒ HelloWorld(int, int)
```

2. 메서드 이름 위에 `<<constructor>>` 명시

```
<<constructor>>
HelloWorld(int, int)
```

앞 예제 Board 클래스에 대한 다이어그램

Board
-num: int -subject: String -content: String -writer: String
+getNum(): int +setNum(int): void +getSubject(): String +setSubject(String): void +getContent(): String +setContent(String): void +getWriter(): String +setWriter(String): void

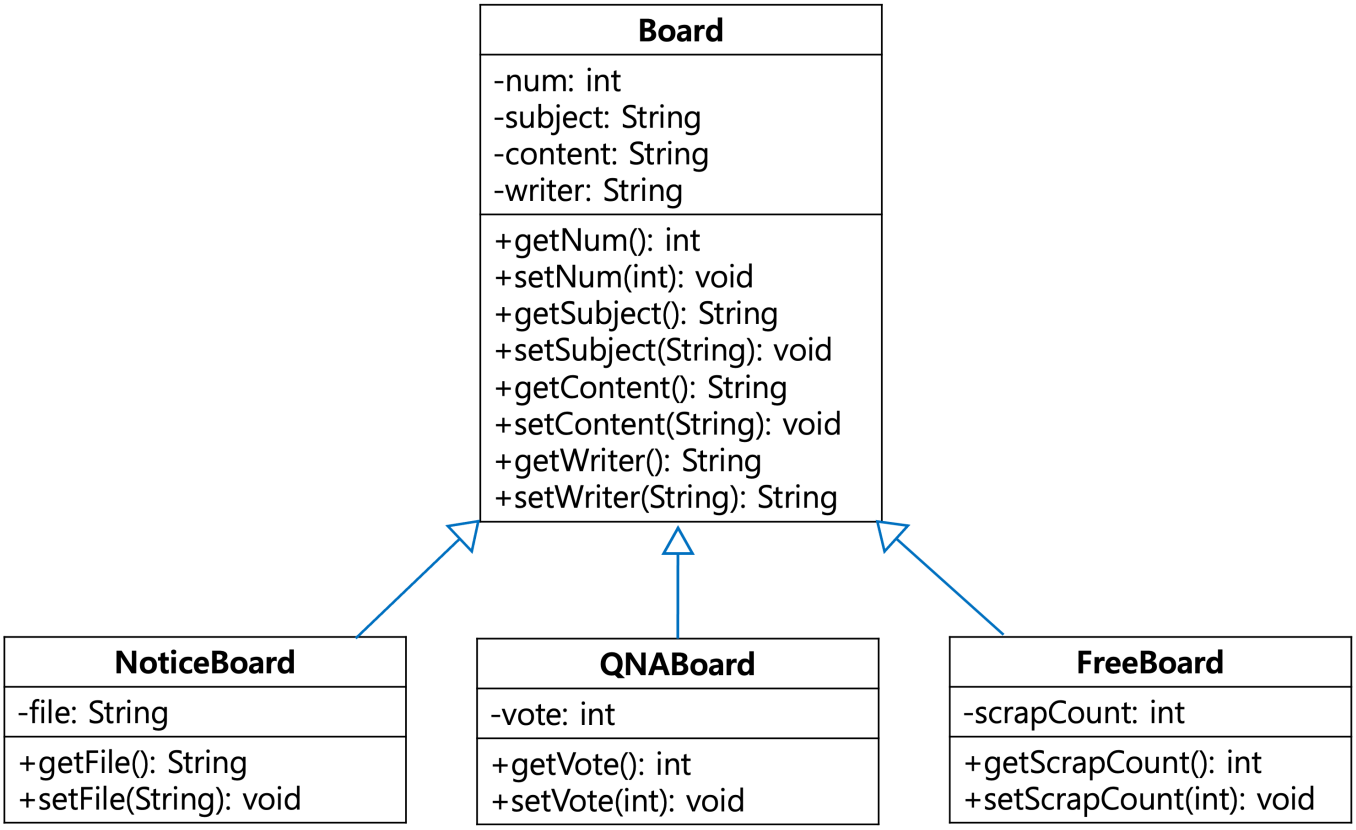
멤버변수

2) IS A 관계

클래스 Foo 가 클래스 Bar를 상속받을 때 Foo is a Bar라고 하고 자식 클래스가 부모를 가리키는 실선 형태의 화살표로 표현한다.

부모는 나다

앞 예제 Board 클래스와 NoticeBoard, QNABoard, FreeBoard 간의 관계



직선 화살표 is a
점선 화살표 has a

3) HAS A 관계

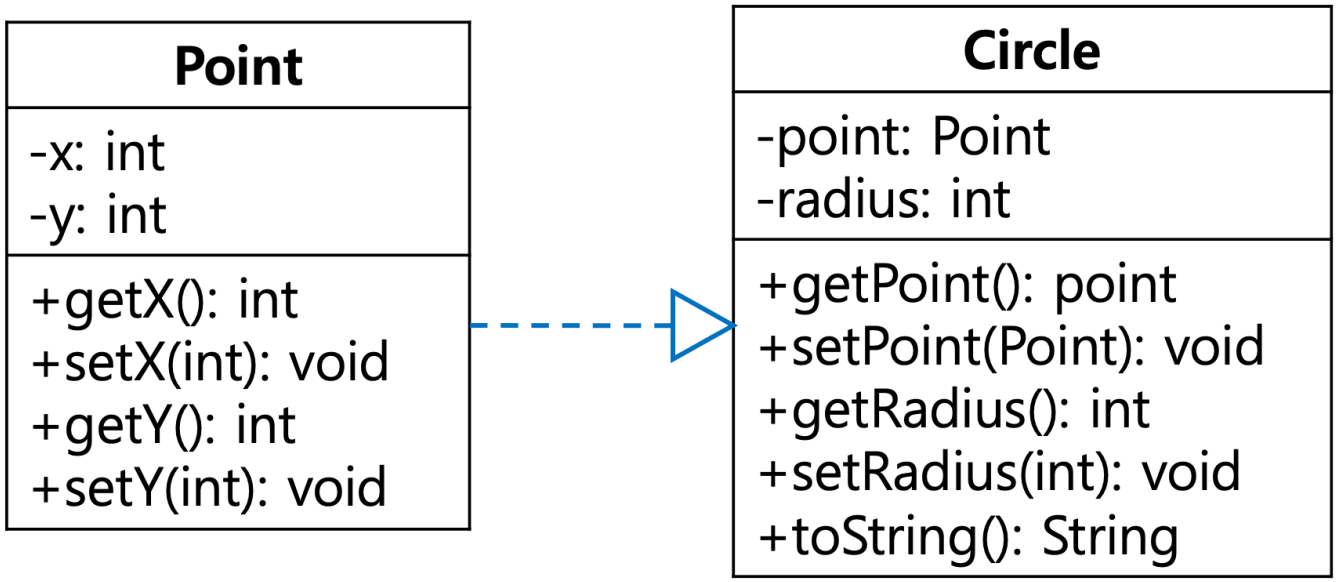
어떤 클래스 Foo 가 다른 클래스 Bar에 대한 **객체를 멤버변수로 갖고 있을 때**, Foo Has A Bar 로 표현하며 이는 객체간의 포함 관계를 의미한다.

원을 표현하는 클래스에 대한 예시

원은 중점(x, y)와 반지름으로 구성된다. 이 때 원의 중점 좌표를 Point 라는 별도 의 클래스로 표현한 경우

HAS A 관계는 점선을 갖는 화살표로 이루어 지며 포함되는 클래스가 포함하는 클래스를 가리킨다.

여기서는 Point 클래스가 Circle 클래스를 가리켜야 하고, Circle HAS A Point 관계가 된다.



> Point.java, Circle.java, Ex03_CircleTest.java