11.추상클래스

1.)추상클래스 이해

추상적으로 메소드를 만든 후 상속받은 하위클래스에게 오버라이딩을 강제하는 것 이다.

오버라이딩을 강제하는 것은 해당 매소드와 클래스를 사용하려면 반드시 재정의해야 한다는 것이다.

모듈처럼 중복되는 부분이나 공통적인 부분은 미리 만들어진 것을 사용하고, 이를 받아 사용하는 쪽에서는 자신에게 필요한 부분만을 재정의하여 사용함으로써 생산성과 효율성을 도모하기 하게된다.

Ex)

```
1  public abstract class HeadQuarterStore {
2      private String str;
3      public HeadQuarterStore(String str) {
4          this.str = str;
5      }
6      public abstract void kimchi();
7
8      public abstract void bude();
9
10     public abstract void bibim();
11
12     public abstract void sunde();
13
14     public abstract void gonggibab();
```

- 접근제어제와 메소드명 사이에 abstract 명령어를 정의한다.
- 해당 클래스에 추상(abstract)메소드가 하나라도 존재하면 해당 클래스는 추상(abstract) 클래스로 정의해야 한다.
- 반대로 추상클래스에는 추상매소드가 반드시 하나 이상 존재해야한다.
- 컨트롤 + 1 단축키로 에러의 방편과 자동완성기능을 사용할수있다.
- 추상클래스는 객체생성을 할 수 없다.
- $1클래스 생성 = $1은 시스템이 만든 클래스라는뜻
- 추상클래스내에는 상수, 변수, 생성자, 일반메소드, 추상메소드1개이상 이 존재할수있고
- 이와 다르게 인터페이스는 상수 추상메소드 만 가능하다.

Ex)

```java
public abstract class SuperClass {
    public abstract void method1();//상속받은 클래스에서 오버라이드 해야함
    public void method2() {         //일반매소드
        System.out.println("SuperClass의 method2");
    }
}

public class ChildClass extends SuperClass{

    @Override
    public void method1() {//method1(추상메소드)
        System.out.println("ChildClass의 method1 - 추상메소드라서 오버라이드 함");
    }
    @Override
    public void method2() {// method2(일반메소드)
        System.out.println("ChildClass의 method2 - 그냥 오버라이드 함");
    }

```

Ex) 다음 예제는 추상클래스를 상속받았음에도 추상메서드를 재정의 하여 사용하지 않을 때 역시 클래스에 abstract를 명시해 다음 상속받을 하위 클래스에게 오버라이딩을 강제하여 넘어갈 수 있다.

```java
public abstract class SuperClass {
    public abstract void method1();//상속받은 클래스에서 오버라이드 해야함
    public void method2() {         //일반매소드
        System.out.println("SuperClass의 method2");
    }
}

//method1(추상매소드), method2(일반매소드)
public abstract class ChildClass2 extends SuperClass {

    @Override
    public void method2() {
        System.out.println("ChildClass2의 method2 - 그냥 오버라이드함");
    }
```

2)@Override

어노테이션(주석)은 로직에 영향을 미치진 않지만 자바 프로그램에 추가적인 정보를 제공하기 위함이다. 예를들어 @override 라는 주석을 달면 해당 매소드가 부모클래스에 있는 매서드를 Override했다는 것을 명시적으로 선언하는것이다. 명시적 선언을 통해 어떠한 매소드를 가르키는지 알수있고 자바에서 문법이나 철자 매개변수등을 체크해 줄수있다. 매서드명 + ctrl + space 단축키로 자동완성이 가능하다.

Ex)

```
@Override
    public void print() {
        super.print();
        System.out.println("   [반]" + ban);
    }
```