

6 월 1 일차 복습

강병화

1.종속

```
class A {
    int a = 10;

    void b () {
        System.out.println("A");
    }
}

// extends 키워드가 바로 상속!
// 상속: 말 그대로 재산을 물려 받는것이다.
//      클래스의 내용물들을 활용할 수 있게 된다.
class AA extends A {
    int a = 20;

    void b () {
        System.out.println("AA");
    }
    void c () {
        System.out.println("C");
    }
}

public class ExtendsTest {
    public static void main(String[] args) {
        A a = new A();
        a.b();
        System.out.println("A a: " + a.a);

        AA aa = new AA();
        aa.b();
        aa.c();
        System.out.println("AA aa: " + aa.a);

        // new의 대상은 AA()이며
        // 접근 데이터는 데이터타입 A를 참조해야한다.
        A a1 = new AA();
        a1.b();
        System.out.println("A a1: " + a1.a);
    }
}
```

1. extends

- 부모에서 선언 / 정의를 모두하며 자식은 메소드 / 변수를 그대로 사용할 수 있음

부모의 메소드를 그대로 사용할 수 있으며 오버라이딩 할 필요 없이 부모에 구현되어있는 것을 직접 사용 가능하다.

. extends는 클래스 한 개만 상속 받을 수 있다.

. extends 자신 클래스는 부모 클래스의 기능을 사용한다.

2.Interface

```
// 인터페이스 작성법
// 1. 일단 interface를 적는다.
// 2. 인터페이스명(일종의 클래스 같은 것이라고 보면 됨)을 적는다.
// 3. 인터페이스 내부에는 매서드 프로토타입을 작성한다.
//    (프로토타입이 뭘까요 ? 매서드의 접근 제한자, 리턴 타입, 매서드 이름, 입력등을 기록한 형태)
interface Remocon {

    // 추상화란 무엇인가 ???????
    // 객체 <<<=== 대표적인 추상화의 예
    // 객체 <<<=== 현 시점에서 우리는 무엇을 생각하는가 ?
    //          new, 메모리에 올라간 데이터들 혹은 정보들 ...
    // 단어가 어떤 함축된 의미를 포함해버렸음(우리는 알게 모르게 사용하고 있었고)
    // 객체란 단어만 보고도 이것이 어떻게 어떻게 형성되었는지 등이 이미 뇌리에 스치고 있음

    // KKK사의 컴퓨터를 켜다.
    // GH사의 라디오를 켜다.
    // A사의 리모콘을 켜다.    =====> 켜다(원진 모르겠지만)
    // B사의 리모콘을 켜다.
    // .....
    // Z사의 리모콘을 켜다.

    // OOP(객체지향)에서 제일 중요시 여기는 것이 바로 추상화다.
    // 현재까지의 내용을 토대로 추상화란 궁극적으로 무엇을 추구하는것인가 ?

    // 복잡하고 어렵고 토나오는것은 우리가 해줄게
    // (자바 라이브러리 개발자 진영 및 스프링 프레임워크 개발 진영)
    // 라이브러리 사용자들은 편하게 API 사용해서 개발만 하세요 ~
    // 이런 입력 ----> Black Box(블랙 박스) ----> 요런 출력이 나와요

    // sout() ==> System.out.println()
```

인터페이스란? 일종의 가이드 라인 동일한 목적 하에 동일한 기능을 보장하기 위해!

>>자바를 통해 개발코드 수정을 줄이고 유지보수성을 높이기 위해

디폴트 메소드를 정의 하는 것 기본 구현부를 제공

여기서 각자 환경에따라 오버라이딩을 통해 재구현 할 수 있다.

인터페이스는 추상 메소드를 통해 강력한 강제성을 가진다