

# 다형성(polymorphism)





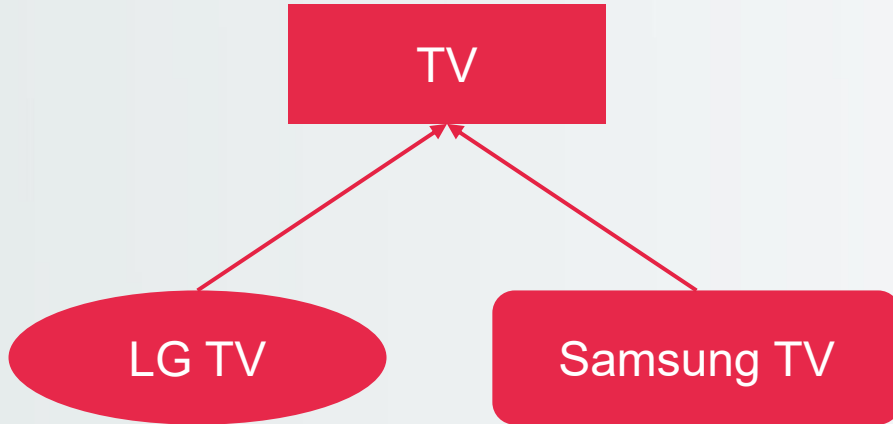
다형성



# 다형성

## 다형성

- 여러 개의 형태를 갖는다는 의미로, 객체지향 프로그래밍 3대 특징 중 하나
- 상속을 이용한 기술로, 자식객체를 부모클래스 타입의 변수로 다룰 수 있는 기술



# 다형성

## 다형성

```
public class SamsungTV extends TV{  
}  
Public class LGTV extends TV{  
}
```

다형성을 이용한 객체 생성 방법

```
TV sTv = new SamsungTV();  
TV lTv = new LGTV();
```

※ 부모타입의 변수에 자식타입의 객체를 저장  
반대로 자식타입 변수는 부모타입객체를 저장할 수 없음



# 다형성

## 클래스 형변환 – up casting

부모타입의 참조형 변수로 자식타입의 객체 주소를 받은 경우 해당 객체는 원래 부모타입의 멤버(변수, 메소드)만 참조가 가능

```
TV sTv = new SamsungTV();
```

이경우 SamsungTV에 추가적인 메소드와 변수가 있더라도 sTv참조형 변수는 TV타입의 변수이기때문에 SamsungTV에 추가로 작성된 메소드와 변수는 사용할 수 없고 TV객체에 있는 변수와 메소드만 사용이 가능함



# 다형성

## 클래스 형변환 – down casting

자식객체의 주소를 받은 부모 참조형 변수를 가지고 자식객체에만 존재하는 멤버를 참조 해야 하는 경우, 자식클래스 타입으로 참조형 변수를 형 변환 해야함  
이러한 형 변환을 down casting이라고 하며, 자동으로 처리되지 않기 때문에 명시적으로 형 변환 해야함

```
TV sTv = new SamsungTV();  
((SamsungTV)sTV).samsungFunc();
```

※ 클래스간의 형 변환은 반드시 상속관계에 있는 클래스끼리만 가능

