

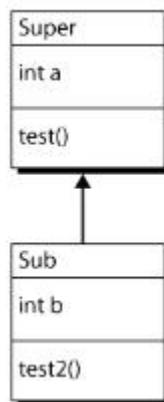
제 18 강 : 다형성

※ 학습목표

- ✓ 다형성의 개념을 설명할 수 있다.
- ✓ 다형성을 활용한 프로그램을 작성할 수 있다.

1. 다형성

- ✓ 한 타입의 참조변수로 여러 타입의 객체를 참조할 수 있도록 하는 것
- ✓ 부모클래스 타입의 참조변수로 자식클래스의 인스턴스를 참조할 수 있도록 함
- ✓ 부모클래스 = 자식



기존의 선언 및 생성

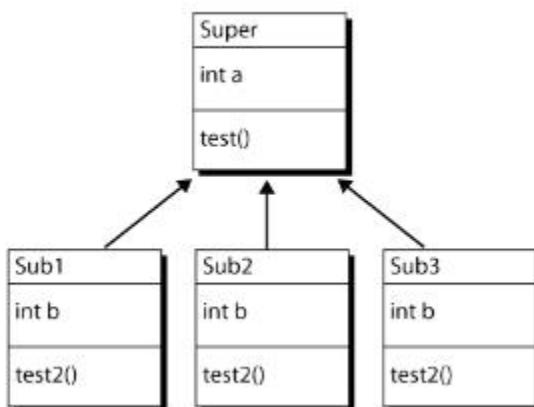
```
Super super = new Super();
Sub sub = new Sub();
```

다형성 선언 및 생성

```
Super s = new Sub();
```

① 상속의 실제 활용

- ✓ 다형성에서 메소드는 자식, 멤버는 부모것을 사용한다.



다형성 선언 및 생성

```
Super s1 = new Sub1();
Super s2 = new Sub2();
Super s3 = new Sub3();
```

[실습]

```
1 package tommy.java.exam01;
2
3 class ParentEx {
4     int foo = 5;
5
6     public int getNumber(int a) {
7         return a + 1;
8     }
9 }
10
11 public class SonEx extends ParentEx {
12     int foo = 7;
13
14     public int getNumber(int a) {
15         return a + 2;
16     }
17
18     public static void main(String args[]) {
19         ParentEx pe = new SonEx();
20         System.out.println(pe.getNumber(0)); // 결과 2
21         System.out.println(pe.foo);         // 결과 5
22     }
23 }
```

2. 다형성을 활용한 종합예제 : 동물병원

✓ 아래의 클래스들을 같은 패키지에 작성하고 실행한다.

✓ Animal.java 작성

```
1 package tommy.java.exam02;
2
3 public class Animal {
4     public String scream() {
5         return "동물 울음소리";
6     }
7
8     public String getName() {
9         return null;
10    }
11 }
```

✓ Animal 클래스를 상속받는 Dog 클래스 작성

```
1 package tommy.java.exam02;
2
3 class Dog extends Animal {
4     private String name;
5
6     public Dog() {
7         name = getClass().getSimpleName();
8     }
9
10    public String scream() {
11        return "멍멍개깽";
12    }
13
14    public String getName() {
15        return name;
16    }
17 }
```

✓ Animal 클래스를 상속받는 Cat 클래스 작성

```
1 package tommy.java.exam02;
2
3 class Cat extends Animal {
4     private String name;
5
6     public Cat() {
7         name = getClass().getSimpleName();
8     }
9
10    public String scream() {
11        return "야옹야옹";
12    }
13
14    public String getName() {
15        return name;
16    }
17 }
```

✓ Animal 클래스를 상속받는 Tiger 클래스 작성

```
1 package tommy.java.exam02;
2
3 class Tiger extends Animal {
4     private String name;
```

```

5
6     public Tiger() {
7         name = getClass().getSimpleName();
8     }
9
10    public String scream() {
11        return "그르르어흥어흥";
12    }
13
14    public String getName() {
15        return name;
16    }
17 }

```

✓ 동물을 치료하는 Hospital 클래스 작성

```

1 package tommy.java.exam02;
2
3 class Hospital {
4     public void inject(Animal animal) {
5         System.out.println(animal.getName() +
6                             "을(를) 치료하기 위해 주사를 놓았습니다.");
7         System.out.println(animal.scream());
8     }
9 }

```

✓ 동물병원 메인클래스 작성

```

1 package tommy.java.exam02;
2
3 public class HospitalMain {
4     public static void main(String[] args) {
5         Hospital hospital = new Hospital();
6         Animal dog = new Dog();
7         Animal cat = new Cat();
8         Animal tiger = new Tiger();
9         hospital.inject(dog);
10        hospital.inject(cat);
11        hospital.inject(tiger);
12    }
13 }

```