

Local Class And Anonymous Class

2016년 9월 29일 목요일 오전 1:00

Local Class

Local Class는 Method 내에 정의되어 있어서, Method 내에서만 Instance의 생성 및 참조변수의 선언이 가능하다.

```
// OuterClass
class OuterClass {

    public LocalClass createLocalClassInst() {
        // Local Class : Method 내에 선언된 Class
        class LocalClass {

        }

        return new LocalClass();
    }
}
```

그런데, 반환된 LocalClass의 Instance는 외부에서 참조변수로 받을 수가 없다...

(Object Class로 받을 수는 있으나 래핑되어서 Object Class의 Method만 사용할 수 있다.)

Local Class의 적절한 사용모델

```
interface Readable {
    public void read();
}

class OuterClass {
    private String myName;

    OuterClass(String name){
        myName = name;
    }

    public Readable createLocalClassInst(){
        class LocalClass implements Readable {
            public void read(){
                System.out.println("Outer inst name : "+myName);
            }
        }

        return new LocalClass();
    }
}

public class UseLocalClass {
    public static void main(String[] args) {
        OuterClass out1 = new OuterClass("First");

        Readable localInst1 = out1.createLocalClassInst();
        /*
            LocalClass는 Interface를 Implements 받았기 때문에 외부에서 Interface의 참조변수를 통해
            LocalClass의 Instance를 참조할 수 있다.
        */
    }
}
```

```

        localInst1.read();

        OuterClass out2 = new OuterClass("Second");
        Readable localInst2 = out2.createLocalClassInst();
        localInst2.read();
    }
}

```

Local Class의 지역변수, 매개변수 접근

```

public Readable createLocalClassInst(final int instID){
    class LocalClass implements Readable {
        public void read(){
            System.out.println("Outer inst name : "+myName);
            System.out.println("Outer inst ID : "+instID);
        }
    }

    return new LocalClass();
}

```

- i. createLocalClassInst Method가 호출되면 Local Class의 Instance가 생성되어 반환된다.
- ii. 반환이 끝나면 createLocalClassInst는 종료되는데, 그 순간 createLocalClassInst의 지역변수와 매개변수는 소멸
- iii. 그 후 LocalClass의 read Method를 호출하면 instID를 참조 해야는데 **이미 사라지고 없다.**

- iv. **그래서 매개변수에 final이라는 특성을 부여해야 LocalClass가 접근이 가능하다.**

final 선언에 의해 참조변수는 어떠한 상황에서도 값의 변경이 불가능하다.

final 변수가 사라지지 않도록 LocalClass가 참조할 수 있는 메모리영역에 **복사 해놓는다!**
그리고 원본과 복사본의 내용이 바뀌면 안되므로 final 선언!

Anonymous Class (이름없는 클래스)

```

public Readable createLocalClassInst(final int instID){

```

// interface는 인스턴스 생성이 불가능하다.

```

    /*
        자바는 Interface에 정의된 메소드를 바로 덧붙여서 정의를 해주면 인스턴스 생성을 허용해준다!
        그리고 이렇게 생성된 인스턴스는 이름이 없기 때문에 Anonymous라고 한다.
    */
    return new Redable(){ public void read(){ System.out.println(instID); }
}

```

```

interface Readable {
    public void read();
}

```

```

public class AnonymousClassDemo {
    public static void main(String[] args){

        Readable readableInterface = new Readable(){
            @Override
            public void read(){

```

```
        System.out.println("인스턴스 생성을 위한 abstract 메소드의 구현");
    }
}
}
```
