

패턴을 통한 객체지향 프로그램의 이해

이소영 (yisy0703@naver.com)



패턴이란

우리의 개발자 선배님들이 객체지향 언어의
장점들을 모아 가장 효율적으로 개발을 할 수
있게 만들어 놓은 틀

Singleton P Pattern

싱글턴 패턴(Singleton pattern)이란 어떤 클래스의 객체는 오직 하나인 유일한 객체를 만들어 여러가지 상황에서 동일한 객체에 접근하기 위해 만들어진 패턴이다. 실제로 많이 쓰이는 유형이니 꼭 이해하자

1. Singleton Pattern

SingletonClass

```
private static SingletonClass INSTANCE;  
private int i=10;  
private SingletonClass() {}
```

```
Static SingletonClass getInstance(){객체가없을때만생성해서 INSTANCE로}  
Int getI(){return i;}  
void setI(int){this.i=i;}
```

FirstClass

```
FirstClass(){SingletonClass.getSingletonClass();  
setI(999);}
```

SecondClass

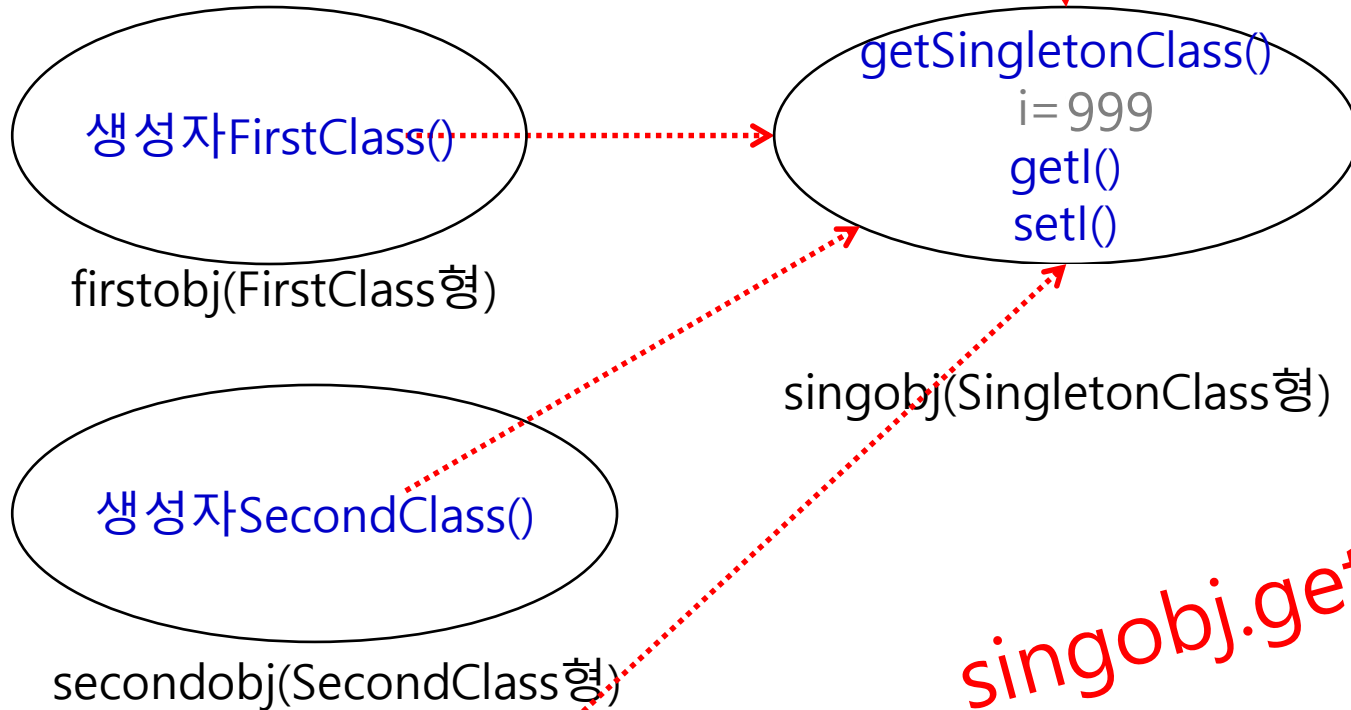
```
SecondClass(){SingletonClass.getSingletonClass();}
```

MainSing

```
Staic void  
Main()
```

1-1 main()

SingletonClass.INSTANCE



singobj.getI() ?

main()함수내에서도 SingletonClass형 객체 생성