

单例模式

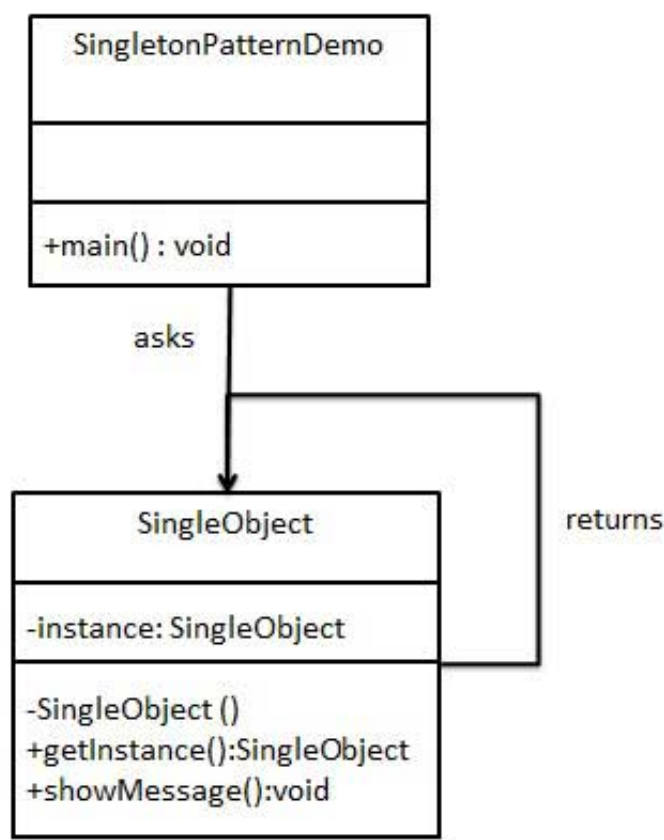
单例模式是Java中最简单的设计模式之一。这个模式属于创建型模式，提供了创建一个对象的最佳实践。

这个模式构造一个单个的类负责去创建一个对象并确保只有一个对象被创建。这个类提供了在不创建该类实例的情况下直接访问这个对象的方法。

实现

我们准备创建一个 `SingleObject` 类。SingleObject拥有一个私有构造器和一个自身的静态实例。

`SingleObject` 提供给外界一个静态方法来获得自身的静态实例。



第一步

创建一个单例类。
`SingleObject.java`

```
public class SingleObject {

    //create an object of SingleObject
    private static SingleObject instance = new SingleObject();

    //make the constructor private so that this class cannot be
    //instantiated
    private SingleObject(){}

    //Get the only object available
    public static SingleObject getInstance(){
        return instance;
    }

    public void showMessage(){
        System.out.println("Hello World!");
    }
}
```

第二步

SingletonPatternDemo.java

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

        //illegal construct
        //Compile Time Error: The constructor SingleObject() is not
        visible
        //SingleObject object = new SingleObject();

        //Get the only object available
        SingleObject object = SingleObject.getInstance();

        //show the message
        object.showMessage();
    }
}
```

第三步

验证输出。

```
Hello World!
```