

```
package com.sxt.server.basic;
```

```
import java.lang.reflect.InvocationTargetException;
```

```
/**
```

```
 * 发射：把java类中的各种方法（方法，属性，构造器，类名）映射成一个个的java对象
```

```
 * 1. 获取class对象
```

```
 * 三种方式：Class.forName（"完整路径"）
```

```
 * 2. 动态创建对象
```

```
 * clz.getConstructor().newInstance()
```

```
 *
```

```
 * @author 江
```

```
 *
```

```
 */
```

```
public class ReflectTest01 {
```

```
    public static void main(String[] args) throws ClassNotFoundException,  
    InstantiationException, IllegalAccessException, IllegalArgumentException,  
    InvocationTargetException, NoSuchMethodException, SecurityException {
```

```
        //三种方式
```

```
        //1. 对象.getClass()
```

```
        Iphone iphone=new Iphone();
```

```
        Class clz=iphone.getClass();
```

```
        //2. 类.class()
```

```
        clz=Iphone.class;
```

```
        //2. Class.forName("包名.类名")
```

```
        clz=Class.forName("com.sxt.server.basic.Iphone");
```

```
        //创建对象
```

```
        Iphone iphone2=(Iphone)clz.newInstance();    //不推荐
```

```
        System.err.println(iphone2);
```

```
        Iphone iphone3=(Iphone)clz.getConstructor().newInstance();
```

```
System.err.println(iphone2);
```

```
}
```

```
}
```

```
class Iphone{
```

```
    public Iphone() {
```

```
    }
```

```
}
```