链滴

java 的反射

作者：liaoshengzhe

原文链接：https://ld246.com/article/1505468135332

来源网站：链滴

1. 反射机制：在程序运行期动态获取其类型信息的机制。

2. java.lang.Class：代表正在运行的类和接口的类型信息对象。

3. 获取Class对象的方式：

1. 通过Object类提供的Class getClass()方法。

Integer integer = Integer.valueOF(12);

Class clazz = integer.getClass();

2. 通过Class类提供的静态方法forName(String 类名)来获取。

Class clazz = Class.forName("java.lang.Integer");

3. 通过类的静态属性class也可以获取： Class clazz = Integer.class;

4. 通过反射获取某个类型的属性、方法、构造方法，父类和实现的接口。

1. 获取类或接口的属性信息：用java.lang.reflect.Field代表。

public Field[] getFields(); 返回一个包含公有字段对应的Field对象数组

public Field getField(String name) throws NoSuchFieldException, SecurityException; 获取指定称的公有字段对应的Field对象

public Field[] getDeclaredFields() throws SecurityException 返回一个包含的所有字段对应的Fiel对象数组

public Field getDeclaredField(String name) throws NoSuchFieldException, SecurityException;

2. 获取类或接口的方法：用java.lang.reflect.Method代表。

public Method[] getMethods() throws SecurityException; 返回一个包含公有 Method对象的数组

public Method getMethod(String name, Class... parameterTypes) throws NoSuchMethodExcetion, SecurityException

返回指定方法名和指定参数信息的Method对象。

3. 获取类的构造方法：用java.lang.reflect.Constructor代表。

public Constructor[] getConstructors() throws SecurityException

public Constructor getConstructor(Class... parameterTypes) throws NoSuchMethodException, ecurityException

4. 获取方法的参数类型和返回值类型：调用Method类提供的对应方法。

public Class[] getParameterTypes(); 按照声明顺序返回 Class 对象的数组

public Class getReturnType();

5. 通过反射来创建该类型的对象：通过Class提供的newInstance()方法调用默认的构造方法来创建类型的一个对象。

Object obj = Class.forName("java.util.Date").newInstance();

6. 通过反射访问某个对象的属性或方法：

Field field = clazz.getDeclaredField("属性名"); //访问属性

field.setAccessible(true); //取消Java语言访问检查

Object obj = field.get(该类型的某个对象); //获取指定对象上该属性的值

field.set(该类型的某个对象, Object 值)；//将指定对象变量上此 Field对象表示的字段设置为指定的值。

```
Method method = clazz.getDeclaredMethod("方法名", 参数类型信息); //访问指定方法
method.setAssessible(true);  //取消Java语言访问检查
Object obj = method.invoke(Object 该类型的某个对象, Object... 实参列表);  //对带有指定参数的定对象调用由此 Method对象表示的底层方法
```

7. 通过反射调用带参的构造方法比较麻烦。很多应用到反射技术的高级框架都要求你的类提供一个带参数的构造方法。

大多数的高级框架的核心技术都是：XML做配置，并解析，然后用反射来创建对象。