

## 反射

武永亮

### 准备知识

- 什么是动态编程语言?
- 答:程序运行时,允许改变程序结构或变量类型—这种语言称为动态语言
  - 从这个观点看, Perl, Python, Ruby是动态语言, C++, Java, C#不是动态语言
  - Java有着一个非常突出的动态相关机制: Reflection

## 讲授思路

- 反射机制的概念以及原理
- 反射机制的具体实现
- 反射机制的应用

#### Java反射的概念

• 在1982年由Smith正式提出了反射的概念

9

在运行状态中,对于任意一个类,都能够知道其所有属性和方法;对于任意一个对象,都能够调用其任意一个方法;

——这种动态获取信息以及动态调用对象的方法的功能 称为java语言的反射机制



#### Java反射功能

• Java反射提供以下功能

• 在运行时判断任意一个对象所属的类

• 在运行时构造任意一个类的对象

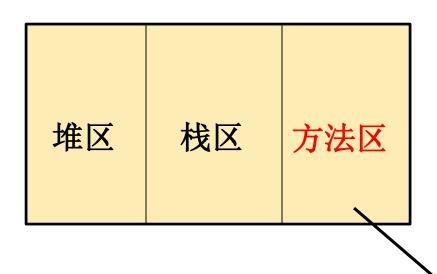
• 在运行时判断任意一个类所具有的成员变量和方法

• 在运行时调用任意一个对象的方法

• 生成动态代理

#### Java反射机制的原理

• Java运行时的数据区



Class对象

武永亮

Java虚拟机装载某类型时,类装载器会定位相应的class文件,然后将 其读入到虚拟机中,并提取class中 的类型信息,信息存储到方法区中。

### Java反射机制中设计的类

- Class: 类的实例表示正在运行的 Java 应用程序中的类和 接口
- Field:提供有关类或接口的属性的信息,以及对它的动态 访问权限
- Constructor:提供关于类的单个构造方法的信息以及对它的访问权限
- Method:提供关于类或接口上单独某个方法的信息

#### Class类

- Class 类十分特殊,其实例用以表达Java程序运行时的类和接口
- · 获取Class对象
  - 通过Object的getClass()方法
  - 通过Class的getSuperClass()方法
  - 通过Class的静态forName()方法
  - 对于包装器类型,通过类名.TYPE属性

## **Class API**

-2- M. 146 and	
万法摘要	
⟨U⟩ <u>Class</u> ⟨? extends U⟩	asSubclass(Class(U) clazz) 强制转换该 Class 对象,以表示指定的 class 对象所表示的类的一个子类。
I	cast (Object obj) 将一个对象强制转换成此 Class 对象所表示的类或接口。
boolean	desiredAssertionStatus() 如果要在调用此方法时将要初始化该类,则返回将分配给该类的断言状态。
static <u>Class</u>	forWame(String className) 返回与带有给定字符串名的类或接口相关联的 Class 对象。
static <u>Class</u>	forWame (String name, boolean initialize, ClassLoader loader) 使用给定的类加载器,返回与带有给定字符串名的类或接口相关联的 Class 对象。
<pre><a annotation="" extends=""> A</a></pre>	getAnnotation(Class(A) annotationClass) 如果存在该元素的指定类型的注释,则返回这些注释,否则返回 null。
Annotation[]	getAnnotations() 返回此元素上存在的所有注释。
String	getCanonicalWame() 返回 Java Language Specification 中所定义的底层类的规范化名称。
<u>Class</u> []	getClasses() 返回一个包含某些 Class 对象的数组,这些对象表示属于此 Class 对象所表示的类的成员的所有公共类和接口。
ClassLoader	getClassLoader() 返回该类的类加载器。
Class(?)	getComponent Type () 返回表示数组组件类型的 Class。
Constructor(I)	getConstructor(Class parameterTypes) 返回一个 Constructor 对象,它反映此 Class 对象所表示的类的指定公共构造方法。
Constructor(?>[]	getConstructors() 返回一个包含某些 Constructor 对象的数组,这些对象反映此 Class 对象所表示的类的所有公共构造方法。
Annotation[]	getDeclaredAnnotations() 返回直接存在于此元素上的所有注释。
<u>Class</u> []	getDeclaredClasses()



#### Java反射机制的应用

- 操作数据库
  - 动态创建SQL语句
- 解析XML
  - 解析XML动态生成对象
- 动态代理
- 其它的框架中使用
  - Struts框架、Spring框架、Hibernate框架

#### Java反射机制的缺点

- 主要的缺点是对性能有影响
  - 使用反射基本上是一种解释操作,您可以告诉JVM您希望做什么并且它满足您的要求。这类操作总是慢于直接执行相同的操作

## 总结

- · Java反射机制的基本原理
  - 反射的概念
  - 反射的相关类及应用
- Java反射机制的基本应用

武永亮

## 课后阅读

• 动态代理

武永亮

# Thank You