

java新手入门：Java反射机制(上) PDF转换可能丢失图片或格式，建议阅读原文

https://www.100test.com/kao_ti2020/145/2021_2022_java_E6_96_B0_E6_89_8B_c104_145312.htm 摘要

Reflection 是Java被视为动态（或准动态）语言的一个关键性质。这个机制允许程序在运行时透过Reflection APIs取得任何一个已知名称的class的内部信息，包括其modifiers（诸如public, static 等等）、superclass（例如Object）、实现之interfaces（例如Cloneable），也包括fields和methods的所有信息，并可于运行时改变fields内容或唤起methods。本文借由实例，大面积示范Reflection APIs。关于本文：读者基础：具备Java语言基础。本文适用工具

：JDK1.5 关键词：Introspection（内省、内观）Reflection（反射）有时候我们说某个语言具有很强的动态性，有时候我们会区分动态和静态的不同技术与作法。我们朗朗上口动态绑定（dynamic binding）、动态链接（dynamic linking）、动态加载（dynamic loading）等。然而“动态”一词其实没有绝对而普遍适用的严格定义，有时候甚至像对象导向当初被导入编程领域一样，一人一把号，各吹各的调。一般而言，开发者社群说到动态语言，大致认同的一个定义是：“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。尽管在这样的定义与分类下Java不是动态语言，它却有着一个非常突出的动态相关机制

：Reflection。这个字的意思是“反射、映象、倒影”，用在Java身上指的是我们可以于运行时加载、探知、使用编译期间完全未知的classes。换句话说，Java程序可以加载一个运行

时才得知名称的class，获悉其完整构造（但不包括methods定义），并生成其对象实体、或对其fields设值、或唤起其methods。这种“看透class”的能力（the ability of the program to examine itself）被称为introspection（内省、内观、反省）。Reflection和introspection是常被并提的两个术语。Java如何能够做出上述的动态特性呢？这是一个深远话题，本文对此只简单介绍一些概念。整个篇幅最主要还是介绍Reflection APIs，也就是让读者知道如何探索class的结构、如何对某个“运行时才获知名称的class”生成一份实体、为其fields设值、调用其methods。本文将谈到java.lang.Class，以及java.lang.reflect中的Method、Field、Constructor等等classes。

100Test 下载频道开通，各类考试题目直接下载。详细请访问 www.100test.com