

Java学习编程经验分享（一）Java认证考试 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/564/2021\\_2022\\_Java\\_E5\\_AD\\_A6\\_E4\\_B9\\_A0\\_c104\\_564743.htm](https://www.100test.com/kao_ti2020/564/2021_2022_Java_E5_AD_A6_E4_B9_A0_c104_564743.htm) 来学习Java也有两个年头了，永远不敢说多么精通，但也想谈谈自己的感受，写给软件学院的同仁们，帮助大家技术的道路上少一点弯路。说得伟大一点是希望大家为软件学院争气，其实最主要的还是大家自身的进步提升。

1. 关于动态加载机制 学习Java比C 更容易理解OOP的思想，毕竟C 还混合了不少面向过程的成分。很多人都能背出来Java语言的特点，所谓的动态加载机制等等。当然概念往往是先记住而后消化的，可有多少人真正去体会过动态加载的机制，试图去寻找过其中的细节呢？提供大家一个方法：在命令行窗口运行Java程序的时候，加上这个很有用的参数：`java verbose *.class` 这样会清晰的打印出被加载的类文件，大部分是jdk自身运行需要的，最后几行会明显的看到自己用到的那几个类文件被加载进来的顺序。即使你声明了一个类对象，不实例化也不会加载，说明只有真正用到那个类的实例即对象的时候，才会执行加载。这样是不是大家稍微能明白一点动态加载了呢？^\_^

2. 关于寻找class文件原理 建议大家在入门的时候在命令行窗口编译和运行，不要借助JCreator或者Eclipse等IDE去帮助做那些事情。尝试自己这样做：`javac -classpath yourpath *.java java -classpath yourpath *.class` 也许很多人都能看懂，设置classpath的目的就是告诉编译器去哪里寻找你的class文件。不过至少笔者今日才弄懂JVM去查询类的原理，编译器加载类要依靠classloader，而classloader有3个级别，从高到低分别是BootClassLoader(名

字可能不准确), ExtClassLoader, AppClassLoader. 这3个加载器分别对应着编译器去寻找类文件的优先级别和不同的路径 : BootClassLoader对应jre/classes路径, 是编译器最优先寻找class的地方。 ExtClassLoader对应jre/lib/ext路径, 是编译器次优先寻找class的地方, AppClassLoader对应当前路径, 所以也是编译器默认找class的地方 其实大家可以自己写个程序简单的测试, 对任何class, 例如A, 调用new A().getClass().getClassLoader().toString() 打印出来就可以看到, 把class文件放在不同的路径下再次执行, 就会看到区别。 特别注意的是如果打印出来是null就表示到了最高级 BootClassLoader, 因为它是C 编写的, 不存在Java对应的类加载器的名字。 寻找的顺序是一种向上迂回的思想, 即如果本级别找不到, 就只能去本级别之上的找, 不会向下寻找。 不过似乎从Jdk1.4到Jdk1.6这一特点又有改变, 没有找到详细资料。 所以就不举例子了。 告诉大家设计这种体系的是Sun公司曾经的技术核心宫力先生, 一个纯种华人哦! 这样希望大家不至于迷惑为什么总报错找不到类文件, 不管是自己写的还是导入的第三方的jar文件 (J2ee中经常需要导入的)。 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](http://www.100test.com)