

Java使用native方法来扩展Java程序的功能 PDF转换可能丢失图片或格式，建议阅读原文

[https://www.100test.com/kao\\_ti2020/284/2021\\_2022\\_Java\\_E4\\_BD\\_BF\\_E7\\_94\\_A8\\_c104\\_284359.htm](https://www.100test.com/kao_ti2020/284/2021_2022_Java_E4_BD_BF_E7_94_A8_c104_284359.htm)

Java不是完美的，Java的不足除了体现在运行速度上要比传统的C慢许多之外，Java无法直接访问到操作系统底层（如系统硬件等），为此Java使用native方法来扩展Java程序的功能。

可以将native方法比作Java程序同C程序的接口，其实现步骤：1、在Java中声明native()方法，然后编译；2、用javah产生一个.h文件；3、写一个.cpp文件实现native导出方法，其中需要包含第二步产生的.h文件（注意其中又包含了JDK带的jni.h文件）；4、将第三步的.cpp文件编译成动态链接库文件；5、在Java中

用System.loadLibrary()方法加载第四步产生的动态链接库文件，这个native()方法就可以在Java中被访问了。

JAVA本地方法适用的情况  
1.为了使用底层的主机平台的某个特性，而这个特性不能通过JAVA API访问  
2.为了访问一个老的系统或者使用一个已有的库，而这个系统或这个库不是用JAVA编写的  
3.为了加快程序的性能，而将一段时间敏感的代码作为本地方法实现。

```
首先写好JAVA文件
/* * Created on 2005-12-19 Author shaoqi */
package com.hode.hodeframework.model0update;
public class CheckFile {
    public native void displayHelloWorld();
    static {
        System.loadLibrary("test");
    }
    public static void main(String[] args) {
        new CheckFile().displayHelloWorld();
    }
}
然后根据写好的文件编译成CLASS文件
然后在classes或bin之类的class根目录下执行
javah -jni com.hode.hodeframework.model0update.CheckFile ,
就会在根目录下得到一
```

个com\_hode\_hodeframework\_model0update\_CheckFile.h的文件  
然后根据头文件的内容编

写com\_hode\_hodeframework\_model0update\_CheckFile.c文件  
#include "CheckFile.h" #include #include JNIEXPORT void  
JNICALL

Java\_com\_hode\_hodeframework\_model0update\_CheckFile\_displa  
yHelloWorld(JNIEnv \*env, jobject obj) { printf("Hello world!\n").

return. } 之后编译生成DLL文件如 “ test.dll ” ， 名称

与System.loadLibrary("test")中的名称一致 vc的编译方法：cl

-I%java\_home%\include -I%java\_home%\include\win32 -LD

com\_hode\_hodeframework\_model0update\_CheckFile.c -Fetest.dll

最后在运行时加参数-Djava.library.path=[dll存放的路径]

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)