**JNI开发入门**

**一、概述**  
JNI(java native interface)，JAVA本地接口调用，目的是为了JAVA可以调用本地程序，主要优点有：

* 提高效率，解决性能问题
* 防止反编译，提高核心代码安全性
* 增强本机接口调用
* 嵌入式开发应用

    如下通过一个最简单的小demo，演示如何进行jni开发。  
**二、demo演示**  
**1. 新建JAVA文件**  
新建一个java类，如下：

1. view plaincopy to clipboardprint?package chb.test.jni;  publicclass HelloJni {          public
2. native
3. void sayHello();
4. static {            System.loadLibrary("hellojni");        }
5. public
6. static
7. void main(String[] args) {            new HelloJni().sayHello();        }    }

复制代码

HelloJni.java在硬盘上的位置：D:\workspace\MyWork\src\chb\test\jni\HelloJni.java  
  
**2. 生成class文件**  
1) 利用javac命令进行编译，进入D:\workspace\MyWork\src\chb\test\jni目录，利用

1. view plaincopy to clipboardprint?
2. javac HelloJni.java

复制代码

2) 利用IDE，例如我利用eclipse开发，自动编译，编译后的文件位于：D:\workspace\MyWork\bin\chb\test\jni目录下  
  
**3. 生成C头文件**  
1) 如果利用javac命令编译，则进入D:\workspace\MyWork\src目录，然后输入  
[view plaincopy to clipboardprint?](http://blog.csdn.net/hbcui1984/archive/2009/12/10/4980805.aspx)

1. javah -jni chb.test.jni.HelloJni

会在D:\workspace\MyWork\src目录下发现新生成的头文件：chb\_test\_jni\_HelloJni.h，文件内容如下：

1. view plaincopy to clipboardprint?
2. /\* DO NOT EDIT THIS FILE - it is machine generated \*/#include <jni.h>/\* Header for class chb\_test\_jni\_HelloJni \*/#ifndef \_Included\_chb\_test\_jni\_HelloJni#define \_Included\_chb\_test\_jni\_HelloJni#ifdef \_\_cplusplusextern
3. "C" {#endif/\* \* Class:     chb\_test\_jni\_HelloJni \* Method:    displayHelloJni \* Signature: ()V \*/JNIEXPORT void JNICALL Java\_chb\_test\_jni\_HelloJni\_displayHelloJni  (JNIEnv \*, jobject);#ifdef \_\_cplusplus}#endif#endif

复制代码

2) 如果利用IDE编译，则直接在编译根目录使用javah命令即可，例如进入：D:\workspace\MyWork\bin，然后使用

1. view plaincopy to clipboardprint?
2. javah -jni chb.test.jni.HelloJni

复制代码

效果相同。  
  
**注意**：因为HelloJni是在chb.test.jni包中的，因此需要写明全路径，否则如果直接到目录：D:\workspace\MyWork\bin\chb\test\jni，然后执行javah HelloJni，会报出异常，如下：

1. view plaincopy to clipboardprint?
2. D:\workspace\ctcc\MyWork\src\chb\test\jni>javah HelloJni错误：无法访问 HelloJni错误的类文件： .\HelloJni.class类文件包含错误的类： chb.test.jni.HelloJni请删除该文件或确保该文件位于正确的类路径子目录中。com.sun.tools.javac.util.Abort        at com.sun.tools.javac.comp.Check.completionError(Check.java:164)        at com.sun.tools.javadoc.DocEnv.loadClass(DocEnv.java:149)        at com.sun.tools.javadoc.RootDocImpl.<init>(RootDocImpl.java:77)        at com.sun.tools.javadoc.JavadocTool.getRootDocImpl(JavadocTool.java:159)        at com.sun.tools.javadoc.Start.parseAndExecute(Start.java:330)        at com.sun.tools.javadoc.Start.begin(Start.java:128)        at com.sun.tools.javadoc.Main.execute(Main.java:66)        at com.sun.tools.javah.Main.main(Main.java:147)javadoc: 错误 - 致命错误2 错误

复制代码

**4. 新建C实现，并生成dll文件**  
新建c工程，本人利用dev c++，源代码如下：

1. view plaincopy to clipboardprint?
2. #include <windows.h>#include <stdio.h>#include <stdlib.h>#include <jni.h>#include "HelloJni.h"JNIEXPORT void JNICALL Java\_chb\_test\_jni\_HelloJni\_sayHello(JNIEnv \* a, jobject b){    printf("hello Jni!\n");    return;}

复制代码

方法很简单，就输出一句话，然后编译工程，生成hellojni.dll，并且将hellojni.dll拷贝到windows\system32目录下  
**5. 执行java代码**  
运行HelloJni的main方法，输出"hello Jni"，运行成功  
  
  
  
**本文比较简单，主要方便大家上手做第一个demo，有个感性的认识。**