

CSDN新首页上线啦，邀请你来立即体验！(http://feed.csdn.net/)

CSDN

博客 (http://feed.csdn.net/?ref=toolbar)

学院 (http://edu.csdn.net/?ref=toolbar)

下载 (http://download.csdn.net/?ref=toolbar)

更多 ▾



登录 (https://passport.csdn.net/account/login?ref=toolbar) 注册 (http://passport.csdn.net/account/mobile/register?ref=toolbar&utm_source=csdnblog1)

NDK开发 从入门到放弃(七：Android Studio 2.2 CMAKE 高效NDK开发)

原创

2016年11月09日 16:36:36

标签：ndk (http://so.csdn.net/so/search/s.do?q=ndk&t=blog)

11351

前言

之前，每次需要边写C++代码的时候，我的内心都是拒绝的。

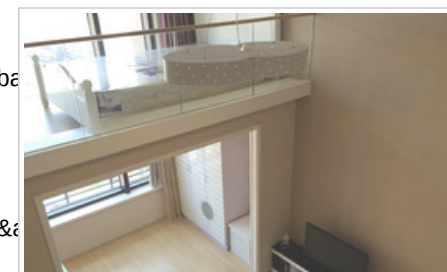
1. 它没有代码提示！！这意味着我们必须自己手动敲出所有的代码，对于一个新手来说，要一个字母都不错且大小写也要正确，甚至要记得住所有的jni函数等，真是太折磨人了...平时写java代码的时候都是写几个字母会出来一大堆提示然后选择的，这样还有一个好处就是很多时候我们不知道有那些函数，但是我们可以通过 obj.，然后就可以看到它有哪些方法函数了。

2. 很多地方会显示为红色，就像是错误提示的那种，当然，如果没错的话还是能编译运行的，但是如果像编写java代码一样，写错的地方才给我红色显示不是更好吗？

这两个问题可折磨死我这个强迫症了...在网上百度了很久如何能够使用Android Studio编写C++代码时有代码提示自动补全功能，但是一直没有找到有效的（不使用CMake的情况下，若有人知道，麻烦告知）。好在一次偶然的机会，在网上看到在 Android Studio 2.2 中愉快地使用 C/C++

(http://blog.csdn.net/wl9739/article/details/52607010)这篇文章，看到上面说Android Studio2.2版本完善了对C/C++的支持，还提及到CMAKE，赶紧把Android Studio从2.0版本升级到了2.2版本，按照介绍新建了一个NDK工程，发现里面居然有我梦寐以求的C++代码自动补全提示！！！喜大普奔！！

赶紧了解学习下Android Studio2.2版本提供的CMAKE方式。打算以后都使用这种方式了，别的不说，就冲可以有C++代码自动补全提示这个功能我就爱上了它了。



loft公寓出租



(http://blog.csdn.net/xiaoyu_93)

码云

2

原创

63

粉丝

57

喜欢

0

(https://github.com/xiaoyu_93)
utm_source=csdnblog1

他的最新文章

更多文章 (http://blog.csdn.net/xiaoyu_93)

安卓，调用系统方法分享到微信
(/xiaoyu_93/article/details/78039611)

Android——App内文件分享功能
(/xiaoyu_93/article/details/76921562)

Android——Loading images with
HTTP POST
(/xiaoyu_93/article/details/73852302)

谷歌地图对接记录（Android）
(/xiaoyu_93/article/details/72865818)



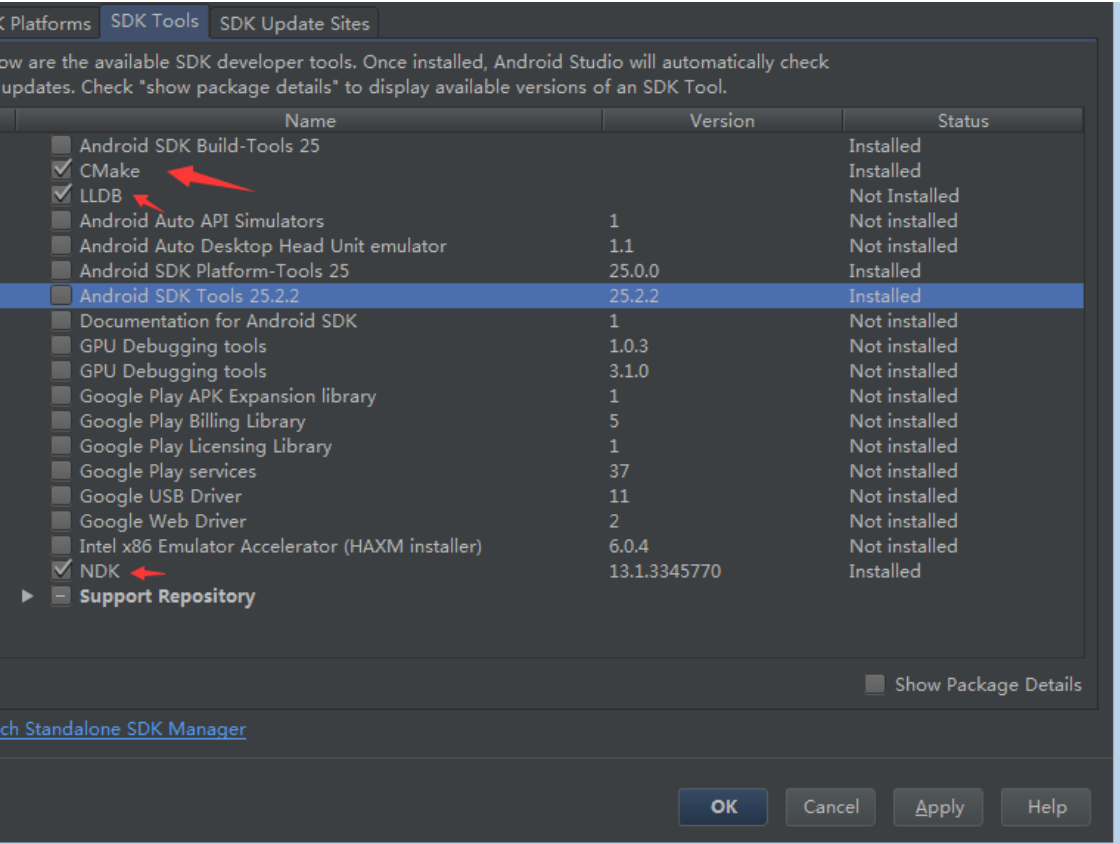
内容举报



返回顶部

工具安装

Android Studio升级到2.2之后，我们可以先配置好NDK开发的一些所需工具，如图，在SDK Tools中勾选安装CMake、LLDB、NDK。



CMake: 外部构建工具。如果你准备只使用 ndk-build 的话，可以不使用它。
LLDB: Android Studio上面调试本地代码的工具。



在线课程



MyM/M在美国点评请施移动端的最佳实践
(http://red.csdn.net/huiyi Course/detail/603?utm_source=blog9)



C语言大型软件设计的面
(http://red.csdn.net/huiyi Course/detail/594?utm_source=blog9)

热门文章

NDK开发 从入门到放弃(一：基本流程入门了解) (/xiaoyu_93/article/details/52870395)
12247

NDK开发 从入门到放弃(七：Android Studio 2.2 CMAKE 高效NDK开发) (/xiaoyu_93/article/details/53082088)
11331

内容举报
返回顶部

Android Studio自带DEMO了解CMAKE

Android Studio升级到2.2版本之后，在创建新的project时，界面上多了一个 Include C++ Support 的选项。勾选它之后将会创建一个默认的C++与JAVA混编的Demo程序。就让我们先来看看这个官方标准Demo吧。

开始之前最好先下载好NDK，见NDK开发 从入门到放弃(一：基本流程入门了解)
(http://blog.csdn.net/xiaoyu_93/article/details/52870395)，即在Project Structure界面Android NDK location处下载或选择正确的路径。或者使用上方提供的工具安装方法来进行下载。否则，创建的新project也会报错，需要配置好后clean。

张利华：夺命的中国IT互联网行业 (/xiaoyu_93/article

7662

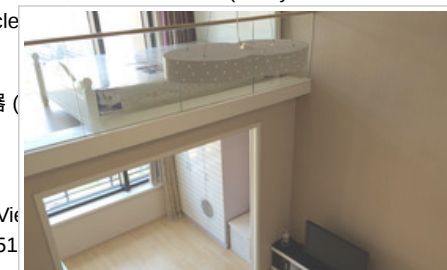
C# 计算器 (7949)

5385

安卓WebView

e/details/51

3533



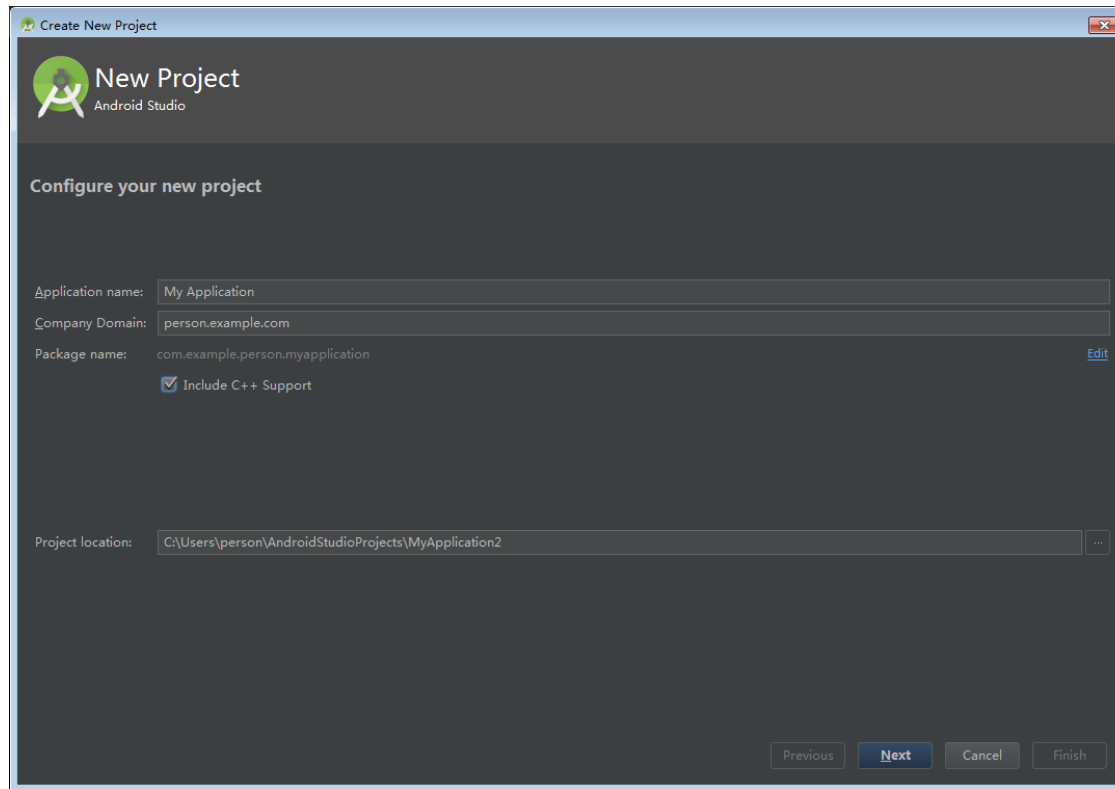
loft公寓出租



内容举报

返回顶部

File -> New -> New Project，在如下界面中勾选 Include C++ Support，然后一路 Next，直到 Finish 为止即可。



项目打开后我们查看目录结构，与常规项目不同的是多了 .externalNativeBuild 文件夹、cpp 文件夹、CMakeLists.txt 文件，如下图：



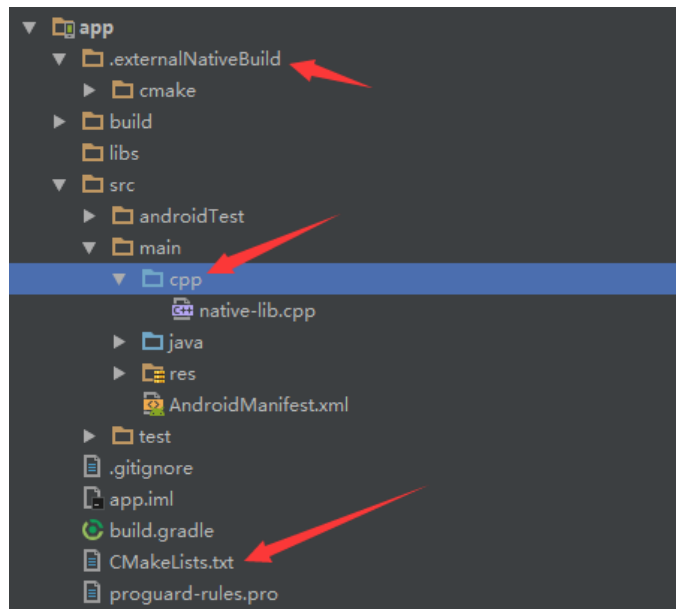
内容举报



返回顶部

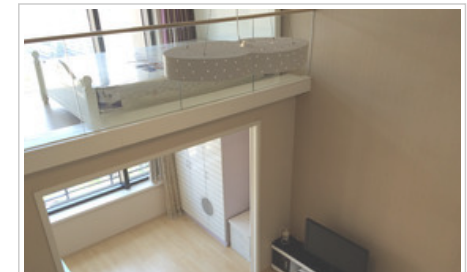


9



这三个东西都是NDK部分：

1. .externalNativeBuild文件夹：cmake编译好的文件, 显示支持的各种硬件等信息。系统生成。
2. cpp文件夹：存放C/C++代码文件，native-lib.cpp文件是该Demo中自带的，可更改。需要自己编写。
3. CMakeLists.txt文件：CMake脚本配置的文件。需要自己配置编写。



loft公寓出租

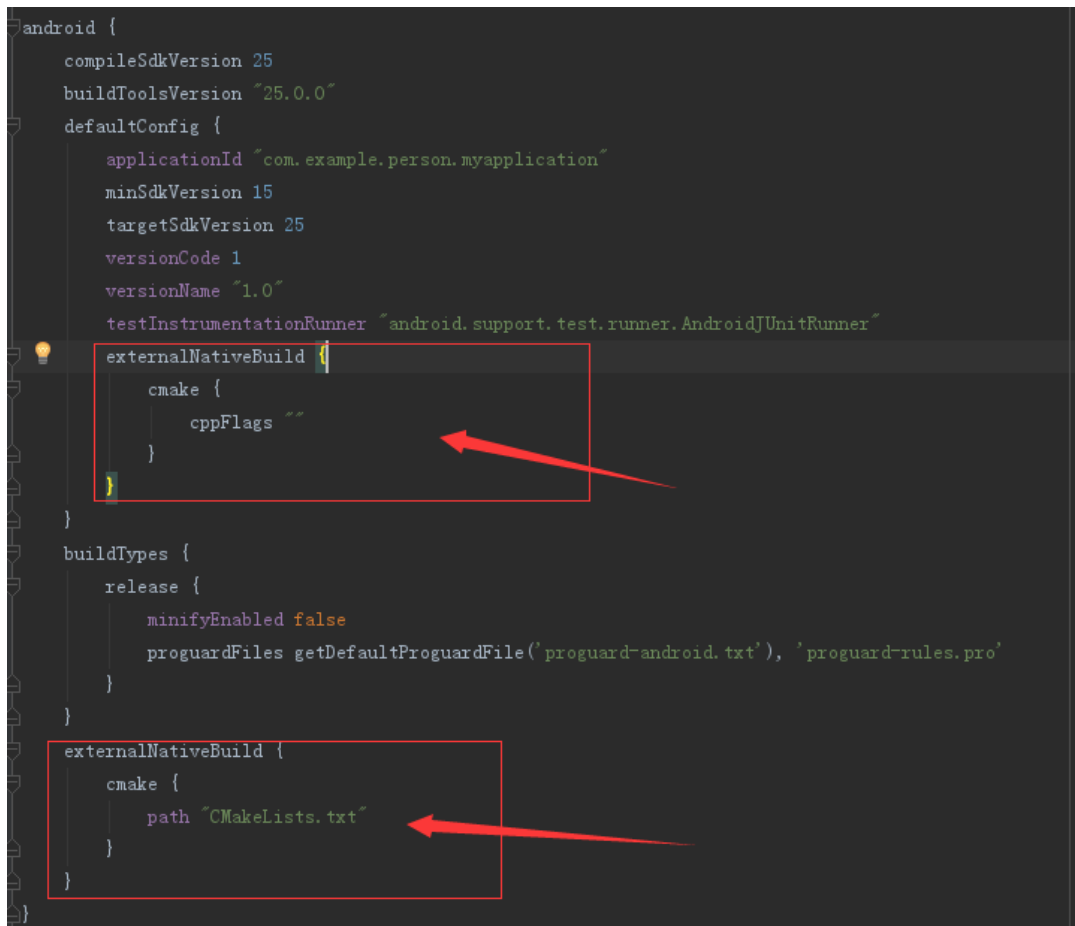


内容举报



返回顶部

Gradle中也有两处不同：



```
android {  
    compileSdkVersion 25  
    buildToolsVersion "25.0.0"  
    defaultConfig {  
        applicationId "com.example.person.myapplication"  
        minSdkVersion 15  
        targetSdkVersion 25  
        versionCode 1  
        versionName "1.0"  
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"  
    }  
    externalNativeBuild {  
        cmake {  
            cppFlags ""  
        }  
    }  
    buildTypes {  
        release {  
            minifyEnabled false  
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'  
        }  
    }  
    externalNativeBuild {  
        cmake {  
            path "CMakeLists.txt"  
        }  
    }  
}
```

java代码：



内容举报

返回顶部

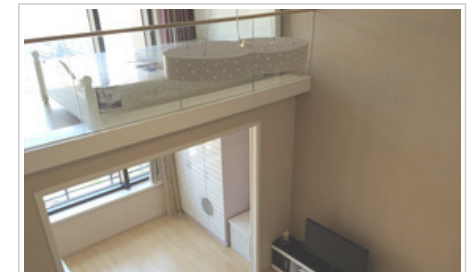


9



```
1 public class MainActivity extends AppCompatActivity {  
2  
3     @Override  
4     protected void onCreate(Bundle savedInstanceState) {  
5         super.onCreate(savedInstanceState);  
6         setContentView(R.layout.activity_main);  
7  
8         // Example of a call to a native method  
9         TextView tv = (TextView) findViewById(R.id.sample_text);  
10        tv.setText(stringFromJNI());  
11    }  
12  
13    /**  
14     * A native method that is implemented by the 'native-lib' native library,  
15     * which is packaged with this application.  
16     */  
17    public native String stringFromJNI();  
18  
19    // Used to load the 'native-lib' library on application startup.  
20    static {  
21        System.loadLibrary("native-lib");  
22    }  
23 }
```

从native-lib.cpp的代码中我们能看到它使用的是静态注册的方式，动态注册的方式代码同传统JNI。



loft公寓出租



内容举报



返回顶部

```
1  #include <jni.h>
2  #include <string>
3
4  extern "C"
5  jstring
6  Java_com_example_person_myapplication_MainActivity_stringFromJNI(
7      JNIEnv* env,
8      jobject /* this */) {
9      std::string hello = "Hello from C++";
10     return env->NewStringUTF(hello.c_str());
11 }
```



9



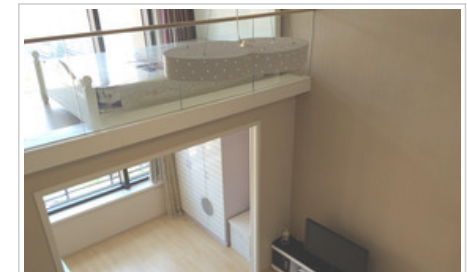
CMakeLists.txt文件中我们需要注意的下面这三个地方。两个library的名字（需一致）以及一个cpp文件的路径，彼此需要对应一致，当我们自己定义library以及自己创建cpp文件时需要对应修改。

```
# Creates and names a library, sets it as either STATIC
# or SHARED, and provides the relative paths to its source code.
# You can define multiple libraries, and CMake builds it for you.
# Gradle automatically packages shared libraries with your APK.

add_library( # Sets the name of the library.
             native-lib
             # Sets the library as a shared library.
             SHARED

             # Provides a relative path to your source file(s).
             # Associated headers in the same location as their source
             # file are automatically included.
             src/main/cpp/native-lib.cpp )

# Searches for a specified prebuilt library and stores the path as a
# variable. Because system libraries are included in the search path by
# default, you only need to specify the name of the public NDK library
# you want to add. CMake verifies that the library exists before
# completing its build.
```



loft公寓出租



内容举报



返回顶部


```
find_library( # Sets the name of the path variable.
             log-lib

             # Specifies the name of the NDK library that
             # you want CMake to locate.
             log )

# Specifies libraries CMake should link to your target library. You
# can link multiple libraries, such as libraries you define in the
# build script, prebuilt third-party libraries, or system libraries.

target_link_libraries( # Specifies the target library.
                      native-lib
                      # Links the target library to the log library
                      # included in the NDK.
                      ${log-lib} )
```



loft公寓出租



内容举报



返回顶部

其中，add_library方法和target_link_libraries方法一一对应，当有多个库，多个C/C++文件时，可写多组add_library方法和target_link_libraries方法。如下图（删除了注释部分）。

```
cmake_minimum_required(VERSION 3.4.1)

add_library(native-lib
    SHARED
    src/main/cpp/native-lib.cpp )
add_library(native-lib2
    SHARED
    src/main/cpp/native-lib-second.cpp)

find_library(log-lib
    log )

target_link_libraries(native-lib
    ${log-lib} )

target_link_libraries(native-lib2
    ${log-lib} )
```

http://blog.csdn.net/xiaoyu_93



运行代码，就能看到效果，调用了C++方法在界面上显示了 Hello from C++ 字符串。这就是CMake方式进行NDK开发的Demo。

还记得传统JNI方式中吗？我们使用了ndk-build来编译C/C++文件为so文件，对于使用过第三方库的开发者来说，对so肯定不陌生。我们只能使用别人给的so文件，而无法看到C/C++源码，更别说去修改了。但是在这里我们好像一直没看到so文件的影子，那么，我们安装运行的apk中，有对应的so文件吗？如果想验证一下apk是否有so文件，我们可以使用 APK Analyzer:

1. 选择 Build > Analyze APK。
2. 选择 apk，并点击 OK。

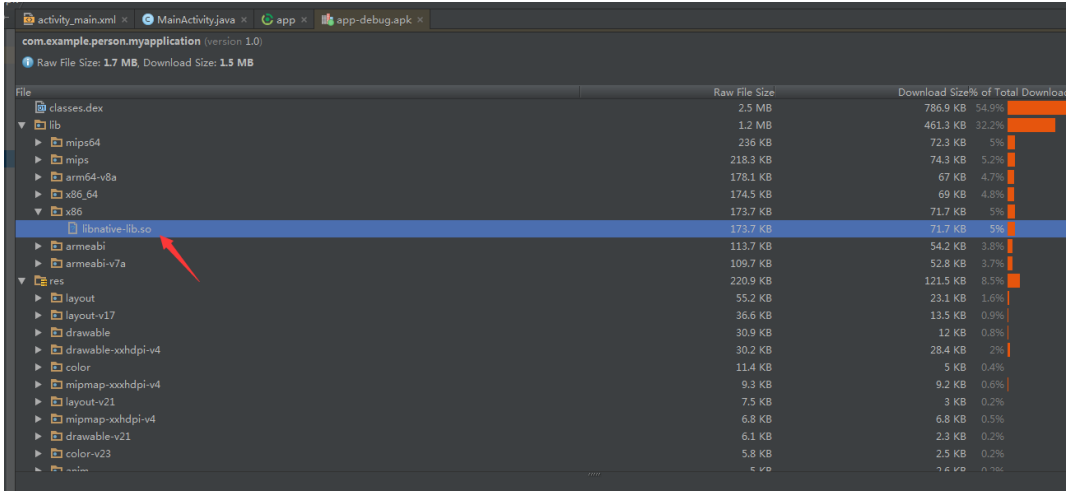


内容举报



返回顶部

3. 如下图，在 APK Analyzer 窗口中，选择 lib/x86/，可以看见 libnative-lib.so。



Tips：Instant Run 并不兼容使用了 native code 的项目。Android Studio 会自动禁止 Instant Run 功能。

但是我个人目前更关注的问题是C++代码自动补全提示的功能，当在cpp文件中写代码时，里面不再是一大片的红色，输入若干个字母的时候，也能给出提示了，简直叼渣天，至于原因...我就不清楚了，为什么在这里就有这个强大的功能了呢？哪位大神能解释下...



内容举报

返回顶部



```
#include <jni.h>
#include <string>

extern "C"
jstring
Java_com_example_person_myapplication_MainActivity_stringFromJNI(
    JNIEnv* env,
    jobject /* this */) {
    std::string hello = "Hello from C++";

    env->Fi
```



CMake编译so文件

在Android Studio的Terminal中输入cmake，会提示：

1	'cmake' 不是内部或外部命令，也不是可运行的程序或批处理文件。
---	------------------------------------


内容举报


返回顶部

需要在下载配置好CMake的环境变量，若没这个问题可跳过这个步骤。在如下网址中下载好对应的文件：
<https://cmake.org/download/> (<https://cmake.org/download/>)

Release Candidate (3.7.0-rc3)

The release was packaged with CPack which is included as part of the release. The .sh files are self extracting gzipped tar files. To install a .sh file, run it with /bin/sh and follow the directions. The OS-machine.tar.gz files are gzipped tar files of the install tree. The OS-machine.tar.Z files are compressed tar files of the install tree. The tar file distributions can be untared in any directory. They are prefixed by the version of CMake. For example, the Linux-x86_64 tar file is all under the directory cmake-Linux-x86_64. This prefix can be removed as long as the share, bin, man and doc directories are moved relative to each other. To build the source distributions, unpack them with zip or tar and follow the instructions in Readme.txt at the top of the source tree. See also the [CMake 3.7 Release Notes](#). Source distributions:

Platform	Files
Unix/Linux Source (has \n line feeds)	cmake-3.7.0-rc3.tar.gz cmake-3.7.0-rc3.tar.Z
Windows Source (has \r\n line feeds)	cmake-3.7.0-rc3.zip

Binary distributions:

Platform	Files
Windows win64-x64 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.7.0-rc3-win64-x64.msi
Windows win64-x64 ZIP	cmake-3.7.0-rc3-win64-x64.zip
Windows win32-x86 Installer: Installer tool has changed. Uninstall CMake 3.4 or lower first!	cmake-3.7.0-rc3-win32-x86.msi
Windows win32-x86 ZIP	cmake-3.7.0-rc3-win32-x86.zip
Mac OSX 10.6 or later	cmake-3.7.0-rc3-Darwin-x86_64.dmg cmake-3.7.0-rc3-Darwin-x86_64.tar.gz
Linux x86_64	cmake-3.7.0-rc3-Linux-x86_64.sh cmake-3.7.0-rc3-Linux-x86_64.tar.gz

下载zip包后，解压缩。然后将cmake.exe所在的路径，添加到path环境变量中。cmake就成为有效命令了。

在CMakeLists.txt文件上级目录下执行 cmake . 命令，报错，暂未解决：

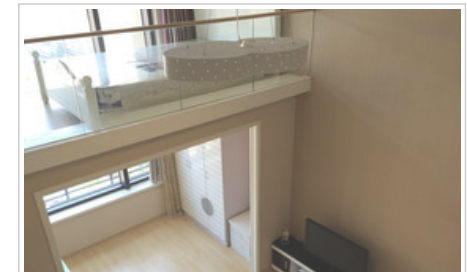




9



```
1  -- Building for: NMake Makefiles
2  -- The C compiler identification is unknown
3  -- The CXX compiler identification is unknown
4  CMake Error in CMakeLists.txt:
5    The CMAKE_C_COMPILER:
6
7     cl
8
9    is not a full path and was not found in the PATH.
10
11   To use the NMake generator with Visual C++, cmake must be run from a shell
12   that can use the compiler cl from the command line. This environment is
13   unable to invoke the cl compiler. To fix this problem, run cmake from the
14   Visual Studio Command Prompt (vcvarsall.bat).
15
16   Tell CMake where to find the compiler by setting either the environment
17   variable "CC" or the CMake cache entry CMAKE_C_COMPILER to the full path to
18   the compiler, or to the compiler name if it is in the PATH.
19
20
21  CMake Error in CMakeLists.txt:
22    The CMAKE_CXX_COMPILER:
23
24     cl
25
26    is not a full path and was not found in the PATH.
27
28   To use the NMake generator with Visual C++, cmake must be run from a shell
29   that can use the compiler cl from the command line. This environment is
30   unable to invoke the cl compiler. To fix this problem, run cmake from the
31   Visual Studio Command Prompt (vcvarsall.bat).
32
33   Tell CMake where to find the compiler by setting either the environment
34   variable "CXX" or the CMake cache entry CMAKE_CXX_COMPILER to the full path
35   to the compiler, or to the compiler name if it is in the PATH.
36
37
```



loft公寓出租



内容举报



返回顶部

```
38 | -- Configuring incomplete, errors occurred!  
39 | See also "E:/AndroidDemo_XY/app/CMakeFiles/CMakeOutput.log".  
40 | See also "E:/AndroidDemo_XY/app/CMakeFiles/CMakeError.log".
```

待完成

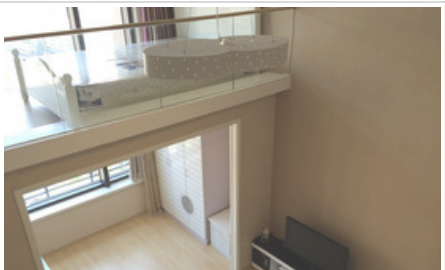
- 👍
- 9
- ☰
- 🔖
- 💬
- 🔗

CMake和传统 JNI在目录结构和配置文件上的区别

现在我们有了CMake和传统JNI两种开发NDK的方法，它们在目录结构和Gradle上有所区别，下面我们将分别介绍目录区别和Gradle配置的区别。

一、目录结构

传统JNI

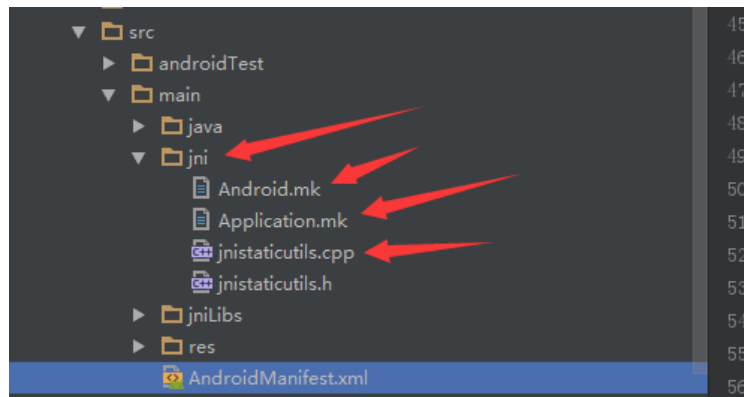


loft公寓出租

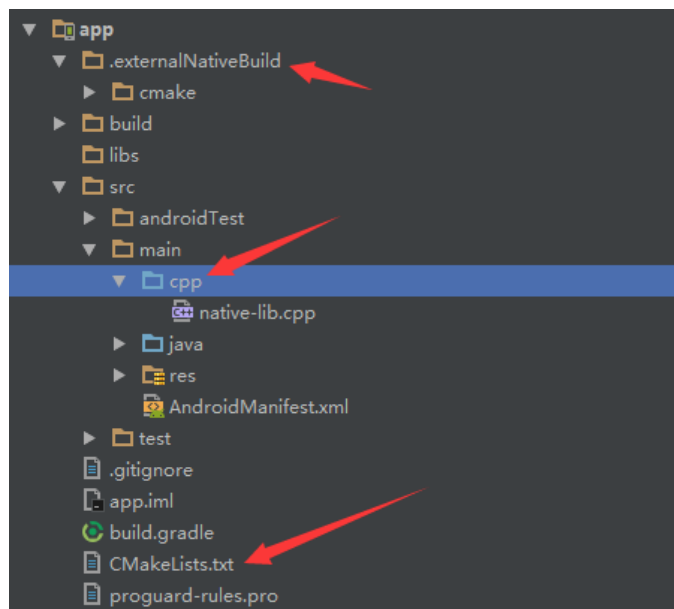




9



CMake



loft公寓出租



内容举报



返回顶部

这两种方式在目录上的区别就是两点：

1. 以前的jni目录改成cpp，名字更换了，下面还是存放C/C++文件。
2. 之前对C/C++文件的编译配置Android.mk、Application.mk文件放在jni目录下，现在改成CMakeLists.txt文件。（事实上这些文件的位置是可任意存放的，只需要配置好就行。但最好还是按照默认习惯放置。）

二、Gradle

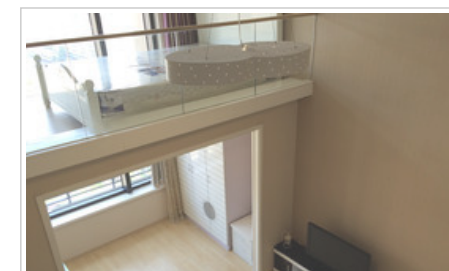
传统JNI

```
defaultConfig {
    applicationId "com.example.person.myapplication"
    minSdkVersion 15
    targetSdkVersion 25
    versionCode 1
    versionName "1.0"

    // NDK
    ndk {
        moduleName "jnistaticutils"
        abiFilters "armeabi", "armeabi-v7a", "x86" //输出指定三种架构下的so库
    }
}

buildTypes {
    release {
        minifyEnabled false
        proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
    }
}

externalNativeBuild {
    ndkBuild {
        path "src/main/jni/Android.mk"
    }
}
```



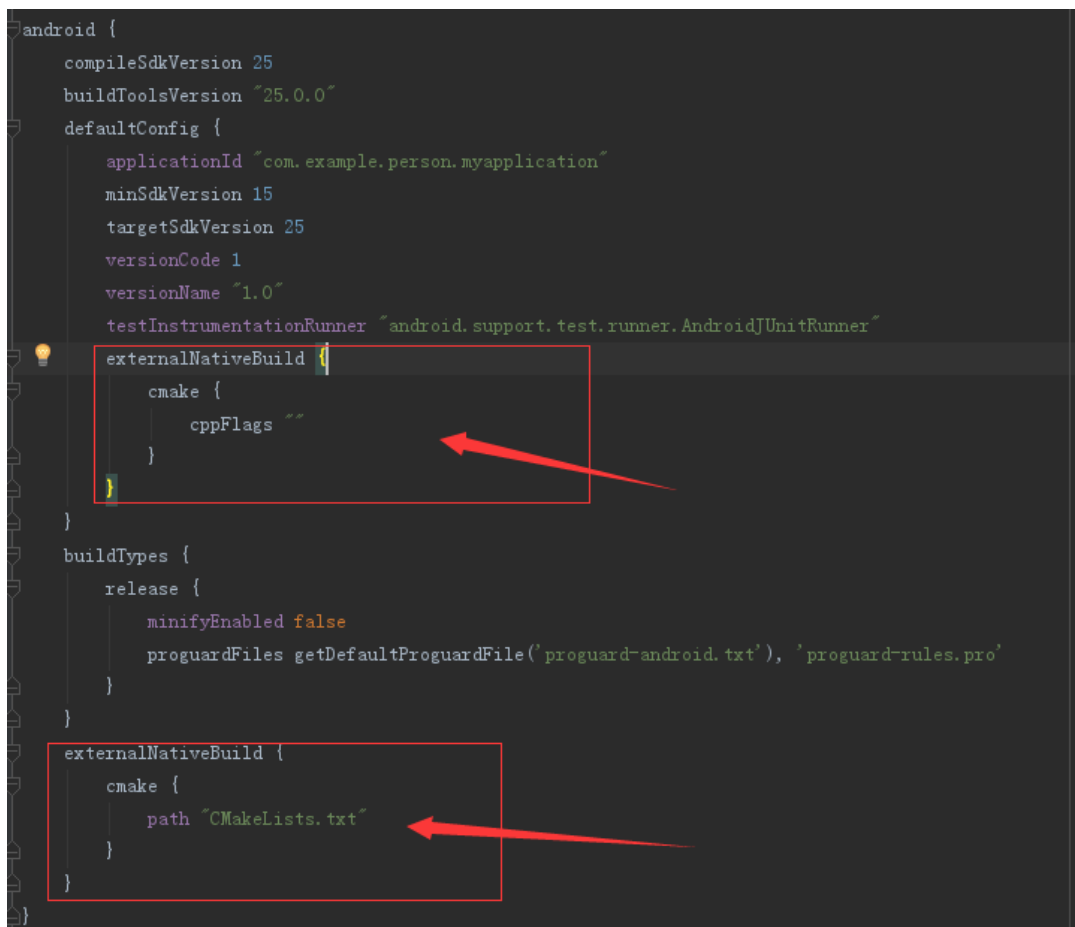
loft公寓出租



内容举报

返回顶部

CMake



```
android {
    compileSdkVersion 25
    buildToolsVersion "25.0.0"
    defaultConfig {
        applicationId "com.example.person.myapplication"
        minSdkVersion 15
        targetSdkVersion 25
        versionCode 1
        versionName "1.0"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    externalNativeBuild {
        cmake {
            cppFlags ""
        }
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
    externalNativeBuild {
        cmake {
            path "CMakeLists.txt"
        }
    }
}
```

事实上，我们在使用传统JNI方式的时候，上面的两处地方我们都省略了，也不会造成什么异常（若Android.mk存放位置在其他地方则需要配置）。CMake方式中第一处也可省略，但是第二处不能省略（同样，位置路径要写对），因为没有它将无法Build生成externalNativeBuild文件夹。

另外，传统JNI开发还需要在项目根目录下的gradle.properties文件中配置



内容举报

返回顶部

```
1 android.useDeprecatedNdk=true
```

否则Build项目的时候会报错。

CMake的优势

1. 可以直接的在C/C++代码中加入断点，进行调试
2. java引用的C/C++中的方法，可以直接ctrl+左键进入
3. 对于include的头文件或者库，也可以直接进入
4. 不需要配置命令行操作，手动的生成头文件，不需要配置android.useDeprecatedNdk=true属性

普通Android项目转NDK开发项目

我们之前说了，在创建新project时若勾选了 Include C++ Support 选项（需Android Studio版本不低于2.2），则该项目就已经是一个NDK开发项目了。那么，若项目已经存在，该如何再转为NDK开发项目呢？

Tips: 在配置好NDK的前提下。且各种地方的配置均不再赘述。如Android.mk中如何配置、CMakeLists.txt文件中的配置、各个地方的统一等。

传统JNI方式，在NDK开发 从入门到放弃(一：基本流程入门了解)

(http://blog.csdn.net/xiaoyu_93/article/details/52870395)中我们了解过：

1. 新建jni目录，写好C/C++代码。静态注册JNI时我们使用了javah -jni对JAVA类进行操作自动生成了jni目录以及对应的头文件（事实上，当我们有一定经验后可以自己写，而不再需要使用该辅助命令来保证不写错，另外动态注册也是一个很值得提倡的方式），然后根据头文件写了C/C++代码。但在动态注册JNI时我们可以自己先创建好jni目录且写好C/C++代码。
2. 在jni目录下创建且配置好Android.mk和Application.mk两个文件。
3. build.gradle文件中根据情况进行配置，可不进行配置使用默认值。



内容举报

返回顶部

4. 通过ndk-build操作，我们能得到对应的so文件，放置在相应位置，java代码中即可调用C/C++代码，运行程序。


回顾前面提到的CMake方式的NDK开发，我们得到如下步骤：

1. 新建cpp目录，写好C/C++代码。
2. 创建且配置CMakeLists.txt文件。
3. build.gradle文件中根据情况进行配置，CMakeLists.txt文件的路径必须配置。
4. java代码中即可调用C/C++代码，运行程序。
5. project的build.gradle文件中，gradle版本不能低于2.2，否则会报错。

```
Error: Could not find method externalNativeBuild() for arguments
[build_e8k5d0w74qu1ajxr9snog$_run_closure1$_closure6@1398219d] on
project ':app' of type org.gradle.api.Project.。
```


版权声明：本文为博主原创文章，未经博主允许不得转载。



 jinsheng713 (/jinsheng713) 4天前 16:37

(/jinsheng713) studio 3.0下建demo只能调试java部分，native调试不了

回复

 C_Seerhuang (/C_Seerhuang) 2017-08-23 18:07

(/C_Seerhuang) 楼猪就算没有CMake，你为什么还不考虑Swig呢？是不知道还是有什么不好用的地方？

内容举报

返回顶部

回复



LSCCMQ (/LSCCMQ) 2017-08-11 10:10

11楼

(/LSCCMQ)CMakeLists.txt文件上级目录下执行cmake .命令，报错，暂未解决：

-- Building for: NMake Makefiles
-- The C compiler identification is unknown
-- The CXX compiler identification is unknown
CMake Error in CMakeLists.txt:
The CMAKE_C_COMPILER:


回复

查看 31 条热评

相关文章推荐


CMake构建NDK (/yuankundong/article/details/52839460)

Android Studio 2.2提供了更好用的NDK工具，使JNI开发更加容易，简化开发过程，而且可以调试了，感觉很不错。介绍在下载ndk的时候我们同时下载了CMake和LLDB，CMake是构建...

 yuankundong (<http://blog.csdn.net/yuankundong>) 2016-10-17 16:04 1525

AndroidStudio2.2 Preview3中NDK开发之CMake和传统 JNI在目录结构和配置文件上的区别 (...)

AndroidStudio2.2 CMake 和传统 JNI 两种开发NDK的方法相关知识

 u012527560 (<http://blog.csdn.net/u012527560>) 2016-06-24 13:36 6951



个人开发者如何通过人工智能盈利？

个人如何开发一款人工智能应用？个人如何利用免费的人工智能工具与平台赚钱？



loft公寓出租





内容举报


返回顶部

(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjcvPW60lZ0qnfK9ujYzP1fVn1DL0Aw-5Hc4nHb3rjD0TAq15HfLPWRznjb0T1Y1PHbvHuhn1w-rjKhrAf40AwY5HDdnH0YrHc1P1D0lgF_5y9YIZ0IQzqMpgwBUvqoQhP8QvIGIAPCmgfEmyPYpGuGIZbEPH--ujN-njT4rH7buWTsuyPbPyDsnyRz5LNYUNq1ULNzmvRqnHDknAwBUAqM0ZFb5HD0mhYqn0KsTWYs0ZNGujYkPHTYn1mk0AqGujY

使用CMake进行android native开发 (/huanghai381/article/details/51547348)

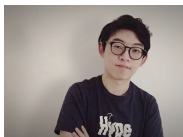
Android NDK中提供了ndk-build脚本，以及若干mk文件，以简化ndk的开发，这对于开发一些小型应用来说足够了.如果对于一些大型工程特别是开源工程，人家是要cmake来组织的该怎么编译呢...

 huanghai381 (http://blog.csdn.net/huanghai381) 2016-05-31 17:16 7458

Android Studio 2.3使用CMake方式NDK入门 (/aidesudi/article/details/68946918)

参考官网：https://developer.android.com/studio/projects/add-native-code.html#new-project 使用CMake需要搭配使用an...

 aidesudi (http://blog.csdn.net/aidesudi) 2017-04-01 18:34 3355




Webpack 前端工程化入门

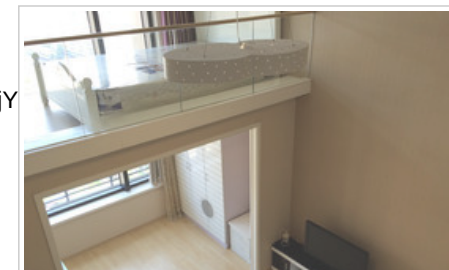
如果你想了解在公司级别的大型项目中是如何做工程化的？如果你想深入了解 Webpack，并且想用它的各种特性来提升构建效率？那么你就有理由来了解一下这门课。

(http://www.baidu.com/cb.php?c=lgF_pyfqHmknjcvPjR0lZ0qnfK9ujYzP1ndnHfY0Aw-5Hc4nj6vPjm0TAq15Hf4rjn1n1b0T1YsPjDYmWNBuHcLnvN-uHR40AwY5HDdnH0YrHc1P1D0lgF_5y9YIZ0IQzqMpgwBUvqoQhP8QvIGIAPCmgfEmvq_lyd8Q1R4uH0vPymvPAmLuhF-PHRduHfLrHcsPjqdIAdxTvqdThP-5HDknHK3mhkEusKzujYk0AFV5H00TZcqN0KdpyfqNHRLPjnvnfKEpyfqNHnsnj0YnsKWpyfqP1cvrHnz0AqLUWYs0ZK45HcsP6KWThnqPWDkPWD)

android studio 2.2 使用cmake编译NDK (/lmw21848/article/details/52183584)

android studio 2.2 已经进入beta版本，新功能添加众多，NDK编程也得到了简化。官方博客介绍。本文介绍如何使用新版android studio调用 c++代码，为了超级通俗易懂，例...

 lmw21848 (http://blog.csdn.net/lmw21848) 2016-08-11 17:51 11365



loft公寓出租



「ZbqnW0v


内容举报


返回顶部

android studio 2.2 C++ 支持（CMAKE方法）(yangzhengqui/article/details/51544687)

上次学到了约束布局，现在继续学习android studio 2.2使用CMAKE编译C++，并支持在C++打断点，写C++代码时也支持打断点。学习的时候是使用了国外的一篇博客的代码：<http://b...>

 yangzhengqui (<http://blog.csdn.net/yangzhengqui>) 2016-05-31 12:47 16274


CMake语法简介（androidstudio中利用CMake开发NDK）(ymeddmn/article/details/53543...

1, file(WRITE filename "message to write"...) WRITE将"message to write"写入filename文件中，如果filename文件不存在则...

 ymeddmn (<http://blog.csdn.net/ymeddmn>) 2016-12-16 13:27 534


sdk tools没有cmake下载项的解决办法。(u014290233/article/details/53418371)

sdk tools没有cmake下载项的解决办法。

 u014290233 (<http://blog.csdn.net/u014290233>) 2016-12-01 09:43 2562

android studio 使用cmake支持C++出现无法sync cmake error的问题 (qq_29343201/article...

问题错误信息External Native Build Issues Build command failed. Error while executing '/root/Android/Sdk/cm...

 qq_29343201 (http://blog.csdn.net/qq_29343201) 2016-09-30 23:55 8779

Android studio sdk tools 无法更新 cmake和LLDB方法 (yang19910524/article/details/5651...

解决Android studio sdk tools没有cmake 更新问题

 yang19910524 (<http://blog.csdn.net/yang19910524>) 2017-02-22 23:27 1690

NDK开发 从入门到放弃(七：Android Studio 2.2 CMAKE 高效NDK开发) (cuiyufeng2/article...

更深入的了解如何开发NDK




内容举报


返回顶部



cuiyufeng2 (<http://blog.csdn.net/cuiyufeng2>) 2017-03-20 14:45 2918

使用AndroidStudio编译NDK的方法及错误解决方案 (/asmcvc/article/details/24457677)

参考资料：【android ndk】macos环境下Android Studio中利用gradle编译jni模块及配置：<http://demo.netfoucs.com/ashqal/article...>



asmcvc (<http://blog.csdn.net/asmcvc>) 2014-04-25 10:34 132212

Android Studio - Failed to find CMake问题 (/v2810769/article/details/75531067)

解决方法只要更新到最新的ndk的库就可以了，这里提供几个下载地址2017/8/14日最新NDK r13bWindows 32-bit android-ndk-r13b-windows-x86.z...



v2810769 (<http://blog.csdn.net/v2810769>) 2017-07-20 15:47 928

cmake编译问题解决 (/u010098331/article/details/50924712)

2 安装cmake软件包 yum install cmake 3 create account of mysql groupadd mysql useradd -g ...



u010098331 (<http://blog.csdn.net/u010098331>) 2016-03-18 16:30 19704

Android Studio NDK 构建工具 CMake初探 (/s402178946/article/details/54140271)

CMake 是 Android studio 支持的一种新的NDK构建工具，这篇文章主要是对官方介绍的一个翻译和一些简单的尝试。



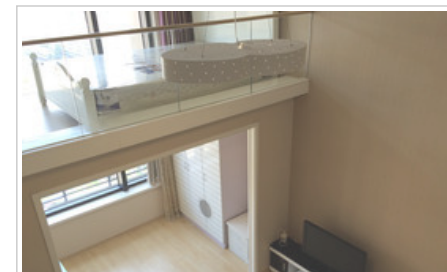
s402178946 (<http://blog.csdn.net/s402178946>) 2017-01-06 17:20 3311

Android studio ndk配置cmake开发native C (/tom_221x/article/details/61210497)

Android 2.2 以后的版本对NDK的支持已经非常好了。最近把一个纯C的android项目，从eclipse ADT迁移到Android studio上。本文是参考 Add C and C++...



tom_221x (http://blog.csdn.net/tom_221x) 2017-03-10 22:38 1820



loft公寓出租



内容举报



返回顶部

基于CMake的JNI开发探索 (/heccjoel/article/details/57085351)

基于CMake的JNI开发探索



heccjoel (<http://blog.csdn.net/heccjoel>) 2017-02-25 17:22 1175



9



android cmake 调试 (/renfujiang/article/details/54649223)

Android jni 支持 Cmake，新建的jni工程会自动的生成一个默认的CmakeList.txt文件，是和build.gradle 在同一个目录，但是感觉不爽，就决定一到cpp 目录和源代码...



renfujiang (<http://blog.csdn.net/renfujiang>) 2017-01-21 23:10 1450

Android Studio 开发SDL2.0最佳实践 - 可以直接编译和调试了，酸爽！ (/gongxp123456/artic...

本文2017-3-14 传统的使用NDK按照命令行来编译SDL2以及自己的C文件，再导入到Android Studio工程编译成为APK，这样十分麻烦，没有代码提示，做个Hello World还可...



gongxp123456 (<http://blog.csdn.net/gongxp123456>) 2017-03-16 11:12 1372



loft公寓出租



内容举报



返回顶部