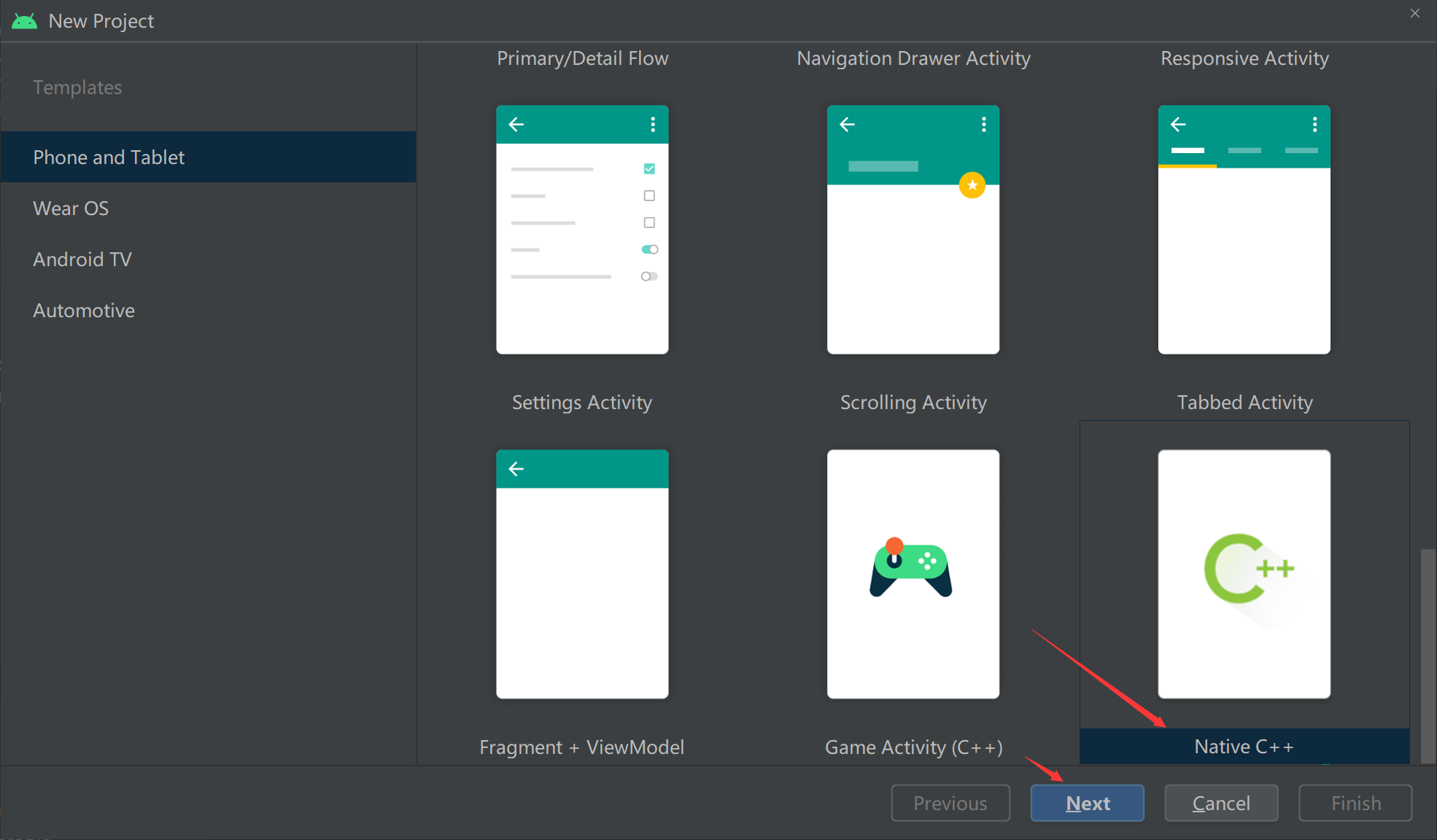
## 背景

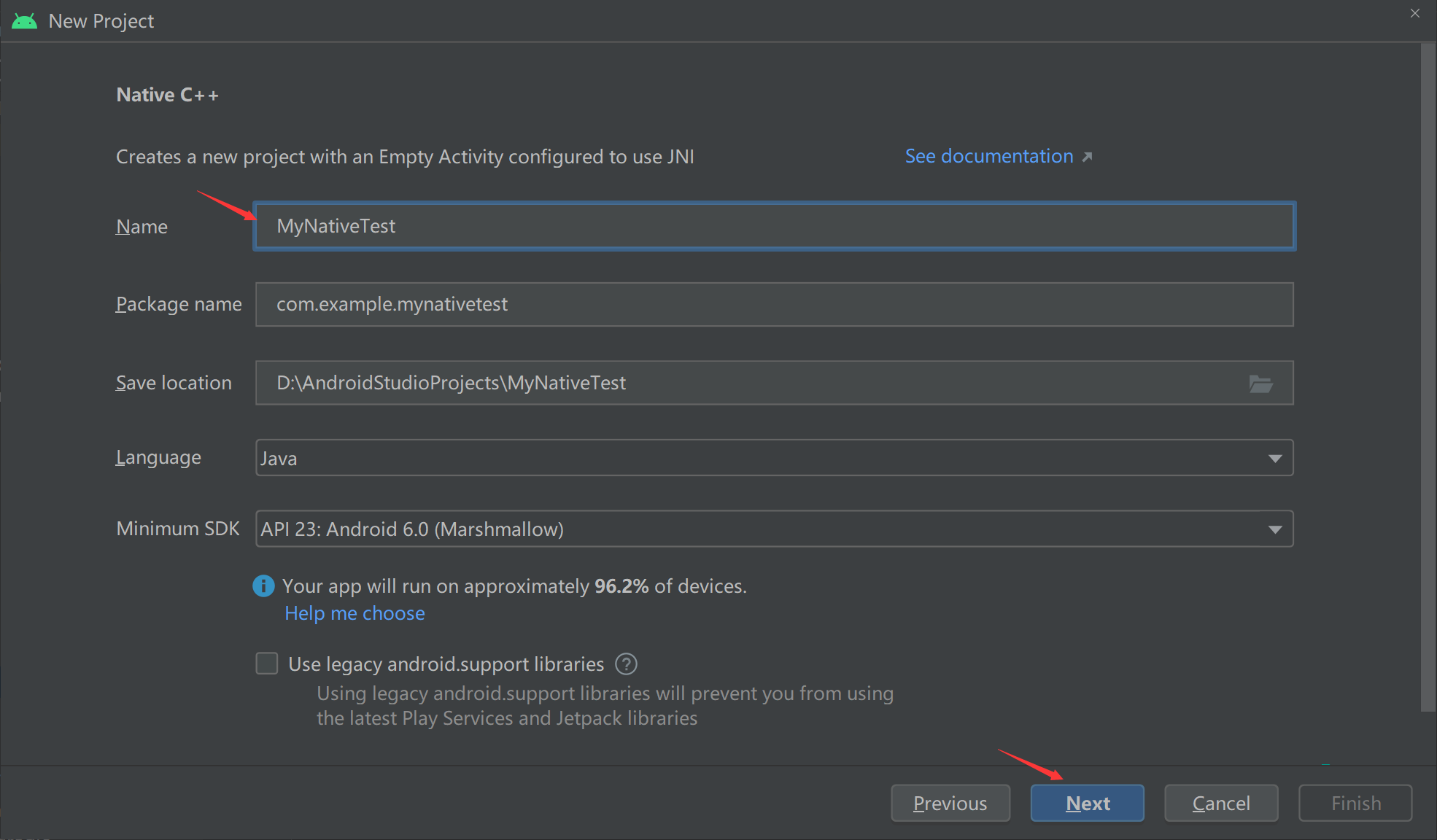
在项目需求中，有些业务只能用C/C++实现以获得更佳的性能和效果，所以需要使用JNI进行编程

## 新建Native项目

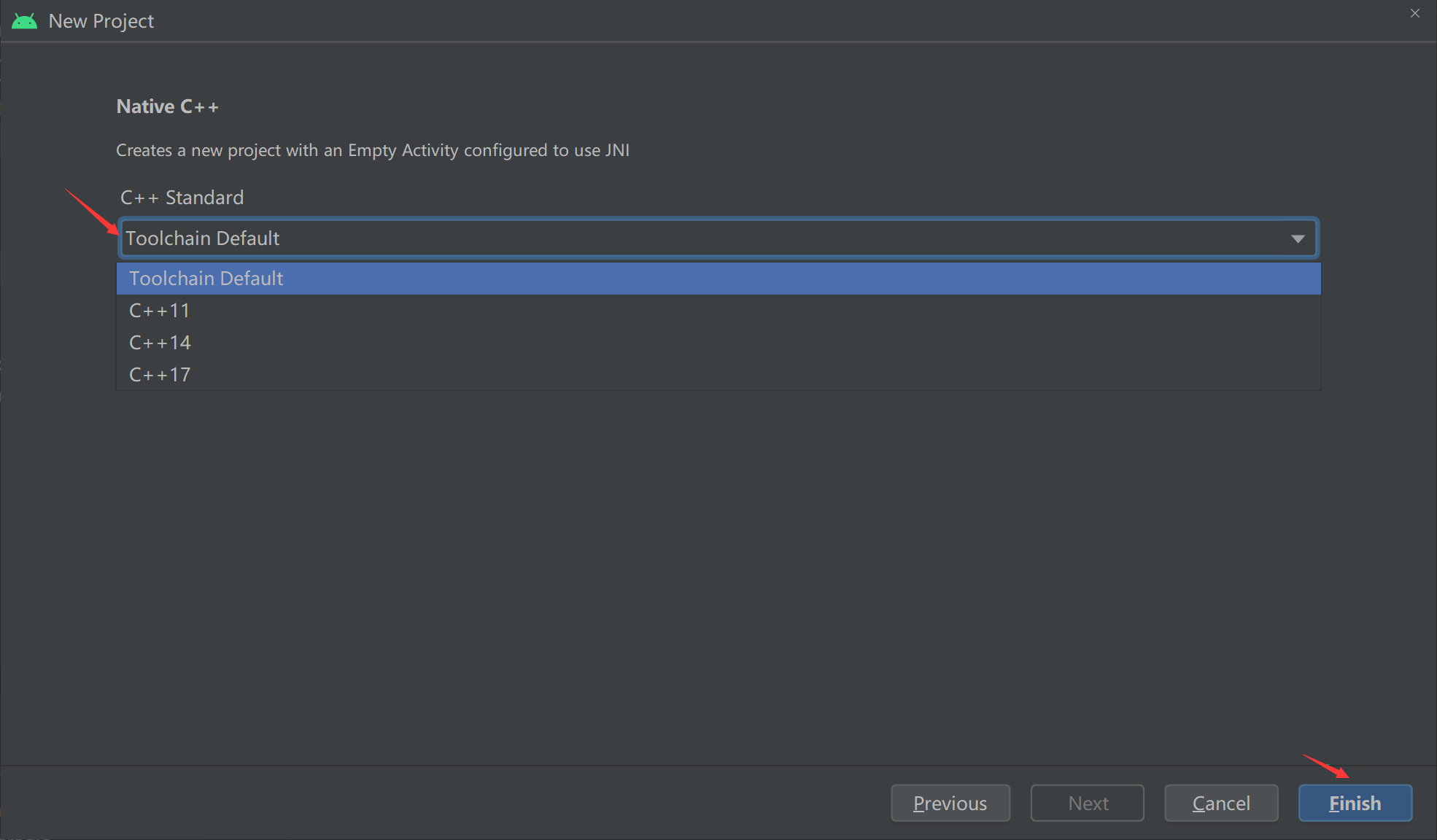
1. File -> New Project -> Native C++，然后点击下一步



1. 填写工程名称，点击Next

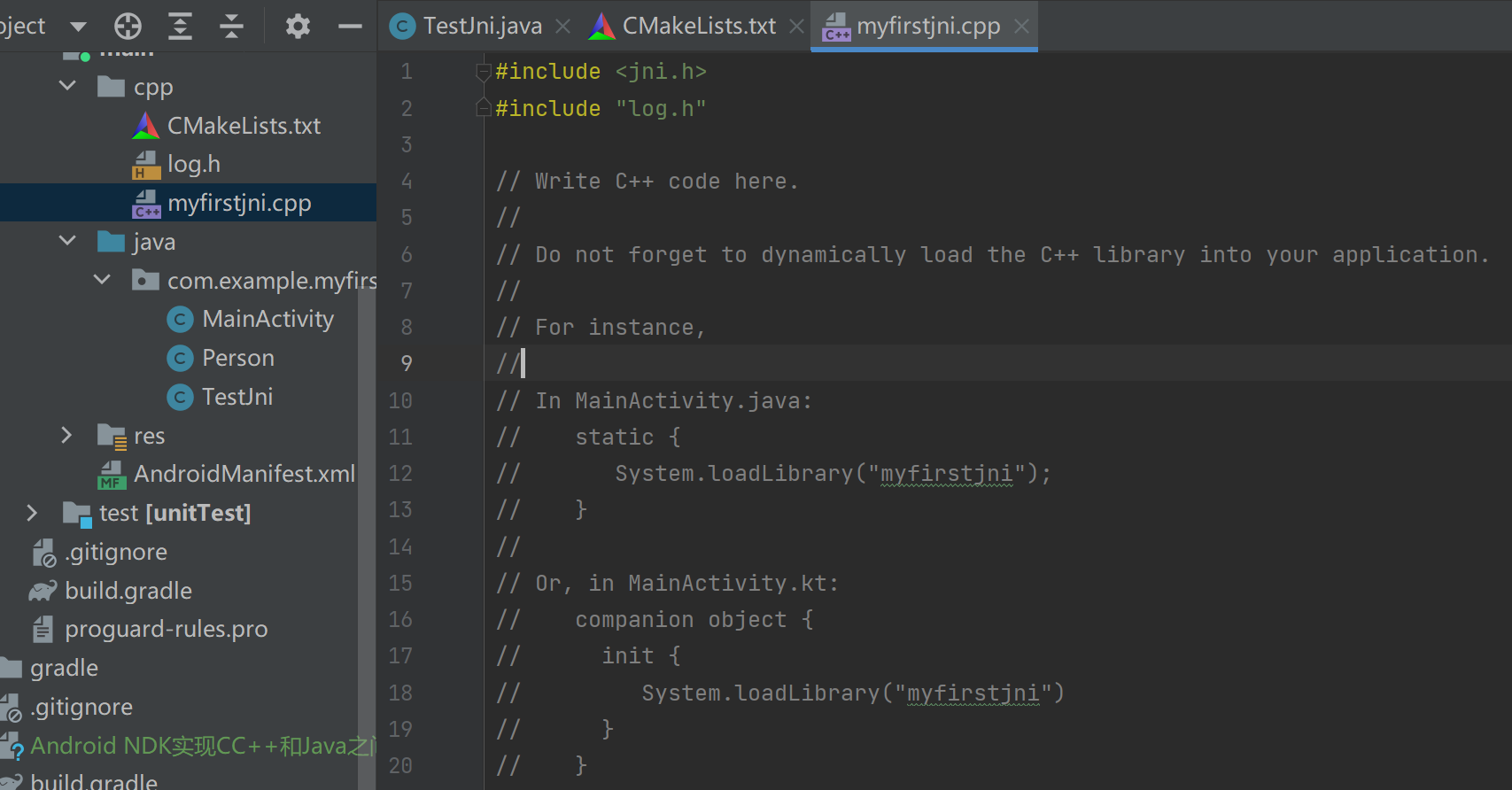


1. 选择C++ standard，这里选择的是工具链默认的，也可以使用其他版本的C++，看个人会使用哪个版本的C++，点击Finish就创建好了



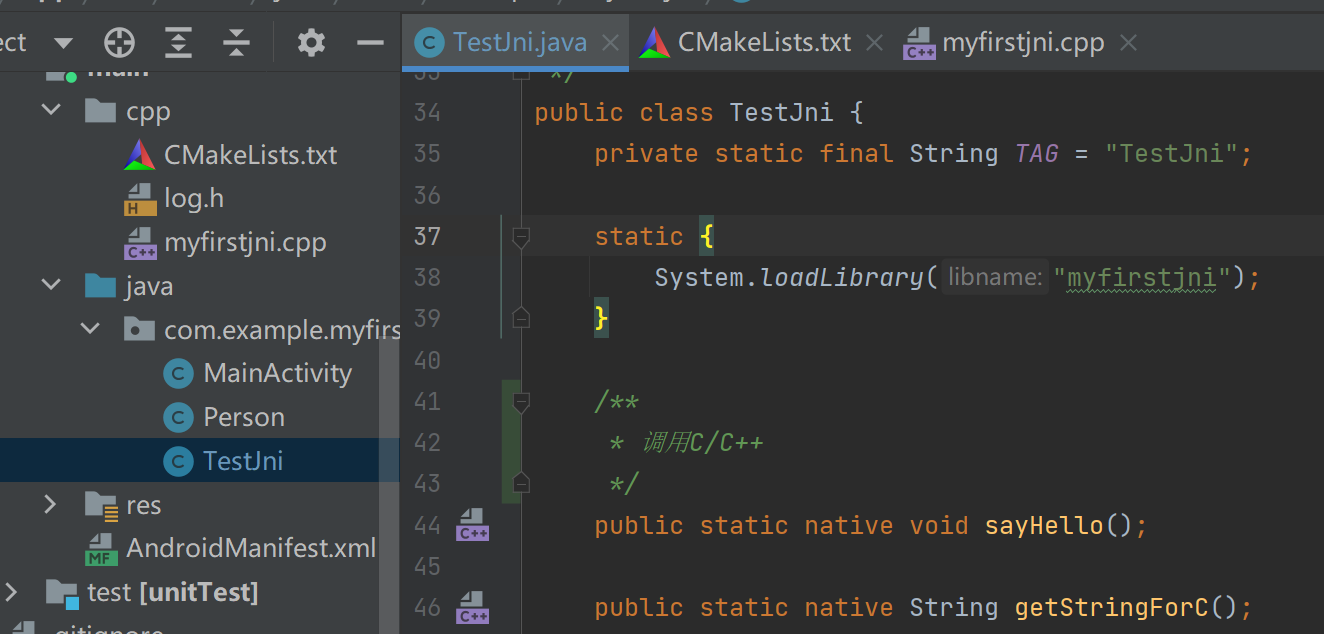
## C/C++和Java相互调用

新建工程后，项目目录cpp路径下的myfirstjni.cpp文件注释中有事例代码，如何加载C/C++编译好的so库

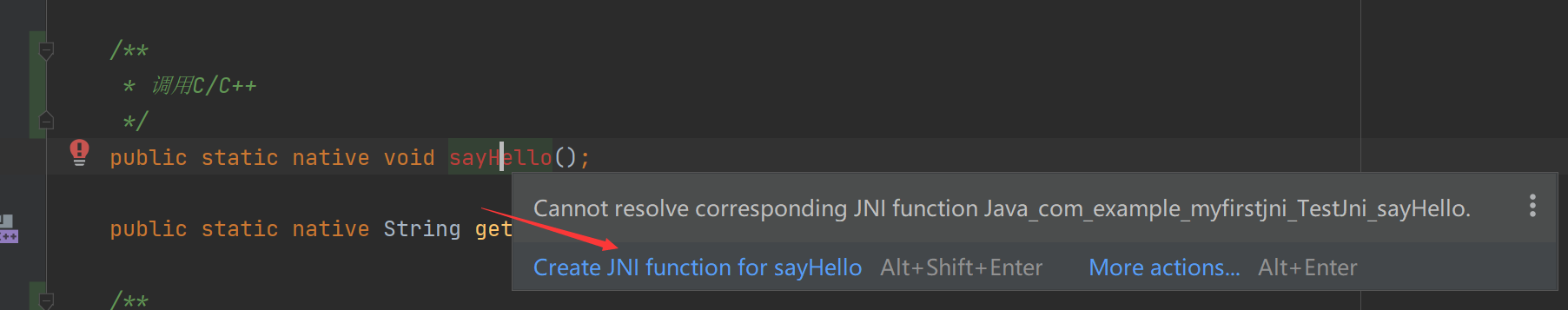


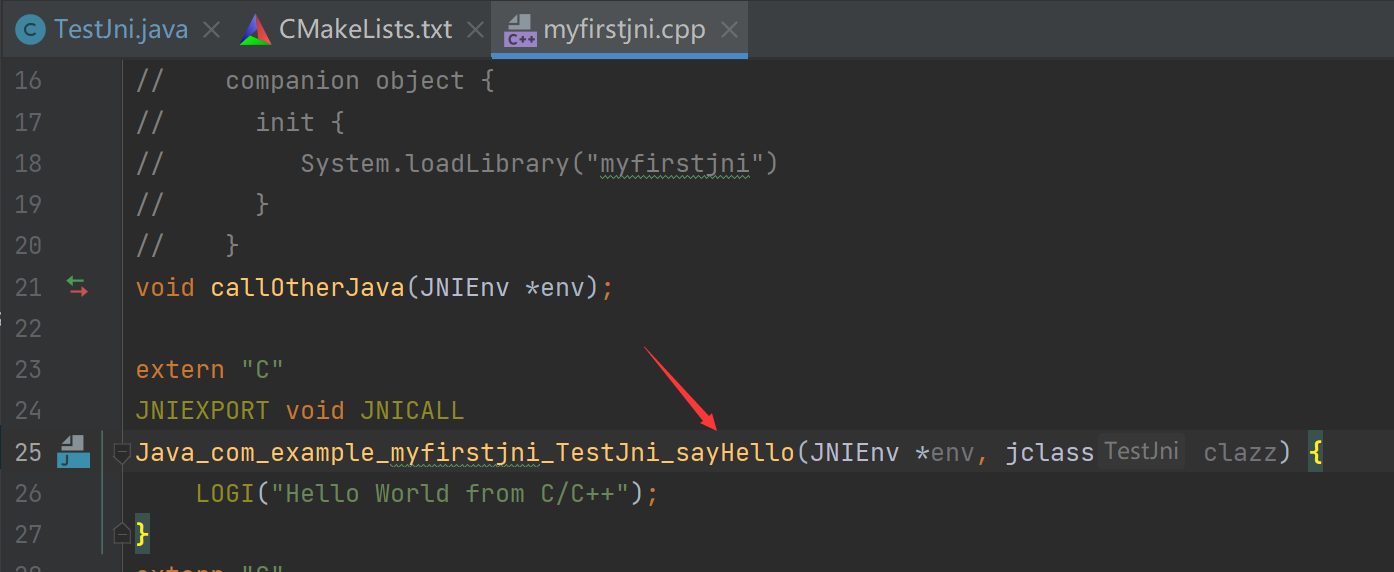
1. **Java调用C/C++代码**

新建TestJni类，增加sayHello和getStringForC两个Native方法

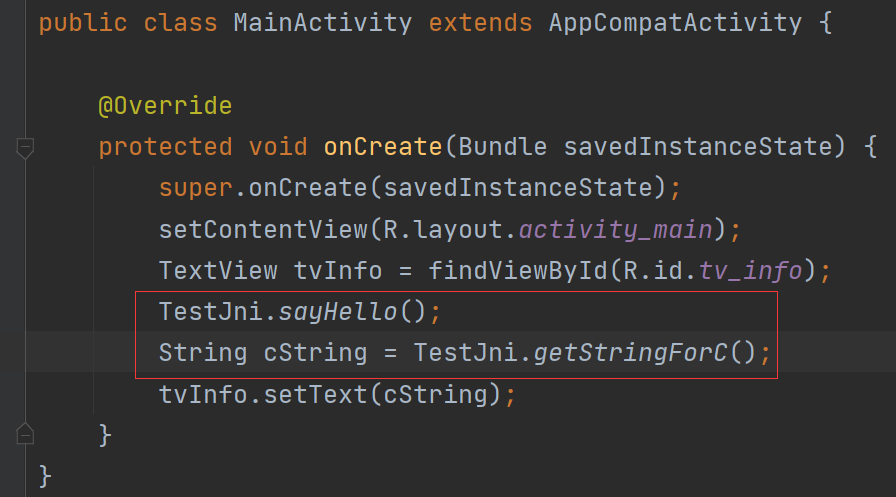


新增方法在myfirstjni.cpp文件中无该方法映射则会出现如下图错误，点击Create JNI function for sayHello即可在myfirstjni.cpp中生成映射方法

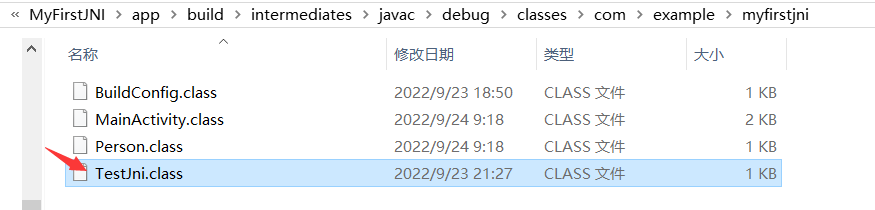




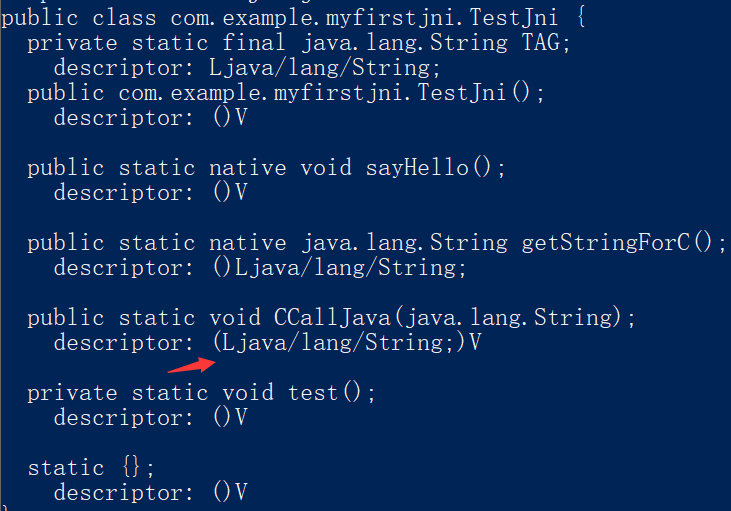
在MainActivity中进行调用C/C++代码



1. **C/C++调用Java代码**
2. **在项目app -> build -> intermediates -> javac目录下，是项目java文件编译生成的class文件**

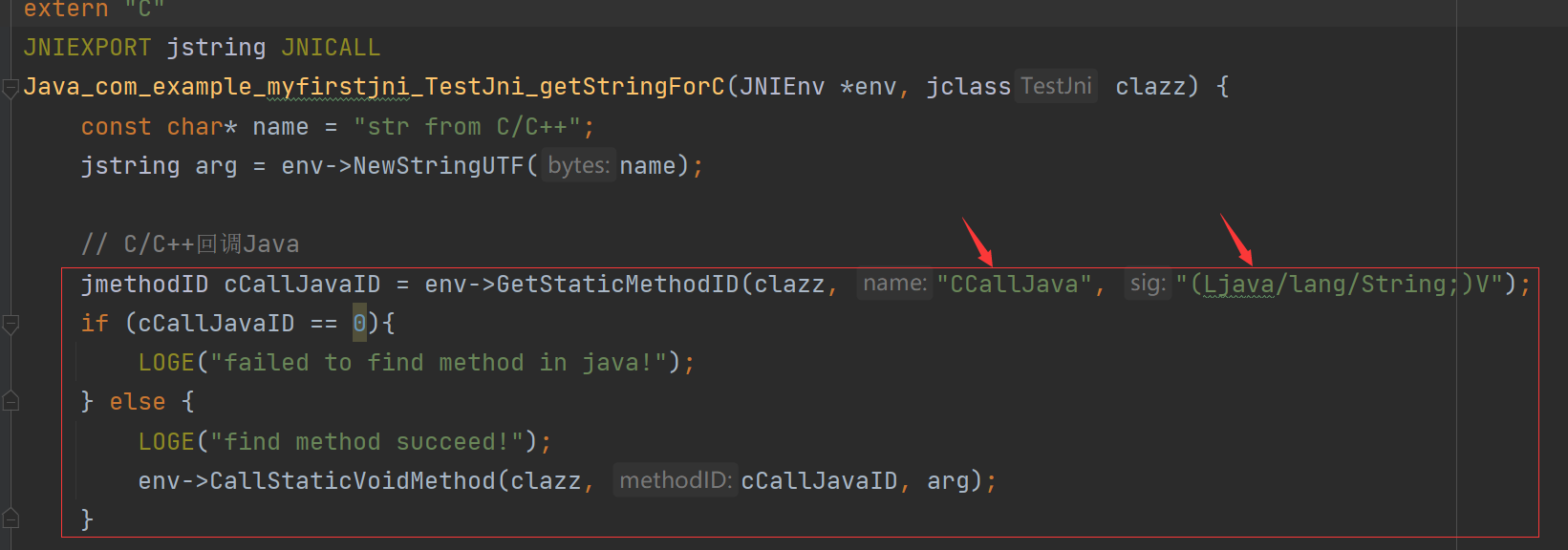


在该目录下打开命令行工具，执行javap -s TestJni，执行结果如下



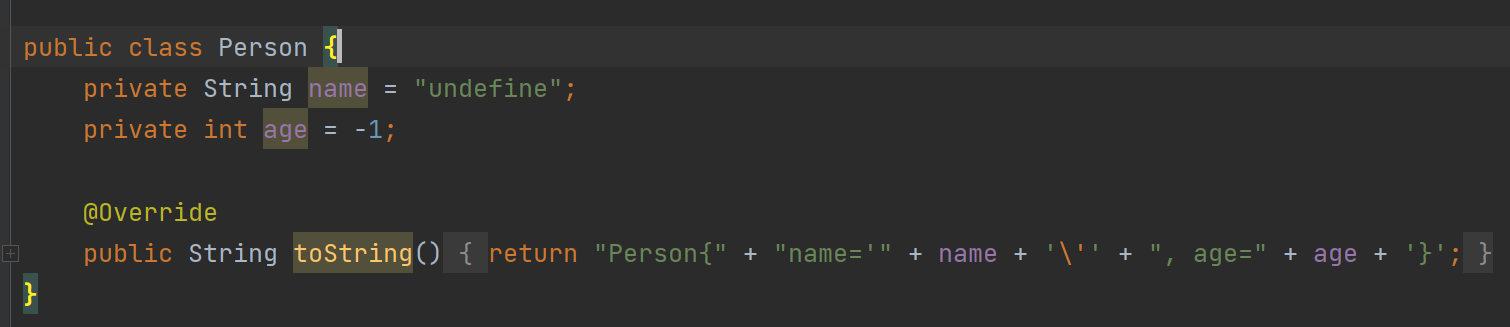
descriptor就是该方法的签名字符串

1. **C/C++代码中通过GetStaticMethodID设置方法名称和签名获取方法，之后通过CallStaticVoidMethod调用Java方法**



1. **调用其他类的方法**

新建Java类Person，如下图



C/C++通过FindClass获取jclass，再通过NewObject创建jobject对象，之后通过CallObjectMethod方法调用，C/C++中调用Java类Person方法toString就实现了，如下图

