Java利用JNA调用c/c++动态库

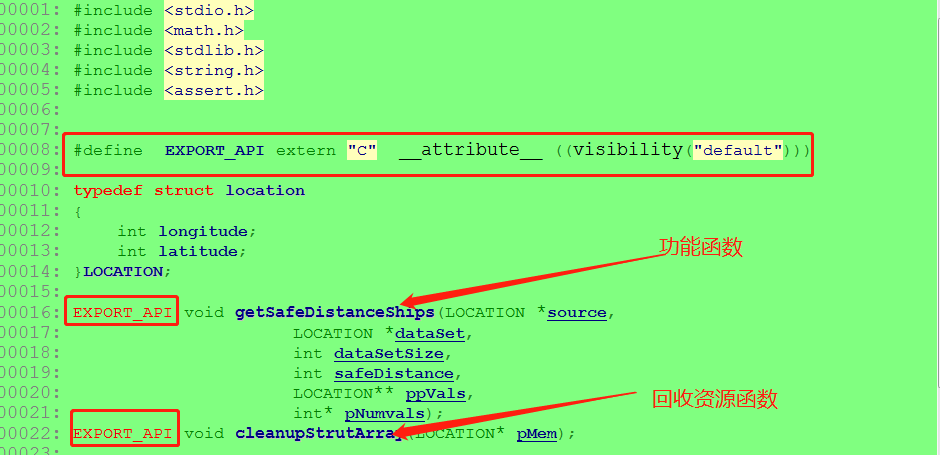
数字地球研究院

# 准备c/c++动态库

### 编写相关c/c++功能函数接口

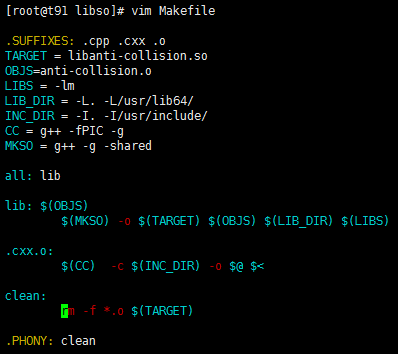
注意：

c++需要设置 extern ”C”作为导出函数前缀。另外，可以利用visibility属性控制动态库中函数、变量、类对外部类可见性。有关visibility属性详情参见：http://my.huhoo.net/archives/2010/03/post-52.html



### 编写相关c/c++功能函数接口

编译动态库，Makefile如下：

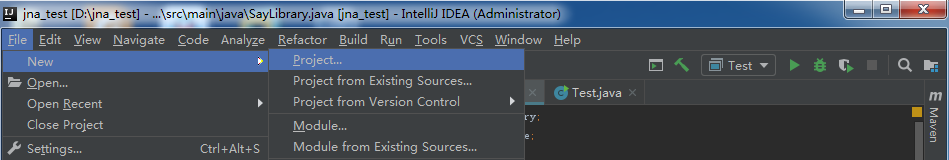


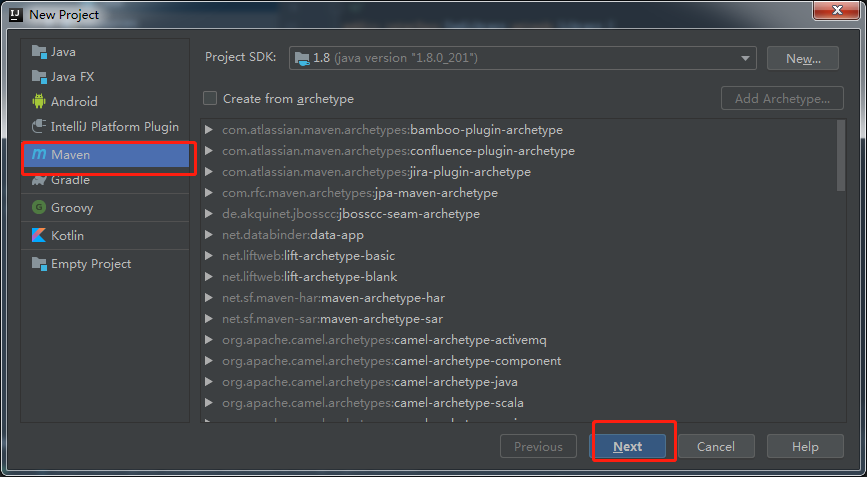
Ok，.so准备完毕。

注意：linux下需要将动态库放置在LB\_LIBRARY\_PATH环境变量所指定路径下

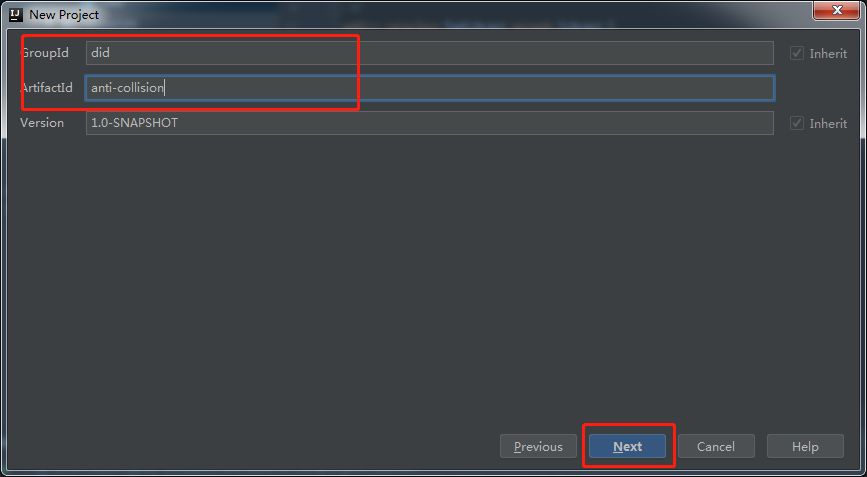
# Java使用jna调用c/c++动态库

## 用maven创建java工程

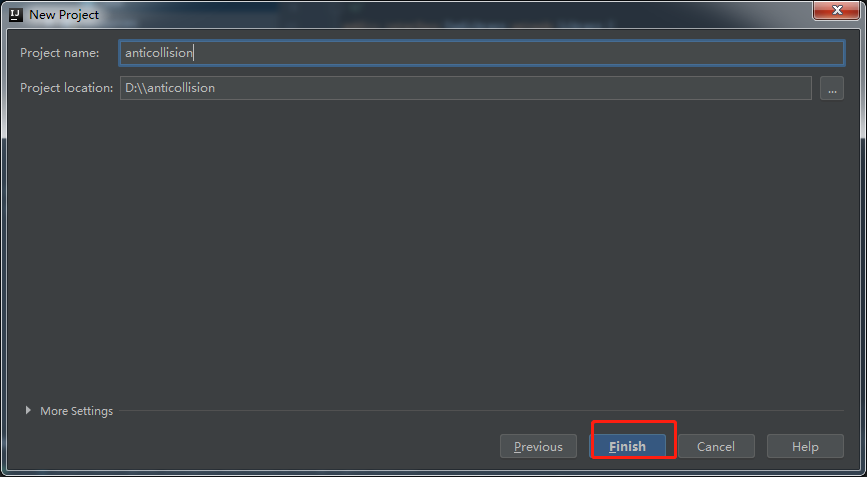




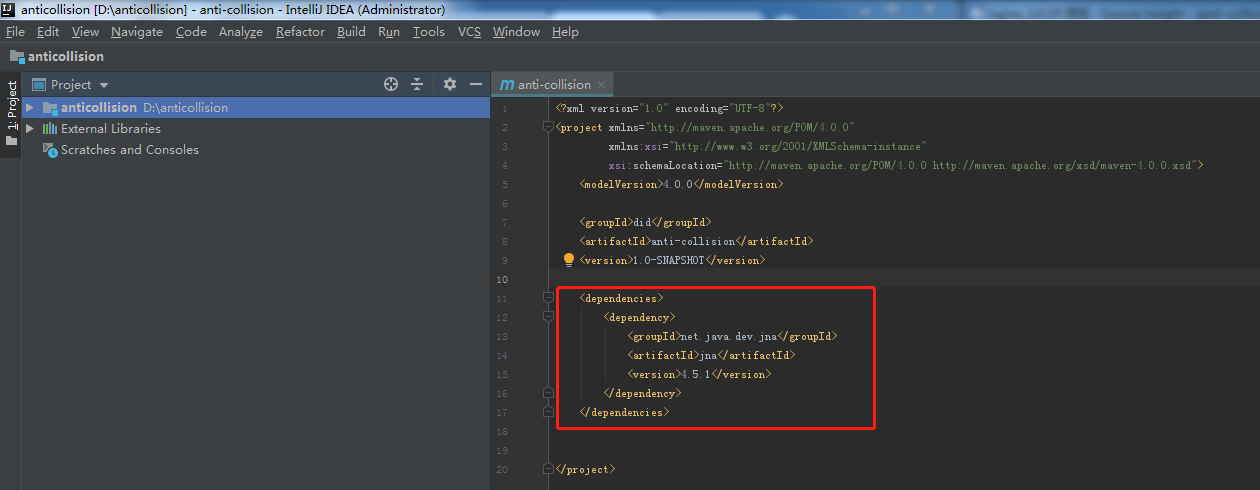
Groupid、ArtifactId任意。



工程名、位置任意。

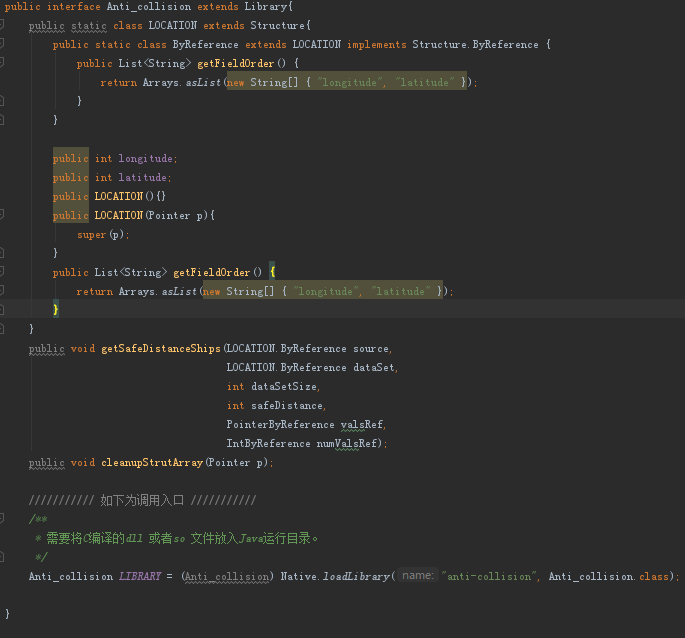


## 将jna引入pom



## 编写java代码

### 然后构建一个接口



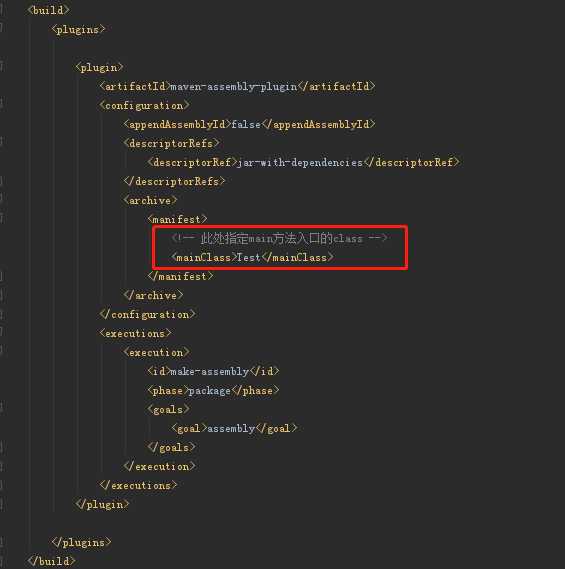
注意：继承Structure时，需要自行实现getFieldOrder，以正确的顺序返回此Structure的字段名称

注意：Native.loadLibrary() 加载动态库时，只需要动态库的名字，去掉lib和.so。例如，动态库为libHelloWorld.so,应如下Native.loadLibrary(“HelloWorld”,…)。

### 当需调用上述接口时，可以如下使用



## 将main方法入口的class加入pom



## 打jar包，上传服务器，java –jar XXX.jar

# 

爬过1、2中的注意事项（都是坑），其实真正的难点是接口中实现类型映射，构建对应实体类。特别是指针、结构体等包裹的时候，所以建议形参尽量设计简单点。

C中结构和javaJNA中类的对应关系，参考地址如下：<https://www.eshayne.com/jnaex/index.html>