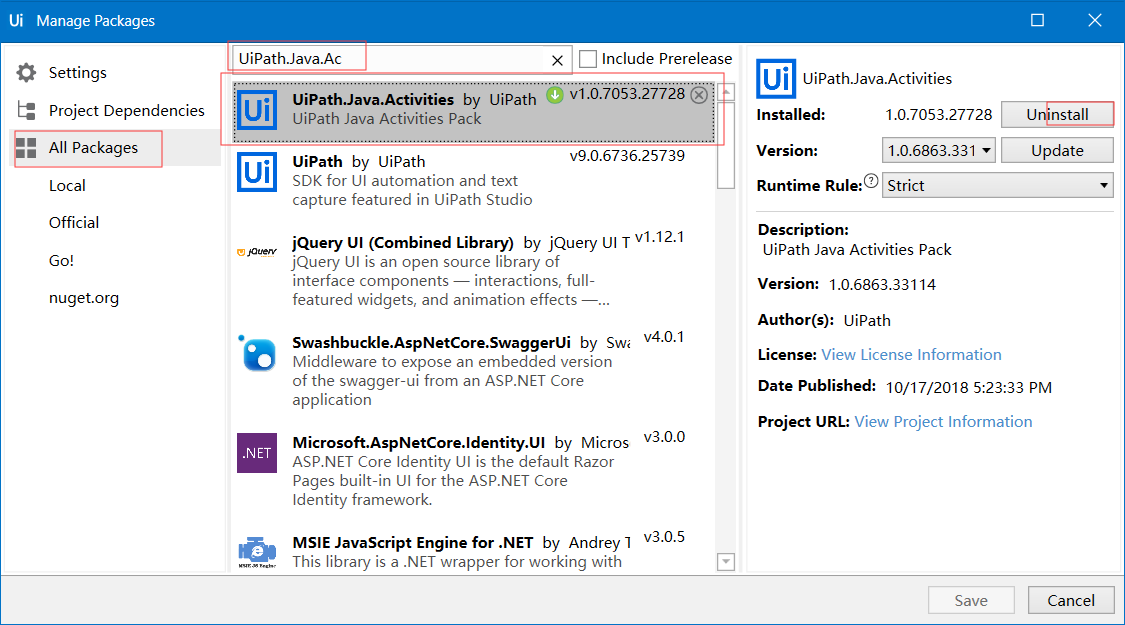
# UiPath如何调用Java

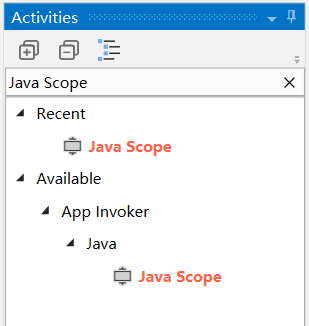
1. **调用Java方法（Invoke Java Method）的介绍**

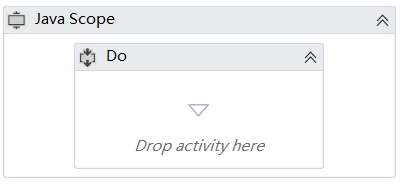
从Java Scope中的.jar加载的方法中调用指定的Java方法。并结果存储在变量中

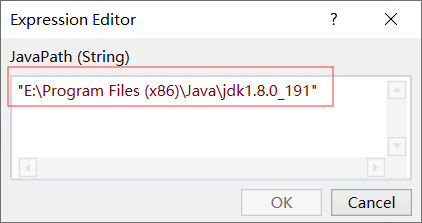
**二、Invoke Java Method 在UiPath中的使用**

1. 打开设计器, 在设计库中新建一个Sequence，为序列命名及设置Sequence存放的路径, 在Activities中搜索Java Scope(如果无法搜索到, 表示你的设计器没有安装Java活动, 在Design选项卡中, 点击”Manage Packages”, 在All Pakcages界面中搜索”Uipath.Java.”, 选择UiPath.Java.Activities, 点击右侧”install”, 之后点击”Save”保存即可)，并将其拖至设计区, 在”JavaLibraryPath”输入Java开发工具包路径(如果本机已配置Java环境变量, 可忽略此路径)

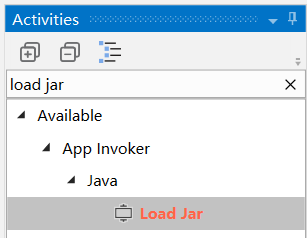


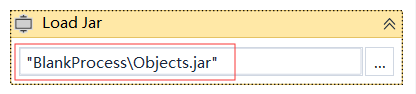




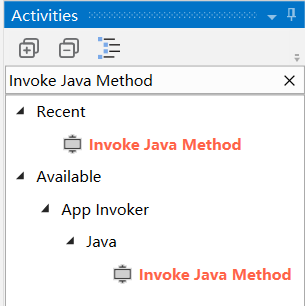


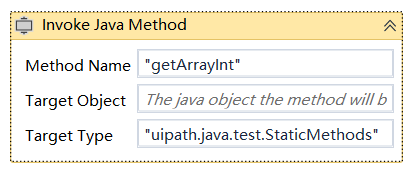
1. 在Activities中搜索Load Jar, 并将其拖至设计区. 在”JarPath”属性中, 输入Jar 路径

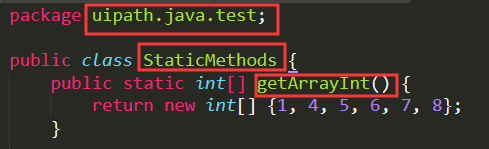


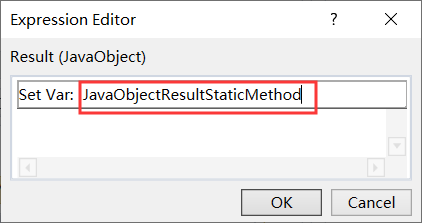


1. 在Activities中搜索Invoke Java Method, 并将其拖至设计区, 在”Method Name”属性中输入调用的Java方法名称”getArrayInt”, 在”Target Type”属性中输入Java包和类的名称的路径(以上方法名和类路径都可通过查看Java源码获取). 在Result 属性中使用Ctrl + k设置JavaObject类型的变量” JavaObjectResultStaticMethod”, 用于接收方法的返回值

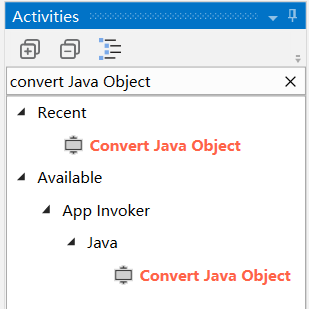


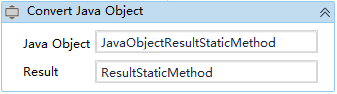




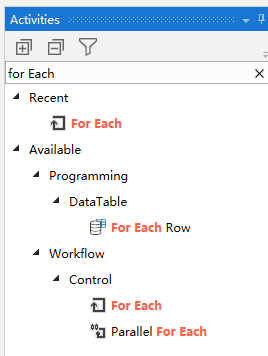


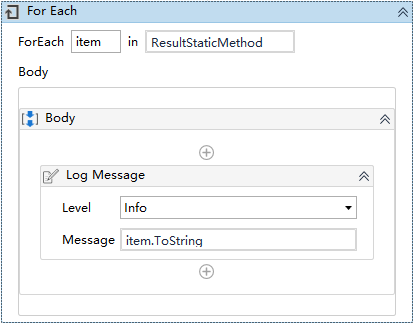
1. 在Activities中搜索Convert Java Object, 并将其拖至设计区, 在Java Object属性输入步骤3返回的Java对象, 将其赋值给” ResultStaticMethod”变量

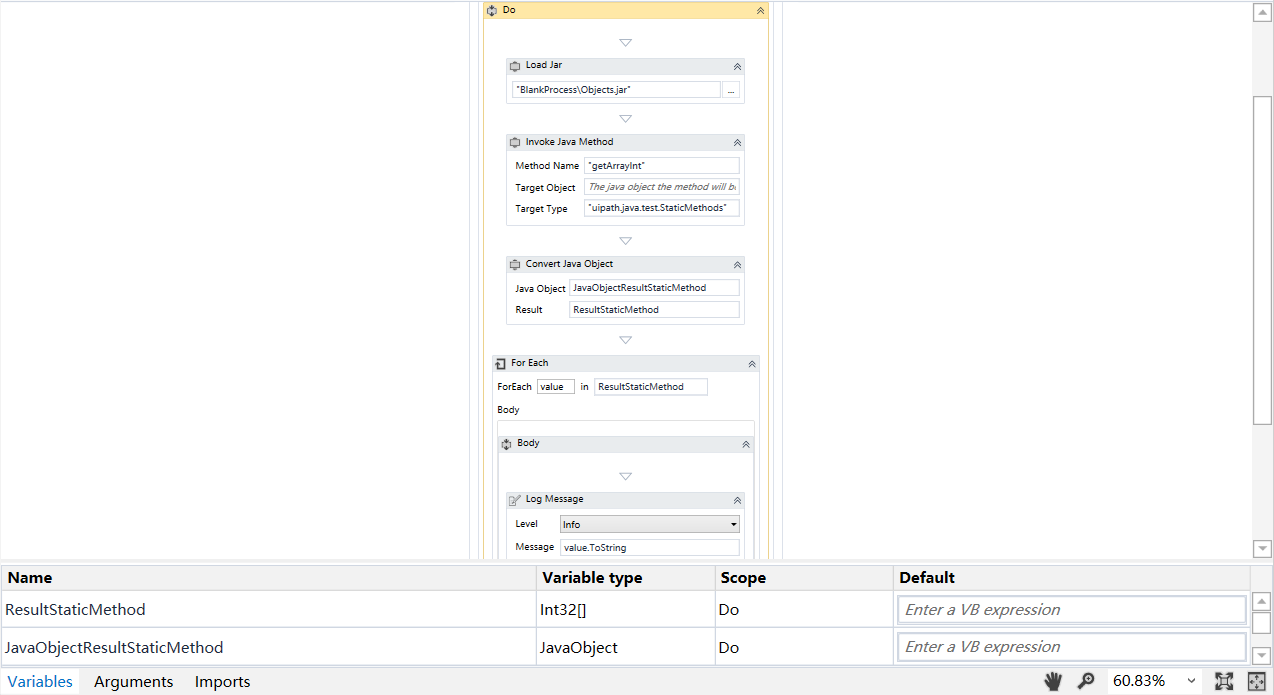




1. 在Activities中搜索For Each, 并将其拖至设计区, 在Values属性中, 输入步骤4返回的对象” ResultStaticMethod”, 在Activities中搜索Log Message, 将其拖至For Each 活动的Body 里面, 在Message 属性中输入” item.ToString”.







#运行完成，你会看到UiPath Studio成功调用了Java包里面的方法, 并在控制台输出了结果

总结：以上流程就用到了调用Java方法（Invoke Java Method）的操作，大家可以动手试试