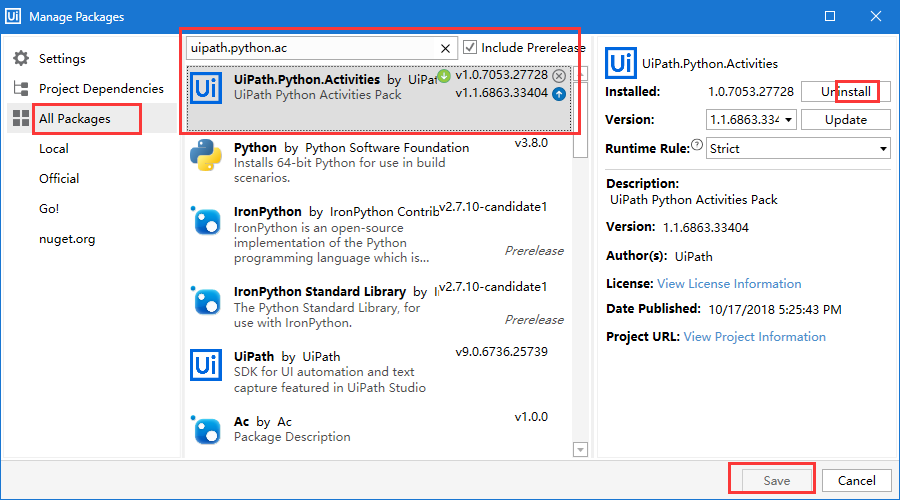
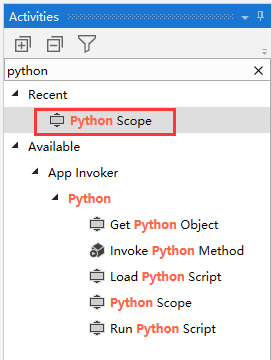
# UiPath如何调用Python

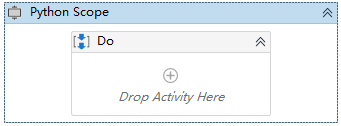
1. **调用Python（Invoke Python Method）的介绍**

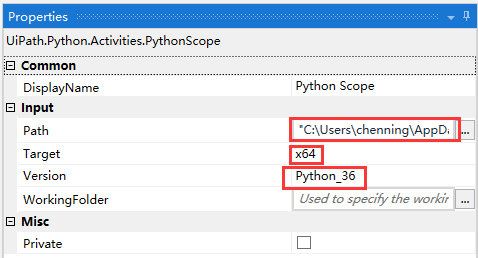
可以直接在工作流程中从Python脚本中运行指定的方法, 并结果存储在变量中

**二、Invoke Python Method在UiPath中的使用**

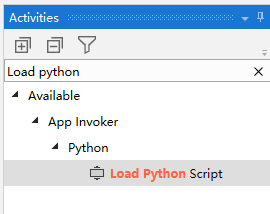
1. 打开设计器, 在设计库中新建一个Sequence，为序列命名及设置Sequence存放的路径, 在Activities中搜索Python Scope(如果无法搜索到, 表示你的设计器没有安装Python活动, 在Design选项卡中, 点击”Manage Packages”, 在All Pakcages界面中搜索”Uipath.Python.”, 选择UiPath.Python.Activities, 点击右侧”install”, 之后点击”Save”保存即可)，并将其拖至设计区, 在”Path”输入本机Python安装路径, 在Target属性中选择操作系统类型, 64位操作系统选择x64, 32位操作系统选择x86, 在Version属性中, 选择你的Python的版本 

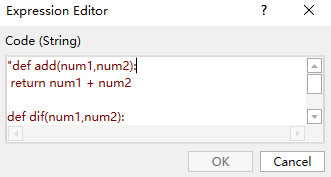


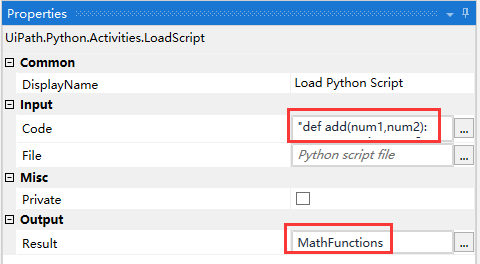




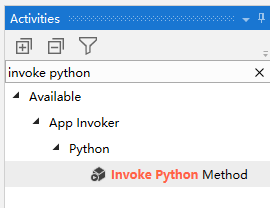
1. 在Activities中搜索Load Python Script, 并将其拖至Python Scope的设计区, 在Code 属性中, 输入你要加载的Python代码. 在Result 属性输入自定义变量名” MathFunctions ", 用于保存返回的结果

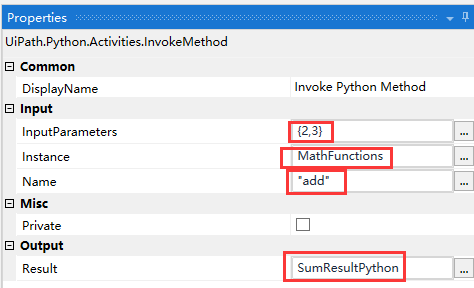




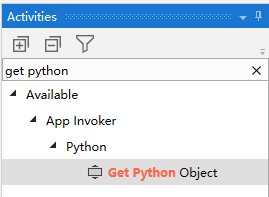


1. 在Activities中搜索Load Python Script, 并将其拖至Python Scope的设计区, 在InputParameters属性中, 输入要调用的Python脚本的输入参数列表(如果有的话), 在Instance属性中输入步骤2返回的结果变量” MathFunctions”, 在Name属性中输入要调用的方法名称”add”. 在Result 属性输入自定义变量名” SumResultPython", 用于保存返回的结果

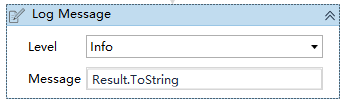


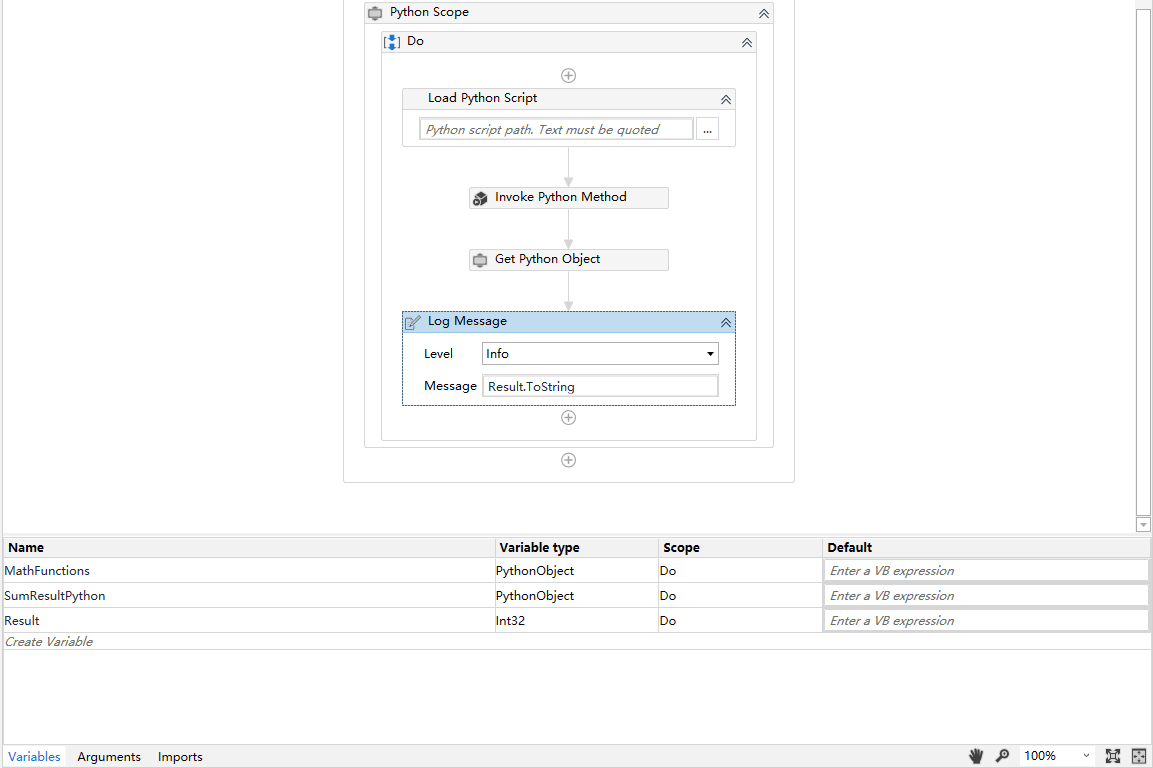


1. 在Activities中搜索get Python Object, 并将其拖至Python Scope的设计区, 在PythonObject属性中输入步骤3的返回结果” SumResultPython”, TypeArgument属性修改为Int32(根据实际情况修改), 在Result属性中输入自定义变量” Result”



1. 在Activities中搜索Log Message, 将其拖至设计区, 在Message 属性中输入”Result .ToString”





#运行完成，你会看到UiPath Studio成功调用了Python脚本, 并在控制台输出了结果

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