

Java 运行 Python 脚本的几种方法实践

Java 运行 Python 脚本，查了下相关的资料，总体上分为两大类方式：引入第三方类库和 Java 自带内库。

第一种：

引入第三方类库 `jython.jar`, 引入类库的执行情况有以下几种

1. 直接执行 `python .py` 文件

```
// 第一种方法：  
//jython引入python，java可通过它调用python脚本。缺点是jython带的库比较少，  
第三方模块无法引入  
//执行一个py文件  
PythonInterpreter interpreter = new PythonInterpreter();  
interpreter.execfile("./hello.py");
```

`hello.py`

```
print 'hello'  
number=[3,5,2,0,6]  
print number  
number.sort()  
print number  
number.append(0)  
print number  
print number.count(0)  
print number.index(5)
```

2. 直接调用 `python.py` 中的某个函数模块

```
PythonInterpreter interpreter = new PythonInterpreter();  
//得到py文件里面的函数模块 add  
interpreter.execfile("./adder.py");  
PyFunction func = interpreter.get("add",PyFunction.class);  
//传参数 a,b  
int a = 2016, b = 1;  
PyObject pyobj = func.__call__(new PyInteger(a), new  
PyInteger(b));  
System.out.println(a+ " + "+b+ " = " + pyobj.toString());
```

`adder.py`

```
def add(a, b):  
    return a + b
```

3. 直接执行 Python 脚本代码

```
PythonInterpreter interpreter = new PythonInterpreter();
    //直接执行 py 命令
    interpreter.exec("import sys\nprint sys.path\nprint
'wangyuanlong'");
```

第二种：

Runtime 法，此种方法不需要引入 jython，而是通过调用 java 自身的一个内置库得到当前系统的 python 环境变量。然后像执行 python 命令一样在 java 中调用。需要指出的是，这种方式不会将 python 的输出定向到 java 的控制台，需要将其输出流重新定义到 java 控制台才能在控制台输出。注：如果执行的 python 脚本有引用第三方包的，建议使用此种方式。使用上面两种方式会报错 `java ImportError: No module named arcpy`。

```
//第二种方法：
    //这种方法可以得到当前系统环境下的所有 python 可用的模块库，可用
try{
    Process pr = Runtime.getRuntime().exec("python test.py");
    BufferedReader in = new BufferedReader(new
        InputStreamReader(pr.getInputStream())); //将 python 脚本的输入流获取到并放
到缓冲区，然后按行读取。
    String line;
    while ((line = in.readLine()) != null) {
        System.out.println(line);
    }
    in.close();
    pr.waitFor();
} catch (Exception e){
    e.printStackTrace();
}
}
```

综述：

第一种方法比较适合调用的脚本是一些基础的不涉及第三方 python 库的脚本，但是他需要引入一个额外的第三方类库 jython，好处他的输出可直接定向到 java 的标准控制台输出，适合控制台输出较多的情况。第二种方法是一个功能比较强大的方法，他的好处很明显，如果 python 脚本本身引入了一些不常用的类库，那么这第二种方法是比较靠谱的。但这个方法不能自动将 python 的输出定向到 java 的控制台，必须手动显示的去操作。实际运用的时候大家可以根据情况使用，就个人而言，我比较偏向第二种调用方式。