

使用 Python 调用 MATLAB

本参考文档展示常见用例，但绝非详尽无遗。>> 图标提供指向 MATLAB® 文档相关部分的链接。有关一般信息，请参阅

<https://www.mathworks.com/products/matlab/matlab-and-python.html>。

设置

要求 >>

安装 MATLAB Engine API for Python

要使用 MATLAB Engine API for Python®，您需要在系统上同时安装这两者。以 Python 包形式安装 MATLAB Engine API。

使用 pip 安装

在系统提示符下：

```
$ python -m pip install
matlabengine
```

使用 setup.py 安装

从系统提示符下运行 setup.py 文件：

```
$ cd [matlabroot]/extern/
engines/python
$ python setup.py install
```

要进行故障排除，请确保拥有适当的权限（以管理员身份运行）并检查 PYTHONPATH 和系统路径。

MATLAB Engine API

使用 MATLAB Engine API 从 Python 调用 MATLAB。>>

导入模块并启动引擎：

```
>>> import matlab.engine
>>> eng = matlab.engine.
start_matlab()
```

通过引擎调用函数：

```
>>> x = eng.sqrt(42.0)

捕获多个输出：
>>> x = eng.
gcd(42.0,8.0,nargout=3)
>>> x = eng.plot(x,y,nargout=0)
```

停止引擎：

```
>>> x = eng.exit()
```

数据类型转换

数据类型将尽可能自动转换。>>

Python	MATLAB
float	double
complex	complex double
int	int64
float(nan)	NaN
float(inf)	Inf
bool	logical
str	char
dict	struct
list	cell array
set	cell array
tuple	cell array

您可以在 Python 中创建 MATLAB 数组以轻松地传递数据给 MATLAB 函数：

```
>>> x = matlab.double([1,4,9])
```

使用 MATLAB App

使用 MATLAB 引擎工作区 >>

MATLAB 引擎工作区可用于访问来自 MATLAB 和 Python 的变量，且便于 App 的使用。

将变量添加到该工作区：

```
>>> x = 4.0
>>> eng.workspace['y'] = x
```

访问 MATLAB 工作区的变量：

```
>>> x = eng.workspace['y']
```

使用 eval 函数执行语句：

```
>>> a = eng.eval('sqrt(y)')
```

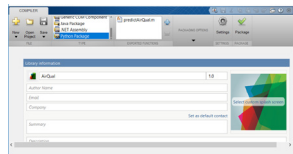
您可以通过使用调用该 App 的命令从 Python 中打开 MATLAB 中的 App：

```
>>> eng.signalAnalyzer()
>>> eng.classificationLearner()
```

创建 Python 包

MATLAB 函数打包 >>

使用库编译器为 MATLAB 函数创建 Python 包：



从 Python 包调用 MATLAB 函数

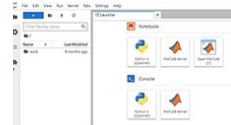
```
>>> import PackageName
>>> pkg = PackageName.
initialize()
>>> result = pkg.Foo()
```

关闭包

```
>>> pkg.terminate()
```

在其他 IDE 中使用 MATLAB

您可以在 Jupyter® 或 VSCode 中使用 MATLAB。安装和设置步骤在此处提供的链接中有详细说明。>>



安装 MATLAB Integration for Jupyter >>

例如，安装该包以从 Jupyter 执行 MATLAB。

```
$ python pip install
jupyter-matlab-proxy
$ install-matlab-kernelspec
```

现在，您可以在浏览器中使用 MATLAB，或使用 MATLAB 内核在您的记事本中运行代码。