

# Jupyter Notebook

## Jupyter Notebook概览

Jupyter Notebook 是一款在线Web应用，能让用户将说明文本、数学公式、代码以及一些可视化的内容全部组合到一个易于共享的文档中。

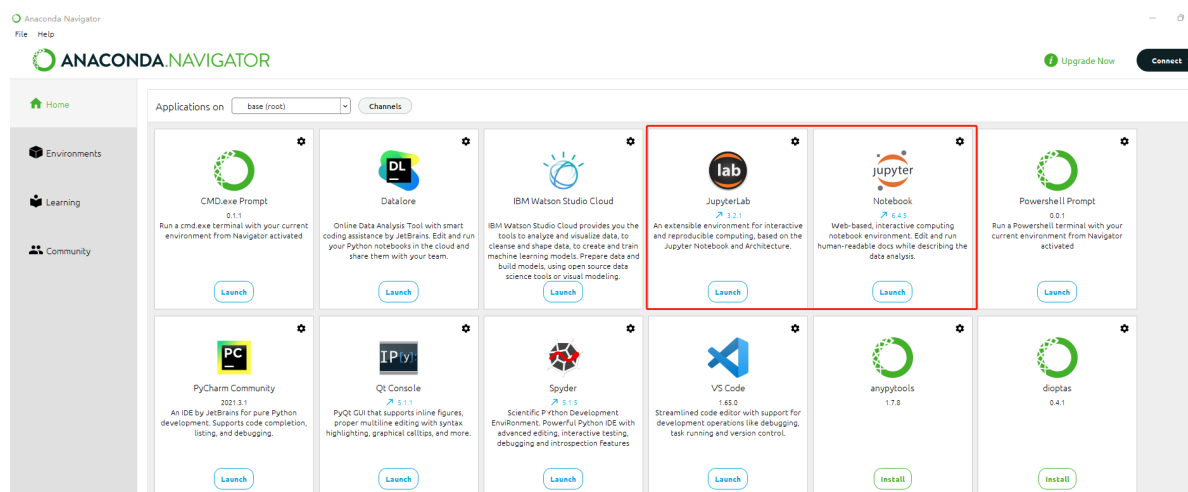
Jupyter名字的由来是它所服务的三种语言的缩写：Julia、Python、R

孙老师在上课时演示代码的主要方式就是使用的Jupyter Notebook

## 安装及打开方式

### 安装

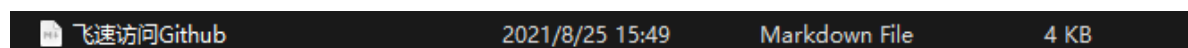
如果你看过之前的Conda教程并且已经成功安装好了Anaconda的话可以省略安装环节，因为Anaconda自带Jupyter Notebook等一系列工具。我们可以在Anaconda Navigator中看到它。



### 打开方式

在你要运行jupyter的目录下键入cmd命令以进入命令行：

可以先在任意磁盘内建立文件夹，这里我建立了一个名为Jupyter Learn的新文件夹：



点击文件路径栏，输入cmd命令以进入命令行：



在命令行窗口中输入命令 `jupyter notebook` 并按下回车，如下：

```
管理员: C:\Windows\System32\cmd.exe - jupyter notebook
Microsoft Windows [版本 10.0.22000.348]
(c) Microsoft Corporation. 保留所有权利。

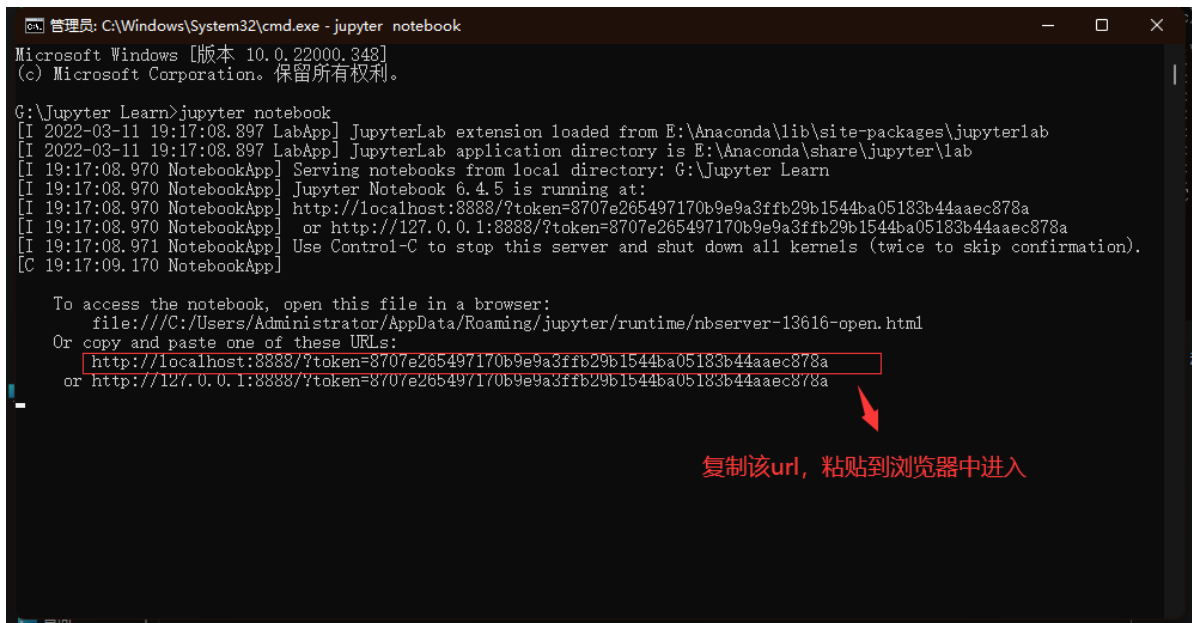
G:\Jupyter Learn>jupyter notebook
[I 2022-03-11 19:17:08.897 LabApp] JupyterLab extension loaded from E:\Anaconda\lib\site-packages\jupyterlab
[I 2022-03-11 19:17:08.897 LabApp] JupyterLab application directory is E:\Anaconda\share\jupyter\lab
[I 19:17:08.970 NotebookApp] Serving notebooks from local directory: G:\Jupyter Learn
[I 19:17:08.970 NotebookApp] Jupyter Notebook 6.4.5 is running at:
[I 19:17:08.970 NotebookApp] http://localhost:8888/?token=8707e265497170b9e9a3ffb29b1544ba05183b44aaec878a
[I 19:17:08.970 NotebookApp] or http://127.0.0.1:8888/?token=8707e265497170b9e9a3ffb29b1544ba05183b44aaec878a
[I 19:17:08.971 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[C 19:17:09.170 NotebookApp]

To access the notebook, open this file in a browser:
    file:///C:/Users/Administrator/AppData/Roaming/jupyter/runtime/nbserver-13616-open.html
Or copy and paste one of these URLs:
    http://localhost:8888/?token=8707e265497170b9e9a3ffb29b1544ba05183b44aaec878a
    or http://127.0.0.1:8888/?token=8707e265497170b9e9a3ffb29b1544ba05183b44aaec878a
```

稍等片刻后，系统会自动打开浏览器并进入以下页面：



注意：如果没有自动打开浏览器进入主页面，你可以复制命令行中的网址，自行进入



注意：在你使用Jupyter Notebook的过程中，你不能关闭上述的命令行窗口，否则Jupyter服务会终止,并且你所写的文件可能会丢失！

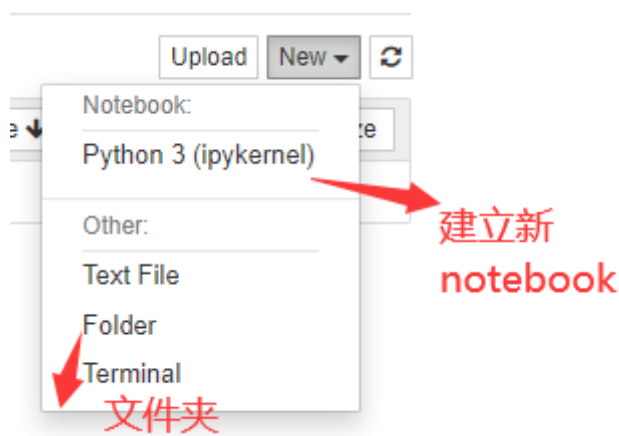
## Jupyter的简单使用

Jupyter的主页面如下：

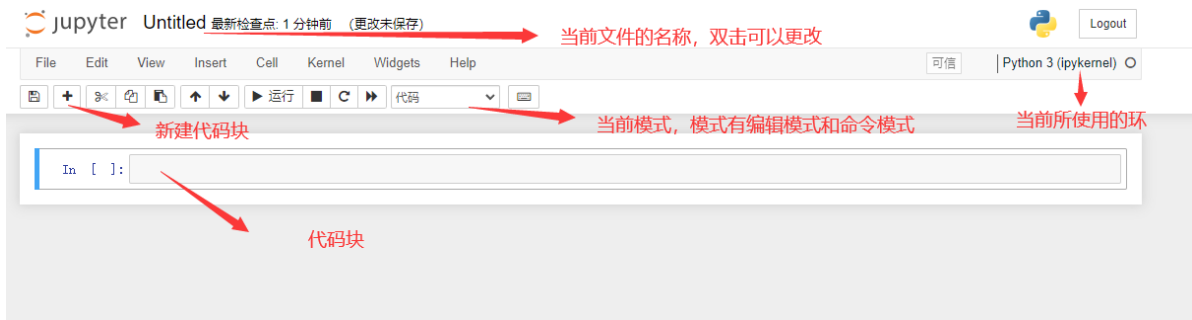


## 建立新Notebook并运行代码

我们来新建一个notebook文件，点击New按钮，选择python：



进入以下页面：



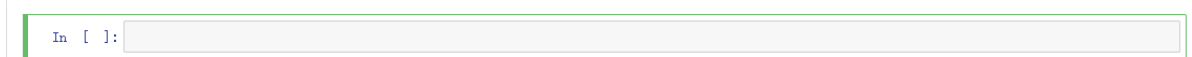
双击即可重命名：

## Cells

代码块又称Cells，通常有两种主要的Cell，分别是Code cell，Markdown cell

- Code cell：包括需要执行的代码，以及其运行结果
- Markdown cell：包含的是Markdown格式的文本及其执行结果

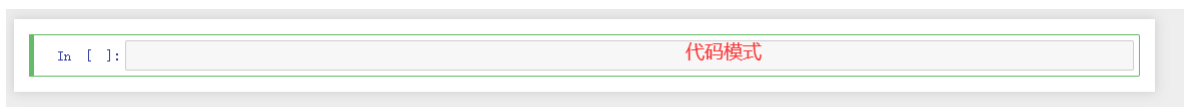
我们可以发现代码块前面的颜色为蓝色，蓝色为命令模式，我们可以使用一些快捷键来进行一些切换操作，另一种为绿色的编辑模式：



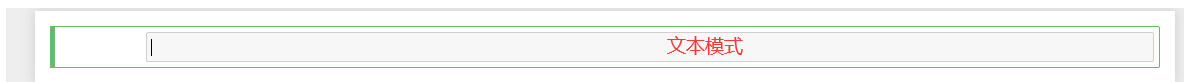
在编辑模式下，我们可以按下 ESC 键以进入命令模式

命令模式下，我们可以按下 ENTER 键以进入编辑模式

前面如果有 `In [ ]:` 字样的为 Code cell，没有的话为 Markdown Cell



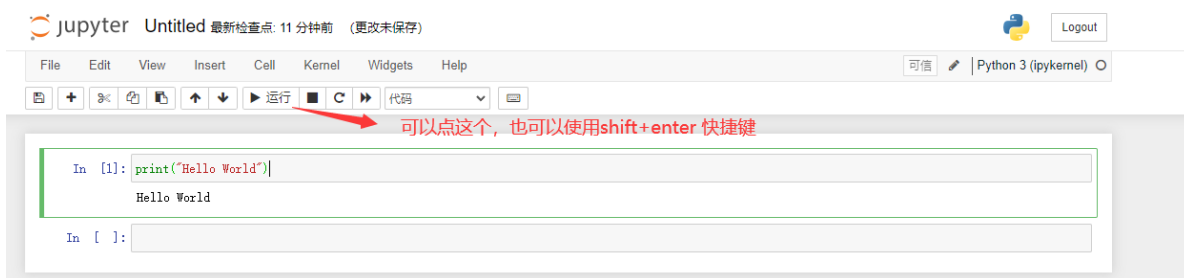
如果要切换模式的话，先按ESC进入命令模式，然后在代码模式下按 M 键以进入文本模式，文本模式下按 Y 键以进入代码模式。



## 运行代码

我们在代码模式下敲入一行代码，要运行的话，可以单击运行键，也可以同时按下 SHIFT+ENTER,如图：

- ctrl + enter 运行当前cell
- shift + enter 运行当前cell 并自动添加下一块cell



可以注意到 code cell 的左侧会有一个标签 `In [1]`，这里的数字表示该代码块运行的次序，即在该 notebook 中，如果没有执行，显示的是 `In [ ]`，如果是该 notebook 第一个运行的代码块，则是 `In [1]`，如果再次运行，则显示 `In [2]`，依次类推，同个代码块多次执行，这个数字也是会改变的。而 `In` 是 `Input` 的缩写。如果代码块运行时间有点久，它会显示 `In [*]`，表示当前代码块正在运行中。

可以使用上下键来进行切换代码块

## 快捷键

notebook 有很多快捷键，可以通过菜单中的 `Help->Keyboard Shortcuts` 查看，也可以直接用快捷键 `Ctrl+Shift+P` 查看。下面简单介绍一些快捷键：

- 编辑模式和命令模式可以通过 `Esc` 和 `Enter` 进行转换，一般是按 `Esc` 进入命令模式，`Enter` 进入编辑模式

在命令模式下：

- 在 cell 之间上下浏览采用上下箭头，或者 `Up` 和 `Down` 键
- `A` 表示在当前 cell 上方插入一个新的 cell，而 `B` 则是下方插入新的 cell
- `M` 表示变为 `Markdown cell`，而 `Y` 是表示变为 `code cell`
- 连续按两次 `D` 是删除当前 cell
- `Z` 是撤销操作
- `Shift` 加上 `Up` 或者 `Down` 可以一次选择多个 cells，接着采用 `Shift + M` 可以合并多个 cells

Jupyter 笔记本有两种不同的键盘输入模式。编辑模式允许您将代码或文本输入到一个单元格中，并通过一个绿色边框的单元格来表示命令模式将键盘与笔记本级命令绑定在一起，并通过一个灰框、左边距蓝色的单元格显示。

## 命令行模式 (按 `Esc` 生效)

编辑快捷键

`F`: 查找并且替换

`Ctrl-Shift-F`: 打开命令配置

`Ctrl-Shift-P`: 打开命令配置

`Enter`: 进入编辑模式

`P`: 打开命令配置

`Shift-Enter`: 运行代码块, 选择下面的代码块

`Ctrl-Enter`: 运行选中的代码块

`Alt-Enter`: 运行代码块并且在下面插入代码块

`Y`: 把代码块变成代码

`M`: 把代码块变成 Markdown

`R`: 清除代码块格式

`1`: 把代码块变成标题 1

`2`: 把代码块变成标题 2

`3`: 把代码块变成标题 3

`4`: 把代码块变成标题 4

`5`: 把代码块变成标题 5

`6`: 把代码块变成标题 6

`K`: 选择上面的代码块

`↑`: 选择上面的代码块

`↓`: 选择下面的代码块

`J`: 选择下面的代码块

`Shift-K`: 扩展上面选择的代码块

`Shift-↑`: 扩展上面选择的代码块

`Shift-J`: 扩展下面选择的代码块

`Ctrl-A`: select all cells

`A`: 在上面插入代码块

`B`: 在下面插入代码块

`X`: 剪切选择的代码块

`C`: 复制选择的代码块

`Shift-V`: 粘贴到上面

`V`: 粘贴到下面

`Z`: 撤销删除

`D`, `D`: 删除选中单元格

`Shift-M`: 合并选中单元格, 如果只有一个单元格被选中

`Ctrl-S`: 保存并建立检查点

`S`: 保存并建立检查点

`L`: 切换行号

`O`: 切换选定单元格的输出

`Shift-O`: 切换选中单元格的输出滚动

`H`: 显示键盘快捷键

`I`, `I`: 中断内核

`O`, `O`: 重启内核 (带确认对话框)

`Ctrl-V`: 从系统剪贴板粘贴

`Esc`: 关闭分页器

`Q`: 关闭分页器

`Shift-L`: 在所有单元格中切换行号, 并保

## 编辑模式 (按 `Enter` 生效)

<code>Tab</code> : 代码补全或缩进	<code>Ctrl-右</code> : 往右跳一个单词
<code>Shift-Tab</code> : 工具提示	<code>Ctrl-退格</code> : 删除前面的单词
<code>Ctrl-]</code> : 缩进	<code>Ctrl-Delete</code> : 删除后面的单词
<code>Ctrl-[</code> : 取消缩进	<code>Ctrl-Y</code> : 重做
<code>Ctrl-A</code> : 全选	<code>Alt-U</code> : 重新选择
<code>Ctrl-Z</code> : 撤销	<code>Ctrl-M</code> : 进入命令行模式
<code>Ctrl-/</code> : 注释	<code>Ctrl-Shift-F</code> : 打开命令配置
<code>Ctrl-D</code> : 删除整行	<code>Ctrl-Shift-P</code> : 打开命令配置
<code>Ctrl-U</code> : 撤销选择	<code>Esc</code> : 进入命令行模式
<code>Insert</code> : 切换重写标志	<code>Shift-Enter</code> : 运行代码块, 选择下面的代码块
<code>Ctrl-Home</code> : 跳到单元格起始处	<code>Ctrl-Enter</code> : 运行选中的代码块
<code>Ctrl-上</code> : 跳到单元格起始处	<code>Alt-Enter</code> : 运行代码块并且在下面插入代码块
<code>Ctrl-End</code> : 跳到单元格最后	<code>Ctrl-Shift-Minus</code> : split cell at cursor(s)
<code>Ctrl-下</code> : 跳到单元格最后	<code>Ctrl-S</code> : 保存并建立检查点
<code>Ctrl-左</code> : 往左跳一个单词	<code>下</code> : 光标下移
	<code>上</code> : 光标上移

## 更换Jupyter中的Conda虚拟环境

---