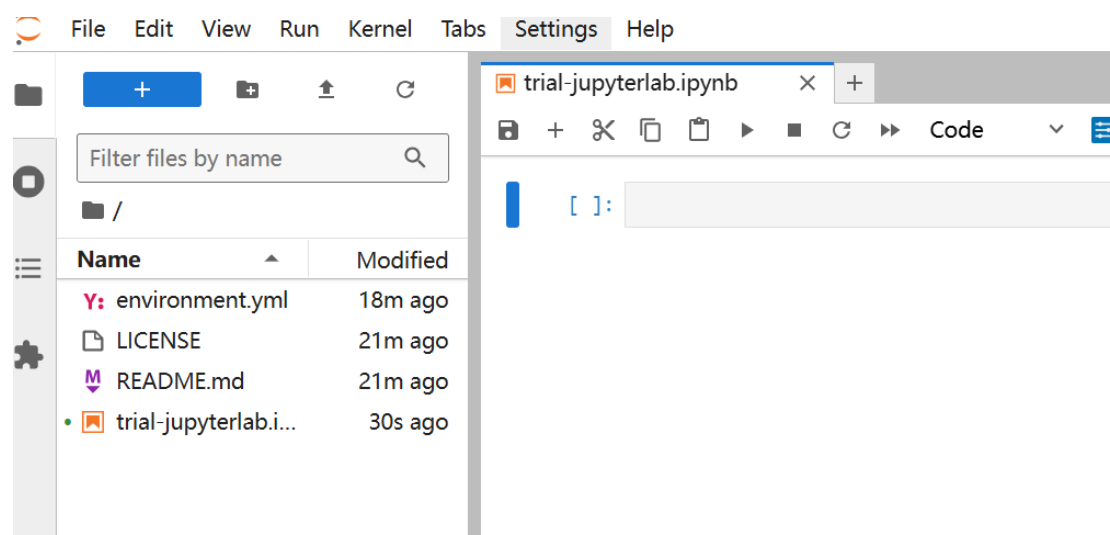


第七周学习报告

1. Fork 第 07 周打卡 仓库至你的名下，然后将你名下的这个仓库 Clone 到你的本地计算机。
2. 用 VS Code 打开项目目录，新建一个 `environment.yml` 文件，指定安装 Python 3.12 和 `jupyterlab`，然后运行 `conda env create` 命令创建 Conda 环境。
3. 在项目目录下，运行 `jupyter lab` 命令，启动后端 (Backend) 服务，在浏览器里粘贴地址访问 前端 (Frontend) 页面

```
74567@DESKTOP-N5C0CLE MINGW64 ~/repo/week07 (main)
$ jupyter lab
[I 2025-04-23 10:10:37.993 ServerApp] jupyter_lsp | extension was successfully linked.
[I 2025-04-23 10:10:38.004 ServerApp] jupyter_server_terminals | extension was successfully linked.
[I 2025-04-23 10:10:38.018 ServerApp] jupyterlab | extension was successfully linked.
[I 2025-04-23 10:10:38.030 ServerApp] notebook | extension was successfully linked.
[I 2025-04-23 10:10:38.042 ServerApp] Writing Jupyter server cookie secret to C:\Users\74567\AppData\Roaming\jupyter\run
time\jupyter_cookie_secret
[I 2025-04-23 10:10:39.252 ServerApp] notebook_shim | extension was successfully linked.
[I 2025-04-23 10:10:39.252 ServerApp] panel.io.jupyter_server_extension | extension was successfully linked.
[I 2025-04-23 10:10:39.376 ServerApp] notebook_shim | extension was successfully loaded.
[I 2025-04-23 10:10:39.384 ServerApp] jupyter_lsp | extension was successfully loaded.
[I 2025-04-23 10:10:39.386 ServerApp] jupyter_server_terminals | extension was successfully loaded.
[I 2025-04-23 10:10:39.399 LabApp] JupyterLab extension loaded from D:\Anaconda\Lib\site-packages\jupyterlab
[I 2025-04-23 10:10:39.399 LabApp] JupyterLab application directory is D:\Anaconda\share\jupyterlab
[I 2025-04-23 10:10:39.400 LabApp] Extension Manager is 'pypi'.
[I 2025-04-23 10:10:39.842 ServerApp] jupyterlab | extension was successfully loaded.
[I 2025-04-23 10:10:39.852 ServerApp] notebook | extension was successfully loaded.
[I 2025-04-23 10:10:39.854 ServerApp] panel.io.jupyter_server_extension | extension was successfully loaded.
[I 2025-04-23 10:10:39.856 ServerApp] Serving notebooks from local directory: C:\Users\74567\repo\week07
[I 2025-04-23 10:10:39.856 ServerApp] Jupyter Server 2.14.1 is running at:
[I 2025-04-23 10:10:39.857 ServerApp] http://localhost:8888/Lab?token=28c9e5440b74d368d3d02573f85ec507908a92c8ed57675b
[I 2025-04-23 10:10:39.857 ServerApp] http://127.0.0.1:8888/Lab?token=28c9e5440b74d368d3d02573f85ec507908a92c8ed5767
5b
[I 2025-04-23 10:10:39.857 ServerApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirm
ation).
[C 2025-04-23 10:10:39.967 ServerApp]
```



4. 在 JupyterLab 页面里，新建一个 Notebook，改名为 trial-jupyterlab.ipynb，在里面实践掌握以下功能：

(1) 在单元格 (Cell) 里编写 Python 代码，按 Shift+Enter 运行 Cell 并下移

(2) 在单元格 (Cell) 上按 ESC 切换到 命令模式 (command mode)，按 Enter 切换到 编写模式 (edit mode)

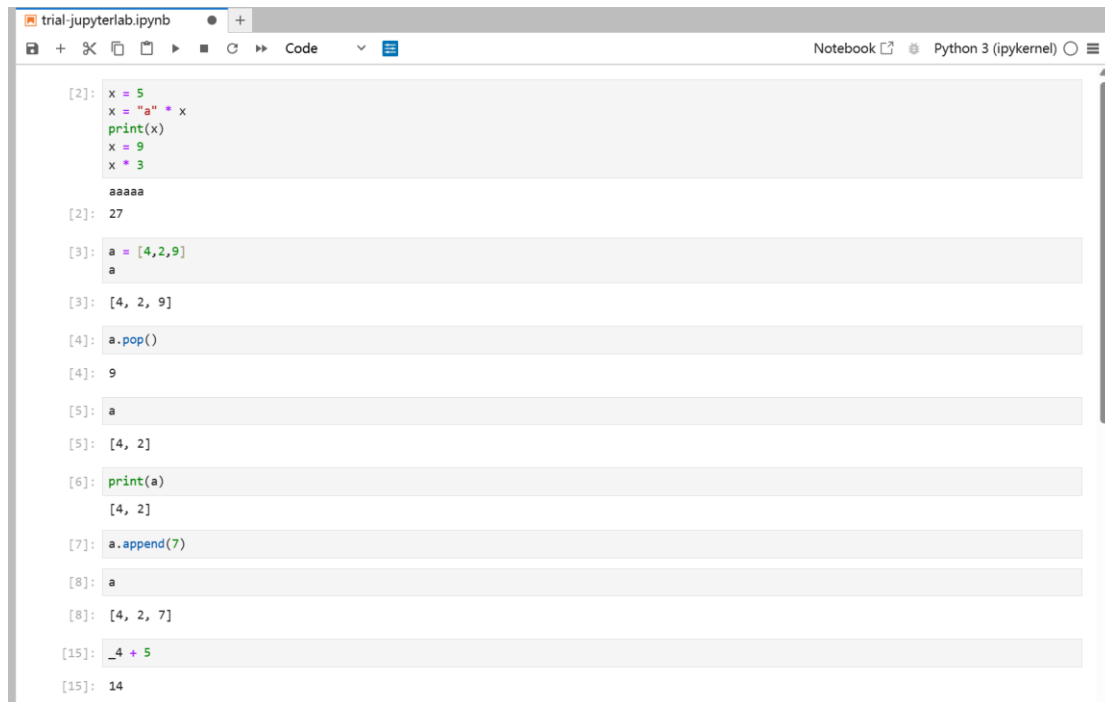
(3) 在单元格 (Cell) 的命令模式下，按 j 选择下一个，按 k 选择上一个，按 a 在上方添加，按 b 在下方添加，按 dd 删除，按住 Shift 多选，按 x 剪切，按 c 复制，按 v 粘贴，按 Shift+M 合并，按 z 撤销，按 Shift+Z 重做，按 Shift+L 显示/隐藏代码行号

(4) 在单元格 (Cell) 的编写模式下，按 Ctrl+Shift+- 切分单元格

(5) 按按钮显示/隐藏 Minimap

(6) 运行单元格 (Cell) 注意序号单调递增

(7) 单元格最后一行如果是表达式 (expression) 且运行后返回的对象不是 None，则计输出 (Out)，否则只计输入 (In)，序号为 i 的输出，可以用 `_i` 变量来引用



The screenshot shows a Jupyter Notebook window titled 'trial-jupyterlab.ipynb'. The interface includes a top bar with icons for file operations and a 'Code' dropdown menu. The notebook contains several code cells with the following content and outputs:

```
[2]: x = 5
     x = "a" * x
     print(x)
     x = 9
     x * 3
     aaaaa
[2]: 27

[3]: a = [4,2,9]
     a
[3]: [4, 2, 9]

[4]: a.pop()
[4]: 9

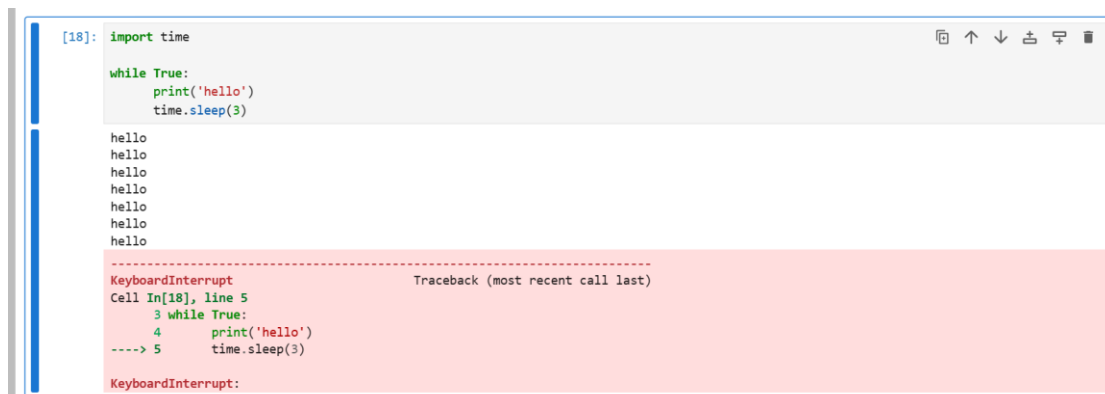
[5]: a
[5]: [4, 2]

[6]: print(a)
[6]: [4, 2]

[7]: a.append(7)
[8]: a
[8]: [4, 2, 7]

[15]: _4 + 5
[15]: 14
```

(8) 单元格 (Cell) 序号为 * 表示代码运行中，尚未返回，
按 ii 可以打断 (KeyboardInterrupt) (类似于终端的 Ctrl+C)



The screenshot shows a Jupyter Notebook cell with the following code and output:

```
[18]: import time

     while True:
         print('hello')
         time.sleep(3)

hello
hello
hello
hello
hello
hello
hello
hello
hello

-----
KeyboardInterrupt                                Traceback (most recent call last)
Cell In[18], line 5
      3 while True:
      4     print('hello')
----> 5     time.sleep(3)

KeyboardInterrupt:
```

- (9) 在单元格 (Cell) 的命令模式下，按 00 重启后端 Python 解释器 (被 Jupyter 称为 Kernel)，重启后需要从上至下重新运行一遍代码 (Shift+Enter)，运行前建议先在菜单里选择 “Edit / Clear Outputs of All Cells” 清空全部页面显示的输出
- (10) 在单元格 (Cell) 的命令模式下，按 m 切换至 Markdown 模式，按 y 切换至 Python 模式
- (11) 用豆包 (或 DeepSeek 等任何大模型) 生成一段示例

Markdown 代码，复制粘贴进 Markdown 单元格，运行以呈现 (Render)

```
# 这是一级标题

## 这是二级标题

### 这是三级标题

这是一段普通的文本。你可以使用 粗体 或者 斜体 来突出显示某些文字。

### 列表
#### 无序列表
- 项目 1
- 项目 2
  - 子项目 1
  - 子项目 2

#### 有序列表
1. 第一项
2. 第二项
3. 第三项

### 链接
你可以通过 \[这个链接\](https://www.example.com) 访问示例网站。

### 图片
([示例图片](https://picsum.photos/200/300))


### 代码块


```
python
def greet(name):
 return f"Hello, {name}!"

print(greet("World"))
```


```

```
## 一、宇宙的浩瀚
宇宙，是一个广袤无垠的空间，充满了无数的恒星、行星、星系和神秘的天体。它的尺度之大，超出了人类的想象。

### 1.1 星系的壮丽
星系是宇宙中由恒星、气体、尘埃和暗物质组成的巨大集合。其中，螺旋星系如银河系，拥有美丽的旋臂结构，宛如宇宙中的巨大风车。
([螺旋星系](https://upload.wikimedia.org/wikipedia/commons/thumb/6/6c/Milky_Way_Arms.jpg/440px-Milky_Way_Arms.jpg))

### 1.2 恒星的生命周期
恒星是宇宙中的能量源泉，它们经历着诞生、成长、衰老和死亡的过程。像太阳这样的恒星，处于主序星阶段，稳定地发光发热。



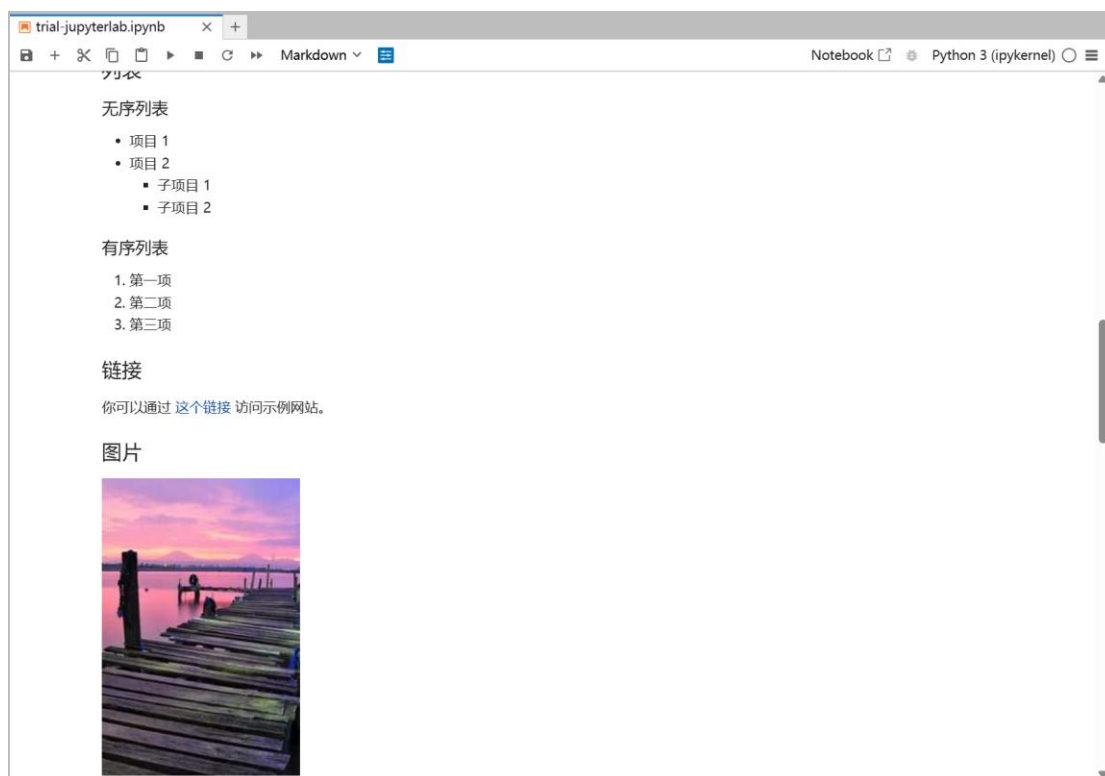
```
python
简单示意恒星质量与寿命的关系
这里只是简单示例，实际关系更复杂
def star_lifetime(mass):
 # 假设寿命与质量的 -2.5 次方成正比
 return 1 / (mass ** 2.5)

sun_mass = 1
print(f"太阳质量 (以太阳质量为单位): {sun_mass}")
print(f"相对寿命: {star_lifetime(sun_mass):.2f}")
简单模拟从地球到月球的距离和飞行时间
假设平均速度 (单位: 千米/小时)
average_speed = 3800
地球到月球的平均距离 (单位: 千米)
distance = 384400
flight_time = distance / average_speed
print(f"从地球到月球, 以平均速度 {average_speed} 千米/小时飞行, 大约需要 {flight_time:.2f} 小时")
简单示意引力波频率范围
import numpy as np
import matplotlib.pyplot as plt

引力波频率范围 (单位: 赫兹)
min_frequency = 1e-4
max_frequency = 1e4
frequencies = np.logspace(np.log10(min_frequency), np.log10(max_frequency), 100)

plt.figure(figsize=(8, 6))
plt.plot(frequencies, np.ones_like(frequencies))
```


```



(12) 用豆包（或 DeepSeek 等任何大模型）生成一段示例 HTML 代码，复制粘贴进 Markdown 单元格，运行以呈现（Render）；注意不支持 CSS

```
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <script src="https://cdn.tailwindcss.com"></script>
  <link href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.7.2/css/all.min.css" rel="stylesheet">
  <title>复杂表格</title>
</head>
<body>
  <table>
    <tr>
      <th>
        姓名
      </th>
      <th>
        年龄
      </th>
      <th>
        性别
      </th>
      <th>
        职业
      </th>
      <th>
        爱好
      </th>
      <th>
        备注
      </th>
    </tr>
    <tr>
      <td>
        张三
      </td>
      <td>
        25
      </td>
      <td>
        男
      </td>
      <td>
        程序员
      </td>
      <td>
        编程、阅读
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        李四
      </td>
      <td>
        30
      </td>
      <td>
        女
      </td>
      <td>
        教师
      </td>
      <td>
        教学、运动
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        王五
      </td>
      <td>
        35
      </td>
      <td>
        男
      </td>
      <td>
        医生
      </td>
      <td>
        医疗、旅行
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        赵六
      </td>
      <td>
        40
      </td>
      <td>
        女
      </td>
      <td>
        工程师
      </td>
      <td>
        设计、音乐
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        孙七
      </td>
      <td>
        45
      </td>
      <td>
        男
      </td>
      <td>
        企业家
      </td>
      <td>
        商业、高尔夫
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        周八
      </td>
      <td>
        50
      </td>
      <td>
        女
      </td>
      <td>
        作家
      </td>
      <td>
        写作、园艺
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        吴九
      </td>
      <td>
        55
      </td>
      <td>
        男
      </td>
      <td>
        科学家
      </td>
      <td>
        研究、登山
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        郑十
      </td>
      <td>
        60
      </td>
      <td>
        女
      </td>
      <td>
        艺术家
      </td>
      <td>
        绘画、舞蹈
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        冯十一
      </td>
      <td>
        65
      </td>
      <td>
        男
      </td>
      <td>
        律师
      </td>
      <td>
        法律、钓鱼
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        陈十二
      </td>
      <td>
        70
      </td>
      <td>
        女
      </td>
      <td>
        心理咨询师
      </td>
      <td>
        咨询、瑜伽
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        林十三
      </td>
      <td>
        75
      </td>
      <td>
        男
      </td>
      <td>
        历史学家
      </td>
      <td>
        历史、考古
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        周十四
      </td>
      <td>
        80
      </td>
      <td>
        女
      </td>
      <td>
        哲学家
      </td>
      <td>
        哲学、冥想
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        吴十五
      </td>
      <td>
        85
      </td>
      <td>
        男
      </td>
      <td>
        生物学家
      </td>
      <td>
        生物、徒步
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        郑十六
      </td>
      <td>
        90
      </td>
      <td>
        女
      </td>
      <td>
        天文学家
      </td>
      <td>
        天文、观星
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        孙十七
      </td>
      <td>
        95
      </td>
      <td>
        男
      </td>
      <td>
        物理学家
      </td>
      <td>
        物理、实验
      </td>
      <td>
        活跃
      </td>
    </tr>
    <tr>
      <td>
        周十八
      </td>
      <td>
        100
      </td>
      <td>
        女
      </td>
      <td>
        化学家
      </td>
      <td>
        化学、研究
      </td>
      <td>
        活跃
      </td>
    </tr>
  </table>
</body>
</html>
```

复杂表格示例

姓名	年龄	城市	职业
张三	25	北京	程序员
李四	30	上海	设计师
王五	22	广州	教师
赵六	35	深圳	医生
孙七	28	成都	律师
周八	26	杭州	记者
吴九	32	南京	会计
郑十	24	武汉	工程师

(13) 用豆包（或 DeepSeek 等任何大模型）生成一段示例 LaTeX 数学公式代码，复制粘贴进 Markdown 单元格，运行以呈现 (Render)；注意要用 \$（行内模式）或 \$\$（整行模式）包围

```
## 微积分相关公式

### 多元函数的泰勒展开
对于一个  $n$  元函数  $f(\mathbf{x})$ ，其中  $\mathbf{x}=(x_1,x_2,\cdots,x_n)^T$ ，在点  $\mathbf{a}=(a_1,a_2,\cdots,a_n)^T$  处的二阶泰勒展开式为：

$$f(\mathbf{x}) = f(\mathbf{a}) + \sum_{i=1}^n \frac{\partial f}{\partial x_i} \bigg|_{\mathbf{x}=\mathbf{a}} (x_i - a_i) + \frac{1}{2} \sum_{i,j=1}^n \frac{\partial^2 f}{\partial x_i \partial x_j} \bigg|_{\mathbf{x}=\mathbf{a}} (x_i - a_i)(x_j - a_j) + R_2(\mathbf{x})$$

其中  $R_2(\mathbf{x})$  是余项。

### 高斯积分
高斯积分是概率论和统计学中非常重要的积分，其表达式为：

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$


## 线性代数相关公式

### 矩阵的特征值与特征向量
对于一个  $n \times n$  的矩阵  $A$ ，如果存在一个非零向量  $v$  和一个标量  $\lambda$  使得  $Av = \lambda v$ ，则称  $\lambda$  是矩阵  $A$  的特征值， $v$  是对应于  $\lambda$  的特征向量。特征值  $\lambda$  满足特征方程：

$$\det(A - \lambda I) = 0$$

其中  $\det$  表示行列式， $I$  是  $n \times n$  的单位矩阵。

## 概率论相关公式

### 二维正态分布的概率密度函数
设  $(X,Y)$  服从二维正态分布  $N(\mu_1,\mu_2,\sigma_1^2,\sigma_2^2,\rho)$ ，其概率密度函数为：

$$f(x,y) = \frac{1}{2\pi\sigma_1\sigma_2\sqrt{1-\rho^2}} \exp\left\{-\frac{1}{2(1-\rho^2)}\left[\left(\frac{x-\mu_1}{\sigma_1}\right)^2 - 2\rho\left(\frac{x-\mu_1}{\sigma_1}\right)\left(\frac{y-\mu_2}{\sigma_2}\right) + \left(\frac{y-\mu_2}{\sigma_2}\right)^2\right]\right\}$$

其中  $\mu_1,\mu_2$  分别是  $X$  和  $Y$  的均值， $\sigma_1^2,\sigma_2^2$  分别是  $X$  和  $Y$  的方差， $\rho$  是  $X$  和  $Y$  的相关系数，且  $-1 < \rho < 1$ 。
```

复杂数学公式示例

微积分相关公式

多元函数的泰勒展开

对于一个 n 元函数 $f(\mathbf{x})$, 其中 $\mathbf{x} = (x_1, x_2, \dots, x_n)^T$, 在点 $\mathbf{a} = (a_1, a_2, \dots, a_n)^T$ 处的二阶泰勒展开式为:

$$f(\mathbf{x}) = f(\mathbf{a}) + \sum_{i=1}^n \frac{\partial f}{\partial x_i} \Big|_{\mathbf{x}=\mathbf{a}} (x_i - a_i) + \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n \frac{\partial^2 f}{\partial x_i \partial x_j} \Big|_{\mathbf{x}=\mathbf{a}} (x_i - a_i)(x_j - a_j) + R_2(\mathbf{x})$$

其中 $R_2(\mathbf{x})$ 是余项。

高斯积分

高斯积分是概率论和统计学中非常重要的积分, 其表达式为:

$$\int_{-\infty}^{\infty} e^{-x^2} dx = \sqrt{\pi}$$

线性代数相关公式

矩阵的特征值与特征向量

对于一个 $n \times n$ 的矩阵 \mathbf{A} , 如果存在一个非零向量 \mathbf{v} 和一个标量 λ 使得 $\mathbf{A}\mathbf{v} = \lambda\mathbf{v}$, 则称 λ 是矩阵 \mathbf{A} 的特征值, \mathbf{v} 是对应于 λ 的特征向量。特征值 λ 满足特征方程:

$$\det(\mathbf{A} - \lambda\mathbf{I}) = 0$$

5. 关闭前端页面, 在后端按 **Ctrl+C** 打断运行中的服务, 回到 **Bash** 提示符

修改 `environment.yml` 文件, 添加 `pip: tushare` (注意, conda-forge 没有收录 tushare, 只能从 PyPI 安装, 参考) 依赖项, 运行 `conda env update` 更新 Conda 环境, 在终端 (Terminal) 激活 week07 Conda 环境, 运行 `ipython` 命令启动 IPython 交互界面 (IPython 是 Jupyter 项目的一部份, ipython 是 jupyterlab 的依赖项之一), 在 IPython 提示符下, 运行下面的 Python 代码设置 Tushare Token, 其中 `***` 要替换成你在 Tushare 平台上的接口 **TOKEN** —— 复制粘贴即可。运行 `set_token` 函数会把 Token 字符串保存在 `~/tk.csv` 文件里, 今后每次使用 tushare 软件包请求数据时都会自动读取并发送 Token, 不需要反复设置。

按 **Ctrl+D** 结束前面的 IPython 进程, 再重新启动一个新的 IPython 进程, 运行下面的 Python 代码向 Tushare 服务器请

求 IPO 新股列表数据，并保存在本地

其中请求数据函数返回的对象 `df` 是 `pandas.DataFrame` 类型，调用其 `to_parquet` 方法能够将内存（memory）中的 `DataFrame` 数据按照 Parquet 格式（Parquet 是大数据领域的首选格式，已经成为业界标准）序列化（serialize）为字节串（bytes）保存到磁盘（disk）。

```
IPython: Crepo/week07
In [6]: df = pro.new_share()
In [7]: df
Out[7]:
```

	ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit_amount	funds	ballot	
0	301636.SZ	301636	泽润新能	20250428	None	1597.0		0.0	0.00	0.00	0.45	0.000	0.
00													
1	001400.SZ	001400	江顺科技	20250415	20250424	1500.0	1500.0	37.36	15.32		1.50	5.604	0.
01													
2	301560.SZ	301560	众捷汽车	20250415	None	3040.0	1216.0	16.50	21.30		0.70	5.016	0.
02													
3	603202.SH	732202	天有为	20250414	20250424	4000.0	2611.0	93.50	13.50		1.25	37.400	0.0
3													
4	301662.SZ	301662	宏工科技	20250408	20250417	2000.0	813.0	26.60	7.05		0.45	5.320	0.
02													
...
1995	002953.SZ	002953	日丰股份	20190424	20190509	4302.0	3872.0	10.52	16.34		1.70	4.526	0.
03													
1996	603697.SH	732697	有友食品	20190423	20190508	7950.0	7155.0	7.87	13.92		3.10	6.257	0.
05													
1997	300772.SZ	300772	运达股份	20190417	20190426	7349.0	6614.0	6.52	22.97		2.80	4.792	0.
04													
1998	603967.SH	732967	中创物流	20190417	20190429	6667.0	6000.0	15.32	22.24		2.60	10.213	0.
04													
1999	300773.SZ	300773	拉卡拉	20190416	20190425	4001.0	3601.0	33.28	22.99		1.20	13.315	0.0
4													

```
[2000 rows x 12 columns]
In [8]:
```

```
In [8]: df.to_parquet("new_share.parquet")
In [9]: df = pro.stock_basic(fields="ts_code,symbol,name,area")
In [10]: df
Out[10]:
```

	ts_code	symbol	name	area
0	000001.SZ	000001	平安银行	深圳
1	000002.SZ	000002	万科A	深圳
2	000004.SZ	000004	国华网安	深圳
3	000006.SZ	000006	深振业A	深圳
4	000007.SZ	000007	全新好	深圳
...
5409	920111.BJ	920111	聚星科技	None
5410	920116.BJ	920116	星图测控	None
5411	920118.BJ	920118	太湖远大	None
5412	920128.BJ	920128	胜业电气	None
5413	689009.SH	689009	九号公司-WD	北京

```
[5414 rows x 4 columns]
```



```

74567@DESKTOP-N5CDCLE MINGW64 ~/repo/week07 (main)
$ curl -O https://raw.githubusercontent.com/cueb-fintech/courses/blobs/8fc08f7bc4dbbf17d356234472795e59c7b9ce2f/stock_basic.parquet
% Total    % Received % Xferd  Average Speed   Time    Time     Time  Current
           Dload  Upload   Total   Spent    Left     Speed
100 422k    0 422k    0     0  309k      0  0:00:01 0:00:01 0:00:00 309k
74567@DESKTOP-N5CDCLE MINGW64 ~/repo/week07 (main)
$ ls -l
total 589
-rw-r--r-- 1 74567 197609 155 4月 23 20:32 environment.yml
-rw-r--r-- 1 74567 197609 18805 4月 23 09:57 LICENSE
-rw-r--r-- 1 74567 197609 121430 4月 23 20:43 new_share.parquet
-rw-r--r-- 1 74567 197609 2239 4月 23 09:57 README.md
-rw-r--r-- 1 74567 197609 432349 4月 23 20:52 stock_basic.parquet
-rw-r--r-- 1 74567 197609 17276 4月 23 19:58 trial-jupyterlab.ipynb
74567@DESKTOP-N5CDCLE MINGW64 ~/repo/week07 (main)
$ ls -lh
total 589K
-rw-r--r-- 1 74567 197609 155 4月 23 20:32 environment.yml
-rw-r--r-- 1 74567 197609 19K 4月 23 09:57 LICENSE
-rw-r--r-- 1 74567 197609 119K 4月 23 20:43 new_share.parquet
-rw-r--r-- 1 74567 197609 2.2K 4月 23 09:57 README.md
-rw-r--r-- 1 74567 197609 423K 4月 23 20:52 stock_basic.parquet
-rw-r--r-- 1 74567 197609 17K 4月 23 19:58 trial-jupyterlab.ipynb
74567@DESKTOP-N5CDCLE MINGW64 ~/repo/week07 (main)

```

询问豆包（或 DeepSeek 等任何大模型），初步了解 Parquet 格式和 CSV 格式的特点和适用领域

new_share 接口只需要 120 积分，如果你有 2000 积分，可以采用与上面类似的方法访问 stock_basic 接口，并将数据保存为 stock_basic.parquet 文件（注意，需要指定 fields 参数获取全部字段）。如果积分暂时不够，可以在终端运行下面的命令，从我们开源的课程仓库下载数据文件到你的本地。

6. 通过 perspective-python 软件包查看 polars.DataFrame 数据，实践交互式可视化：

```
[12]: d1 = d1.with_columns(  
      ipo_date=pl.col.ipo_date.str.to_date("%Y%m%d"),  
      issue_date=pl.col.issue_date.str.to_date("%Y%m%d"),  
      )
```

```
[13]: d2 = d2.with_columns(  
      list_date=pl.col.list_date.str.to_date("%Y%m%d"),  
      )
```

```
[16]: type(d1)
```

```
[16]: polars.dataframe.frame.DataFrame
```

```
[ ]: type(d2)
```

```
[17]: d1
```

```
[17]: shape: (2_000, 12)
```

ts_code	sub_code	name	ipo_date	issue_date	amount	market_amount	price	pe	limit_amount	funds	ballot
str	str	str	date	date	f64	f64	f64	f64	f64	f64	f64
"301636.SZ"	"301636"	"泽润新能"	2025-04-28	null	1597.0	0.0	0.0	0.0	0.45	0.0	0.0
"001400.SZ"	"001400"	"江顺科技"	2025-04-15	2025-04-24	1500.0	1500.0	37.36	15.32	1.5	5.604	0.01
"301560.SZ"	"301560"	"众捷汽车"	2025-04-15	null	3040.0	1216.0	16.5	21.3	0.7	5.016	0.02
"603202.SH"	"732202"	"天有为"	2025-04-14	2025-04-24	4000.0	2611.0	93.5	13.5	1.25	37.4	0.03
"301662.SZ"	"301662"	"宏工科技"	2025-04-08	2025-04-17	2000.0	813.0	26.6	7.05	0.45	5.32	0.02
...
"002953.SZ"	"002953"	"日丰股份"	2019-04-24	2019-05-09	4302.0	3872.0	10.52	16.34	1.7	4.526	0.03
"603697.SH"	"732697"	"有友食品"	2019-04-23	2019-05-08	7950.0	7155.0	7.87	13.92	3.1	6.257	0.05
"300772.SZ"	"300772"	"运达股份"	2019-04-17	2019-04-26	7349.0	6614.0	6.52	22.97	2.8	4.792	0.04
"603967.SH"	"732967"	"中创物流"	2019-04-17	2019-04-29	6667.0	6000.0	15.32	22.24	2.6	10.213	0.04
"300773.SZ"	"300773"	"拉卡拉"	2019-04-16	2019-04-25	4001.0	3601.0	33.28	22.99	1.2	13.315	0.04