## 一.pycharm下载安装

pycharm下载地址：  
http://www.jetbrains.com/pycharm/download/#section=windows  
下载详细步骤：

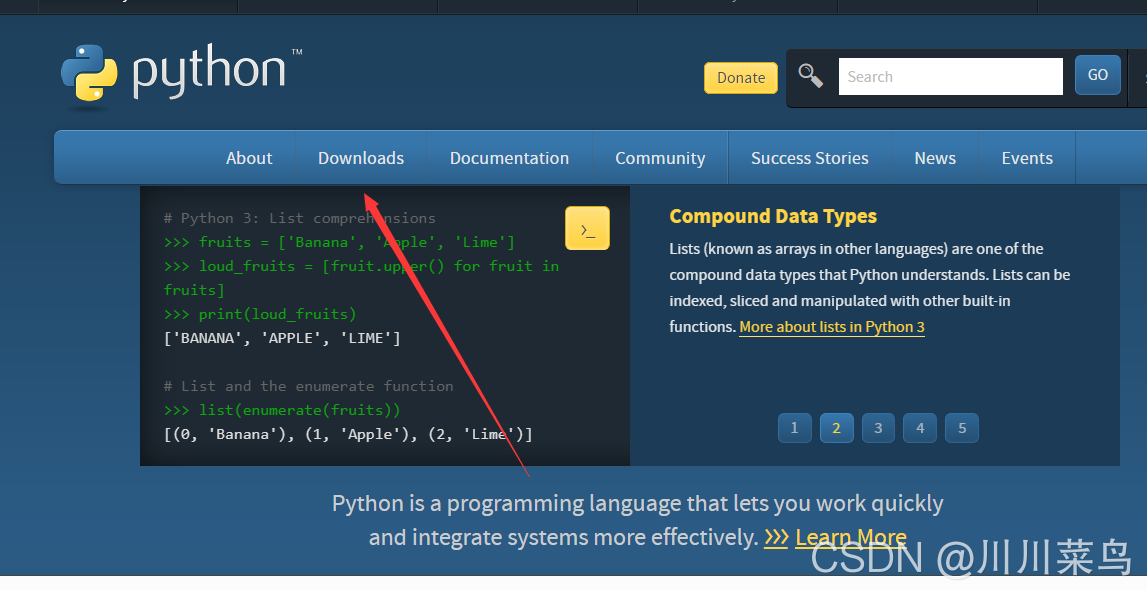


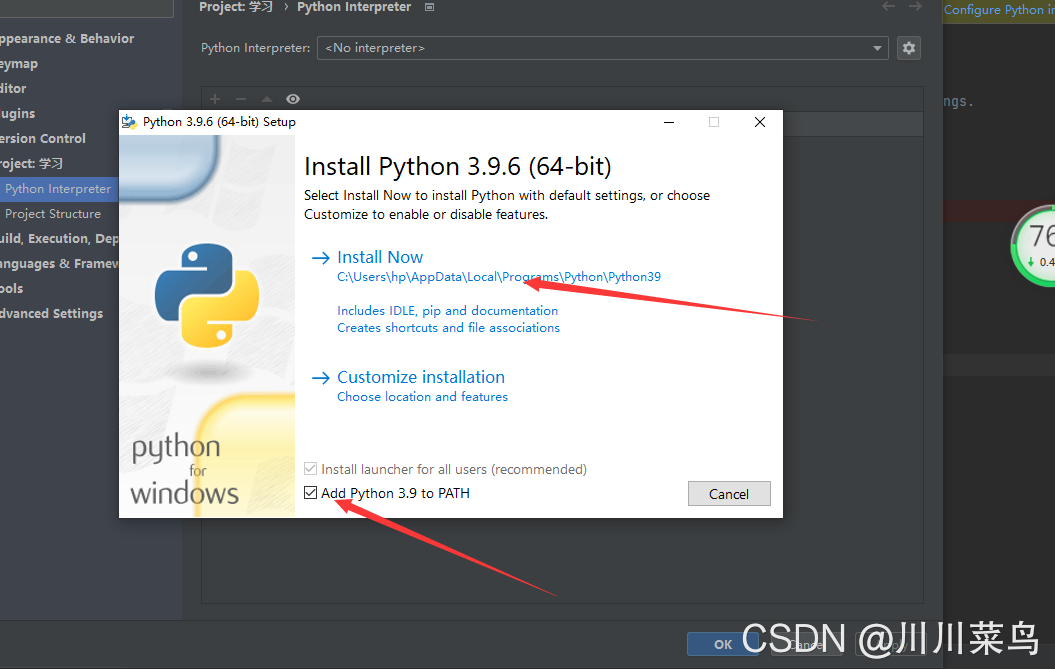
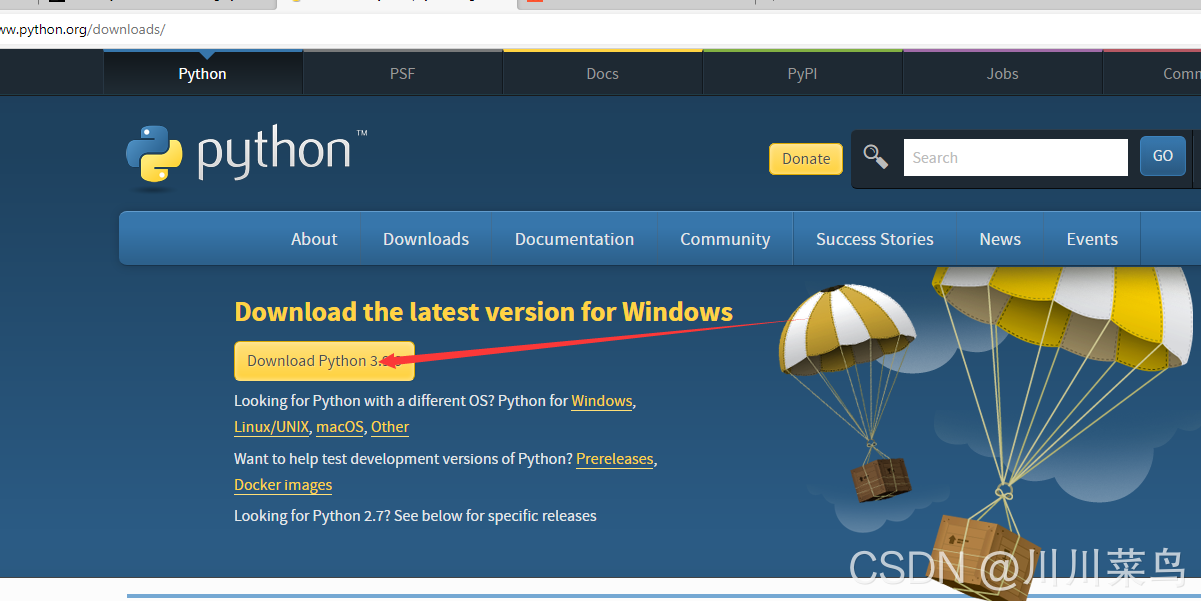
有专业版和免费版，我们在这使用的是专业版

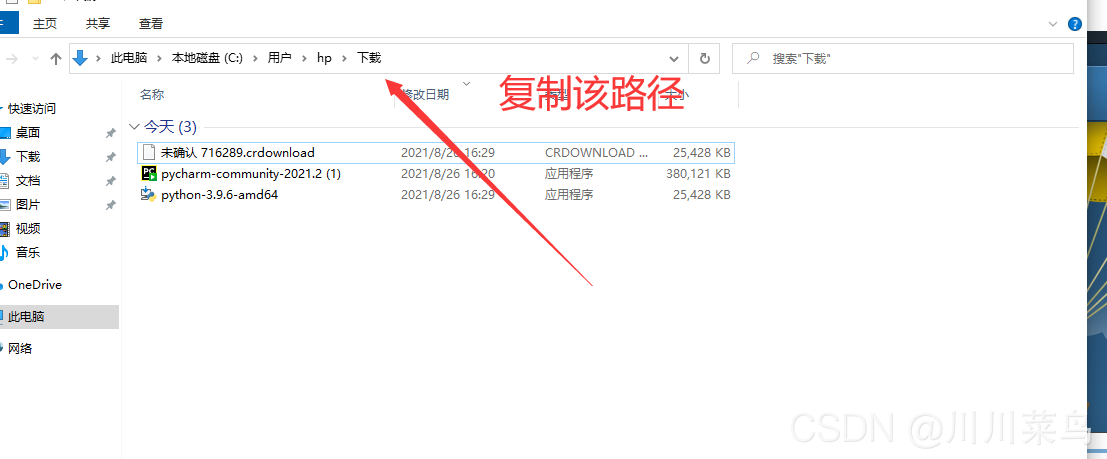
**二.**[python](https://so.csdn.net/so/search?q=python&spm=1001.2101.3001.7020)**下载安装**

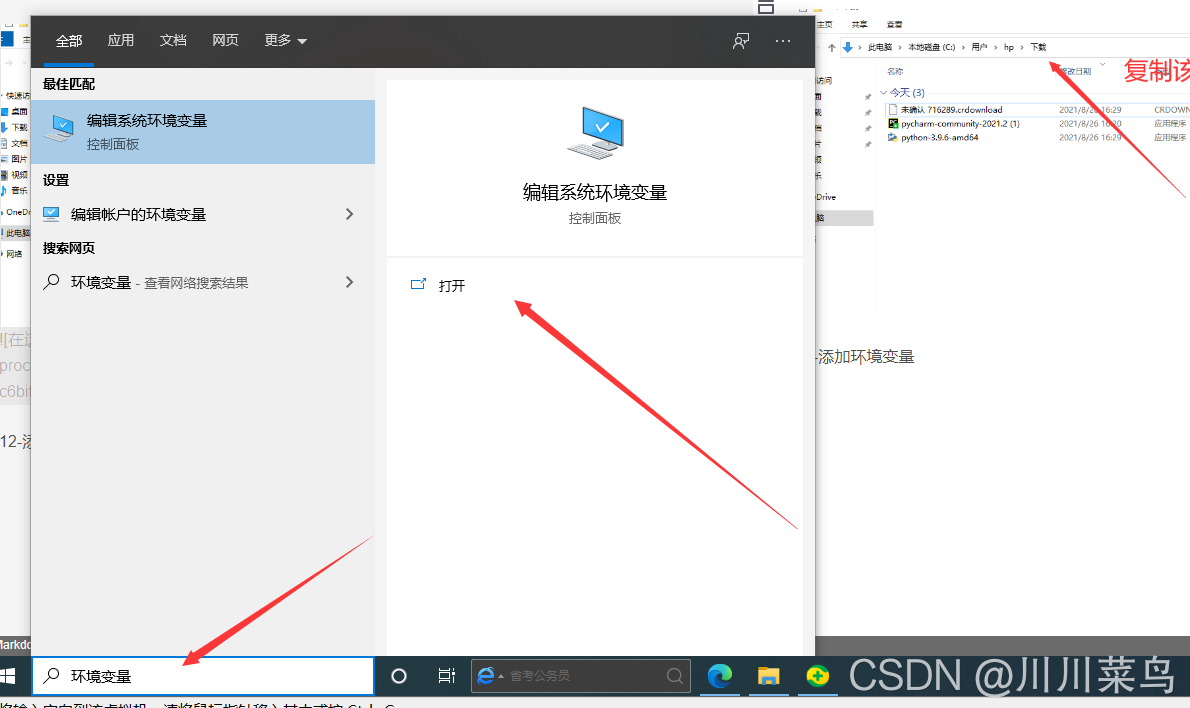
9-python官网：

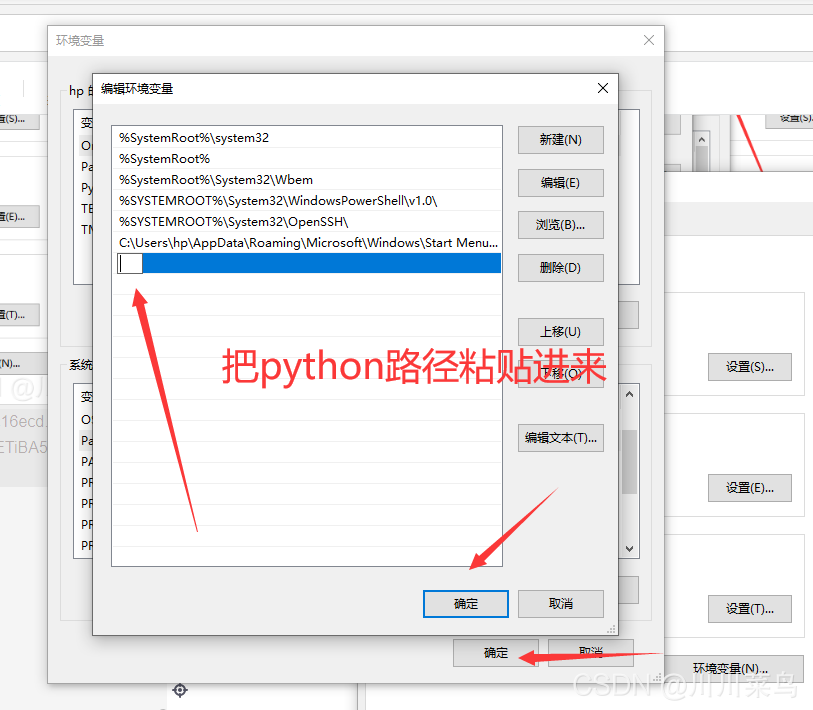
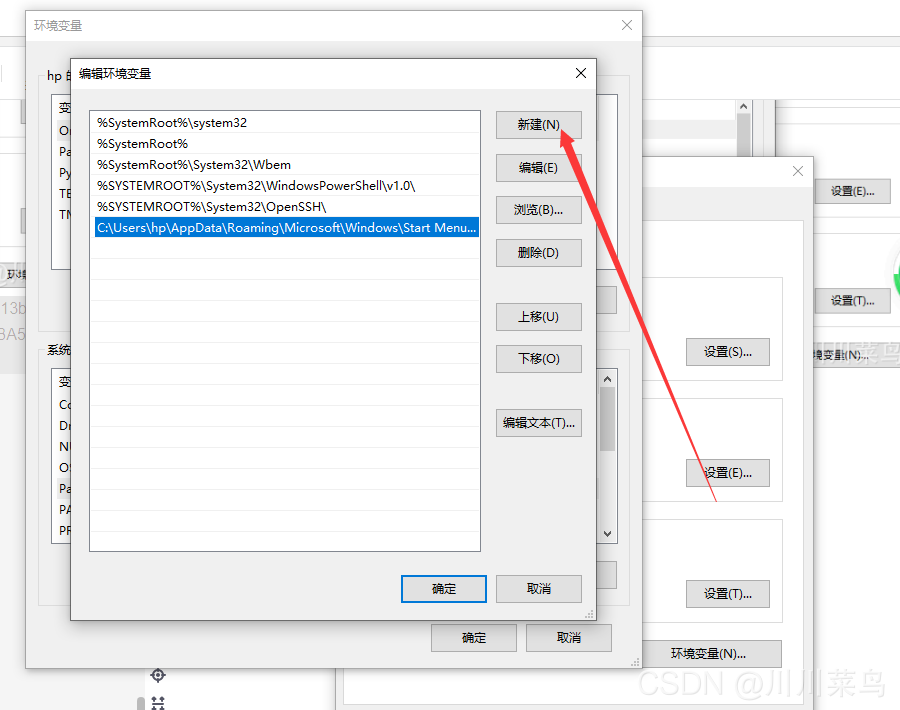
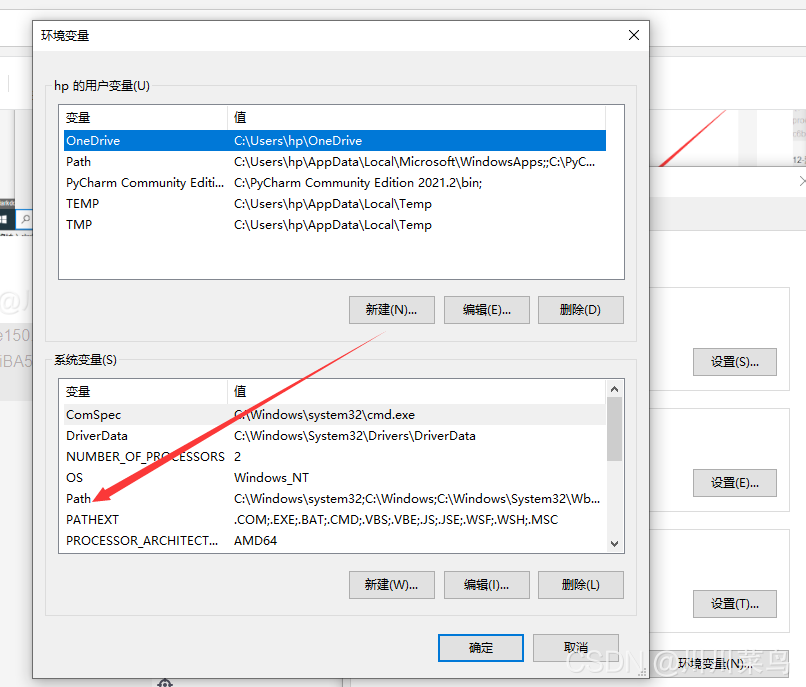
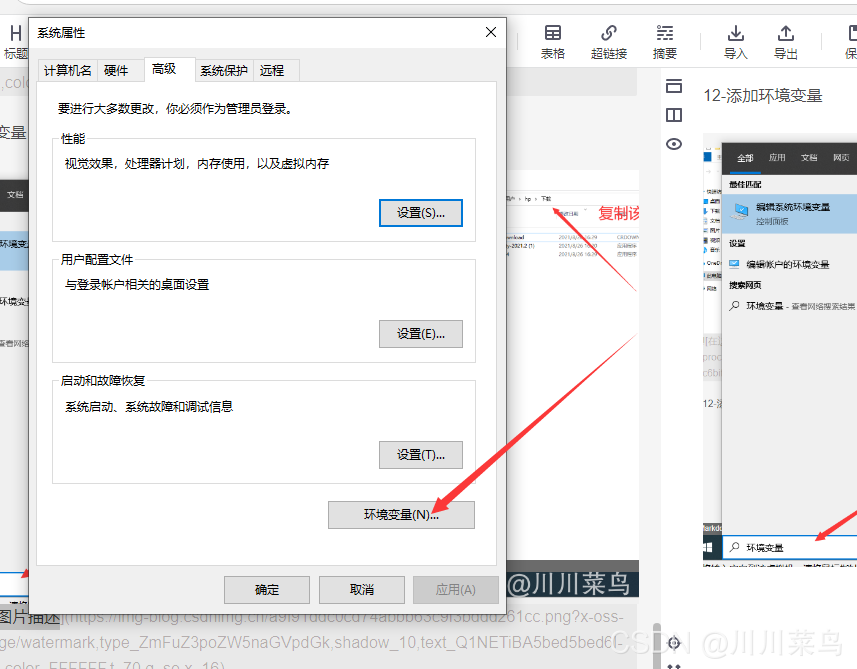
https://www.python.org/





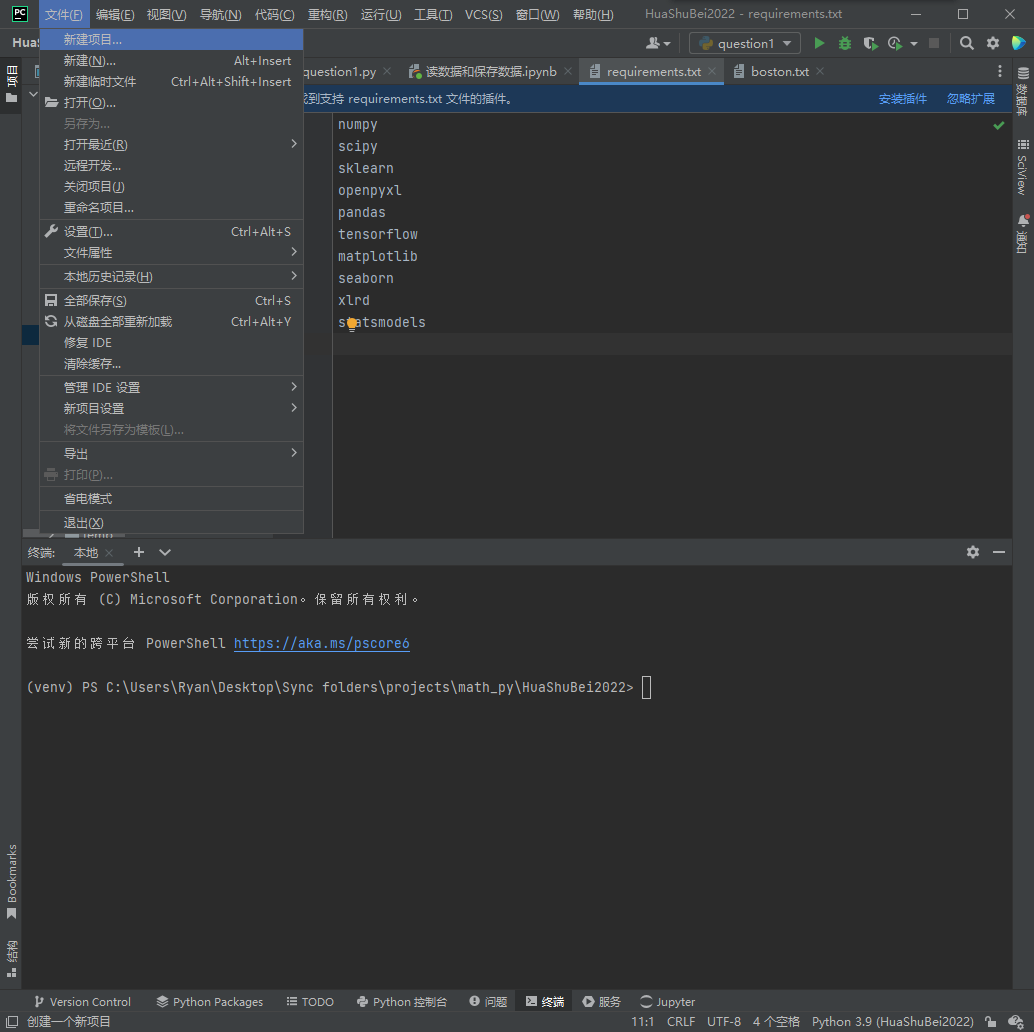


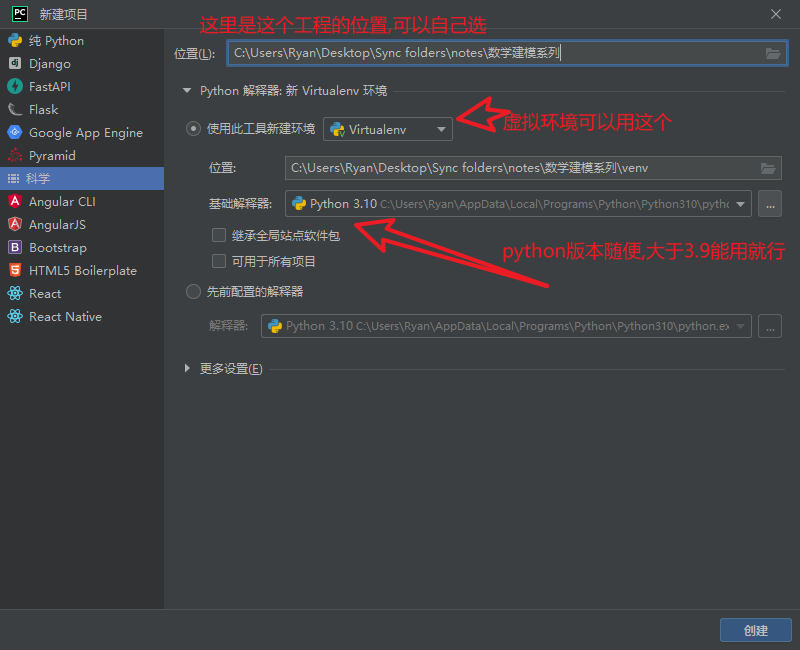




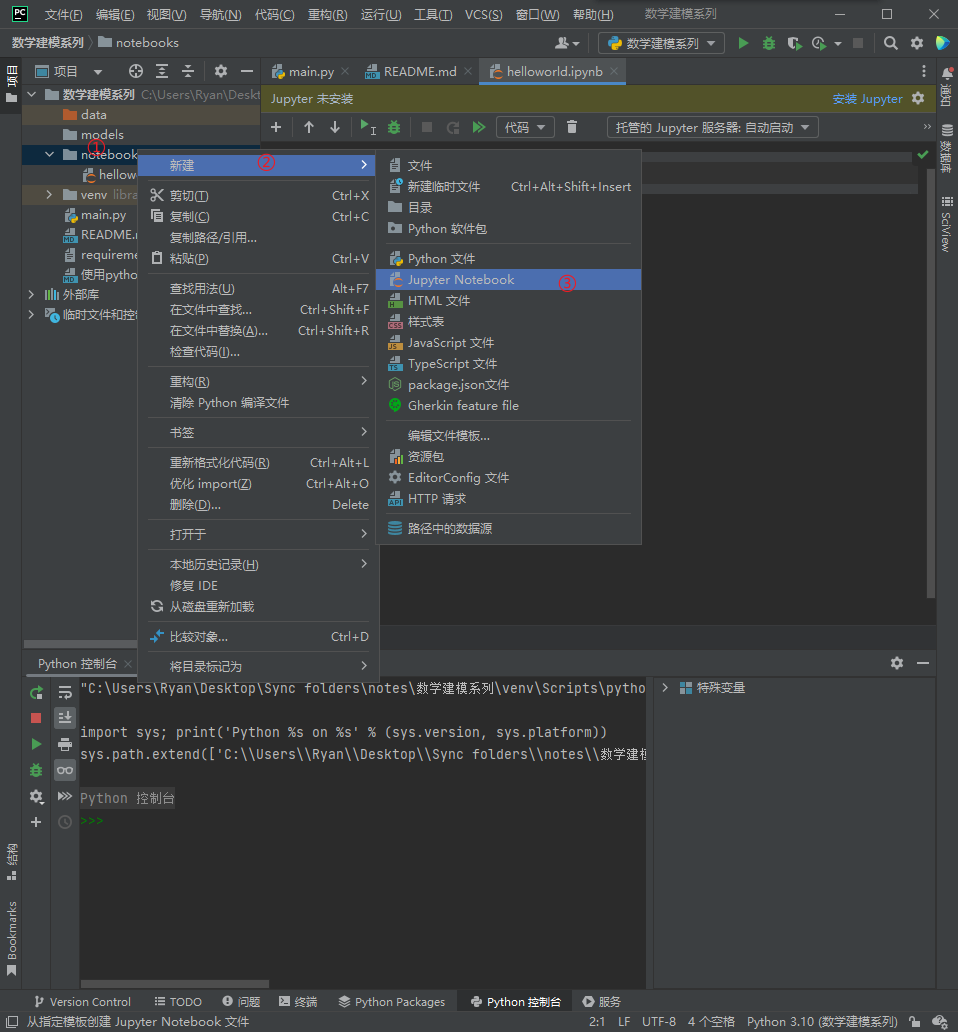
## 三、进行环境部署

1. 使用pycharm新建一个工程



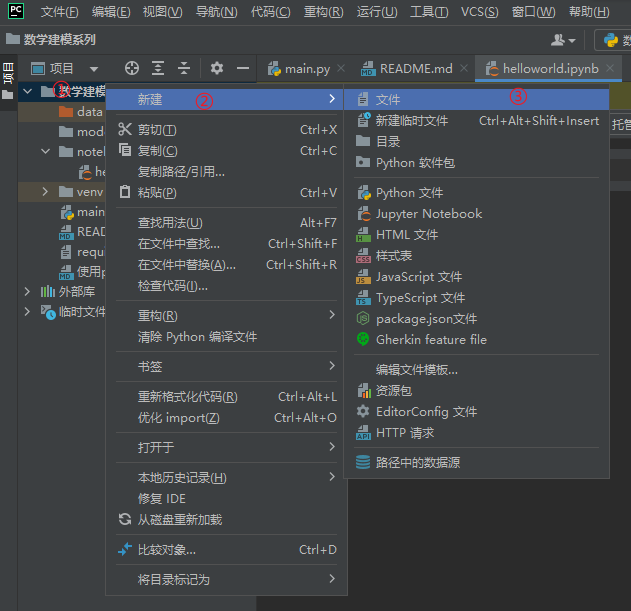


新建一个jupyter notebook文件,名字随便起



文件新建好了之后会看到这么个提示: jupyter没安装

新建一个requirements.txt 文件,在requirements.txt 文件里





里边的内容如下(复制粘贴保存即可).

Jupyter

numpy

scipy

sklearn

openpyxl

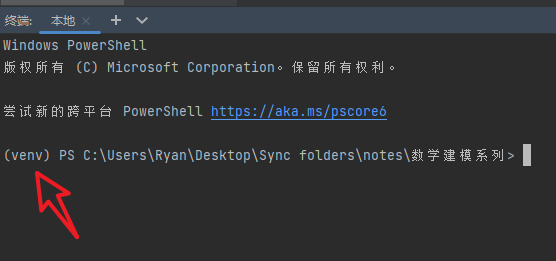
pandas

matplotlib

seaborn

xlrd

点击左下角终端查看是否有（venv）字样，如果有说明已经进入虚拟环境



**打开终端,输入以下内容即可安装所需要的主流的python的包**

**pip install -r requirements.txt**

## 四、运行代码

## 复制代码

import pandas as pd  
import numpy as np  
import matplotlib.pyplot as plt  
  
# Prior probability values  
prior\_probs = [0.2, 0.3, 0.5]  
  
# 读取文件数据  
filename = 'C:/Users/金煜/PycharmProjects/scientificProject2/file.xlsx'  
data = pd.read\_excel(filename, usecols=[11, 12, 13, 14])  
  
# 检查数据列名  
if 'Handedness' in data.columns:  
 data = data.drop('Handedness', axis=1)  
  
# 将空值替换为0  
data = data.fillna(0)  
  
# 使用贝叶斯公式  
counts = np.zeros((data.shape[0], 3))  
for i in range(data.shape[0]):  
 if data.iloc[i, 0] != 0 and data.iloc[i, 1] != 0 and data.iloc[i, 2] != 0 and data.iloc[i, 3] != 0:  
 counts[i, 0] = (data.iloc[i, 0] / data.iloc[i, 3]) \* prior\_probs[0]  
 counts[i, 1] = (data.iloc[i, 1] / data.iloc[i, 3]) \* prior\_probs[1]  
 counts[i, 2] = (data.iloc[i, 2] / data.iloc[i, 3]) \* prior\_probs[2]  
# 删除所有特征值都是0的行  
counts = counts[~np.all(counts == 0, axis=1)]  
# 计算后验概率  
posterior\_probs = counts / counts.sum(axis=1)[:, None]  
# 输出后验概率值  
print('贝叶斯公式输出的值:\n', posterior\_probs)  
# 输出特征的概率值  
print('特征的概率值:\n', counts)  
# 绘制柱状图  
fig, ax = plt.subplots()  
index = np.arange(3)  
bar\_width = 0.35  
opacity = 0.8  
  
rects1 = ax.bar(index, counts.mean(axis=0), bar\_width,  
 alpha=opacity, color='b',  
 label='Mean probability')  
  
rects2 = ax.bar(index + bar\_width, counts.max(axis=0), bar\_width,  
 alpha=opacity, color='g',  
 label='Max probability')  
  
# 添加先验概率柱状图  
rects3 = ax.bar(index, prior\_probs, bar\_width,  
 alpha=opacity, color='r',  
 label='Prior probability')  
  
ax.set\_xlabel('Feature')  
ax.set\_ylabel('Probability')  
ax.set\_title('Probability of Each Feature')  
ax.set\_xticks(index + bar\_width / 2)  
ax.set\_xticklabels(('UPDRS\_ON', 'UPDRS\_OFF', 'Yrs Since Diagnosis'))  
ax.legend()  
  
plt.tight\_layout()  
plt.savefig('概率柱状图.png') # 保存图像为名为 概率柱状图.png 的文件  
plt.show()  
# 绘制折线图  
plt.plot(counts[:, 0], label='UPDRS\_ON')  
plt.plot(counts[:, 1], label='UPDRS\_OFF')  
plt.plot(counts[:, 2], label='Yrs Since Diagnosis')  
  
# 添加先验概率折线  
plt.plot(prior\_probs, label='Prior probability', linestyle='--')  
  
plt.legend()  
plt.title('Bayesian Formula Output')  
plt.savefig('贝叶斯公式.png') # 保存图像为名为贝叶斯公式.png 的文件  
plt.show()

## 注意：此行代码是读取数据来源excel文件，打开途径请自己改动

## 注意：此行代码是读取数据来源excel文件，打开途径请自己改动

## 运行此代码

## 

## 得出利用贝叶斯公式计算3个变量的二维图像

## 和一张带有先验概率和特征概率的柱形图

## 

## 和贝叶斯公式输出的值，特征的概率值:

## 