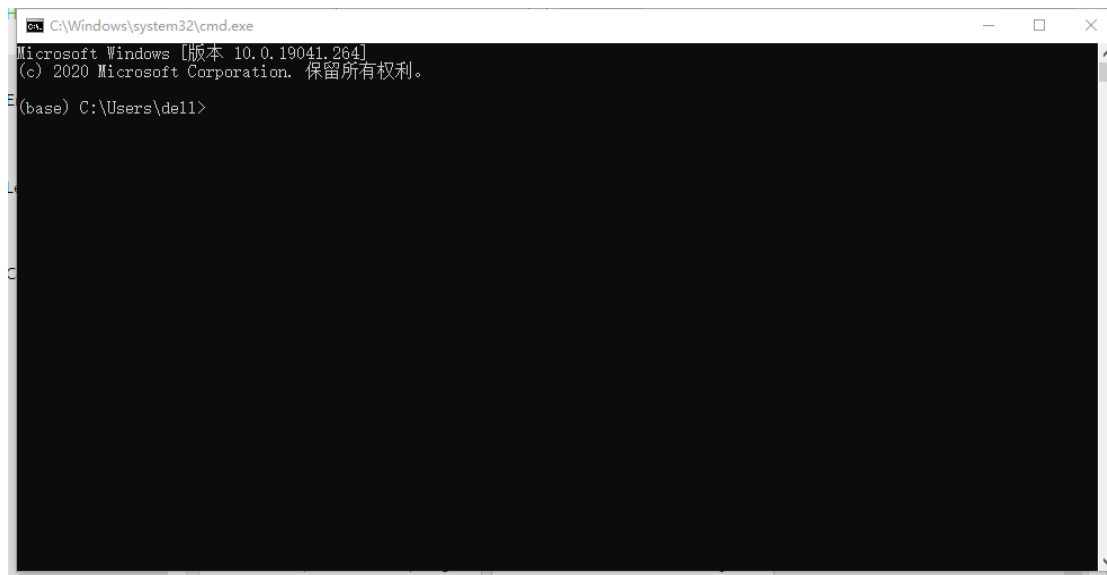


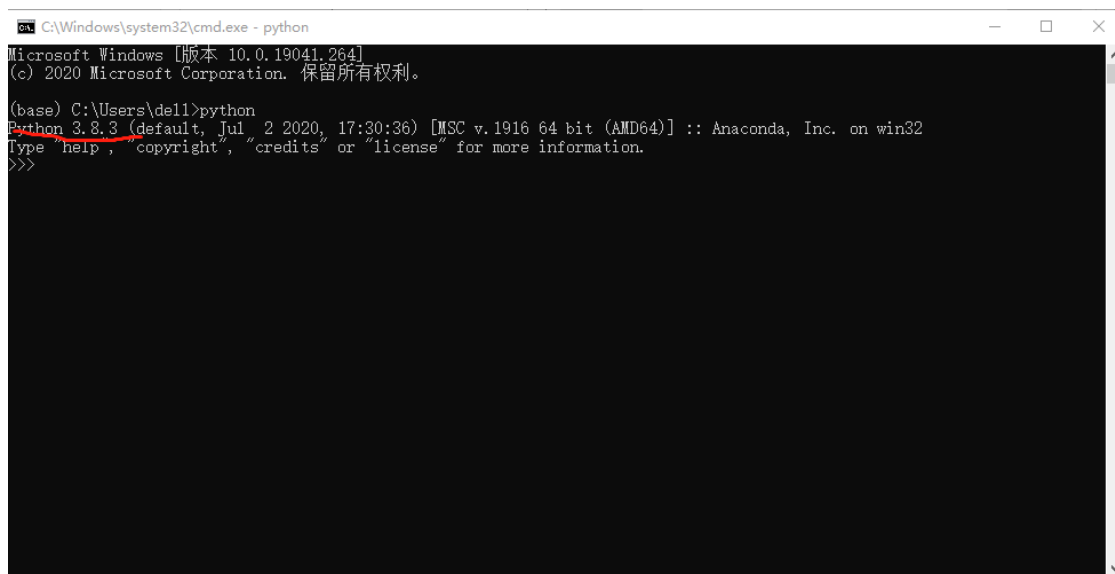
Anaconda 软件

一、 CMD.exe.pormpt (IDLE) 命令提示符



1、 python 环境

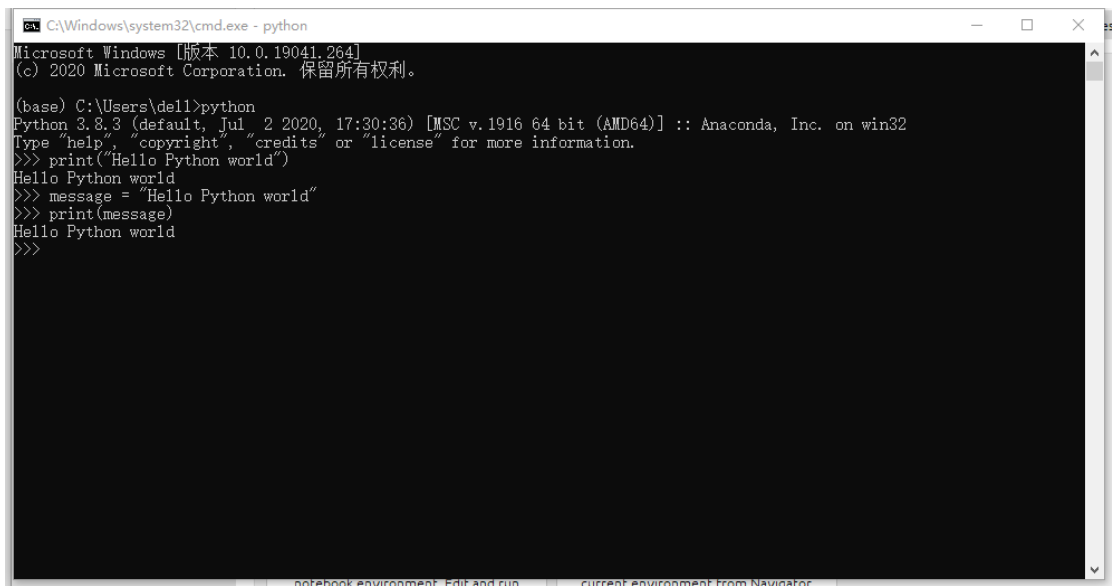
计算机已经安装了 python，并且添加 python 环境变量的，可以输入：python 输入可知 python 版本。



2、 输入命令

(1) 直接用 `print()` 函数输出 “Hello Python world”

(2) 通过 `message` 变量输出“Hello Python world”



```
C:\Windows\system32\cmd.exe - python
Microsoft Windows [版本 10.0.19041.264]
(c) 2020 Microsoft Corporation. 保留所有权利。

(base) C:\Users\dell>python
Python 3.8.3 (default, Jul 2 2020, 17:30:36) [MSC v.1916 64 bit (AMD64)] :: Anaconda, Inc. on win32
Type "help", "copyright", "credits" or "license()" for more information.
>>> print("Hello Python world")
Hello Python world
>>> message = "Hello Python world"
>>> print(message)
Hello Python world
>>>
```

3、简单应用（编辑器等同）

(1) `print(3+5)`

```
>>> print(5+3)
```

(2) 直接输入：3+5

```
>>> 5+3
```

(3) 高位数的数据计算：987654321*123456789

```
>>> 123456789*987654321
```

(4) 字符串之间的相加

```
>>> print("well water" + "river")
```

(5) 字符串*数值

```
>>> print("I love python\n"*5)
```

(6) 转义符 `\n`

```
>>> print("I love python\n"*5)
```

(7) 字符串 + 数值

报错，字符串（`str`）不可以与整形（`int`）进行相加

```
>>> print("I love python\n" + 8)
Traceback (most recent call last):
  File "<stdin>", line 1, in <module>
TypeError: can only concatenate str (not "int") to str
```

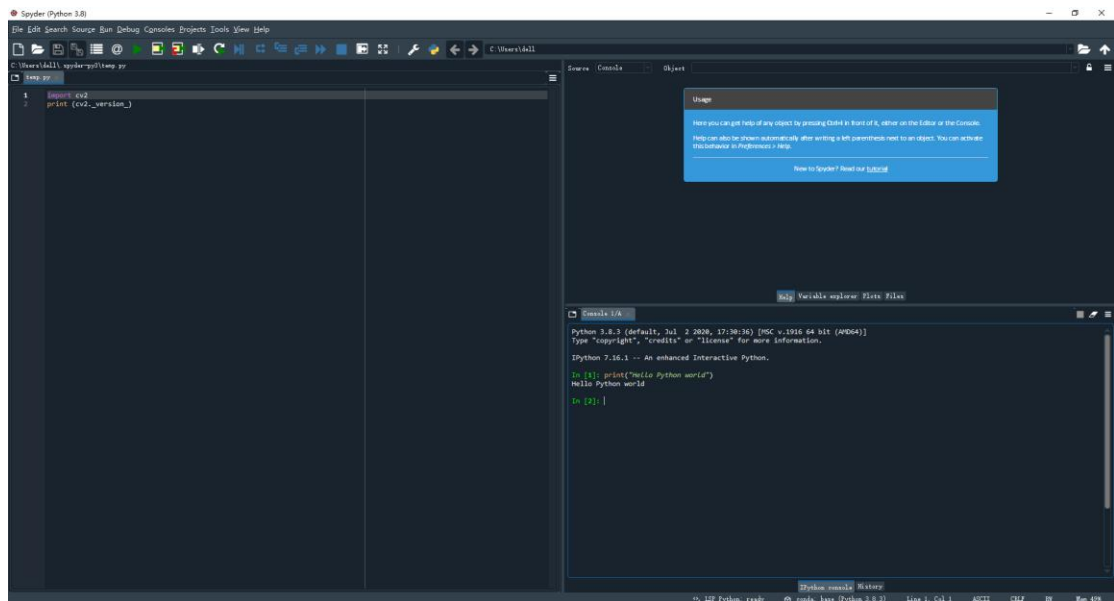
二、 Spyder (python 编辑器)

1、 界面介绍:

左处为程序编辑区域

右上方: 选择“variable explorer”: 变量管理窗口, 存储变量值、图像等

右下方: 程序输出窗口, 也是 IDLE 窗口



2、 编写程序:

(1) 注释: 可以用# 或者 “""" """ (三组双引号)

```
1 #python
2
3 """python"""
```

(2) 函数 str() 类型使用:

错误:

```
1 age = 22
2 message = "happy" + age + "rd Birthday!"
3 print(message)
4
```

变量 age 可能表示数值 23, 也可以表示字符 2 和 3, 需要调用 str() 函数。将非字符串表示为字符串:

正确:

```

1 age = 22
2 message = "happy" + str(age) + "rd Birthday!"
3 print(message)
4

```

(3) 列表用 `[]` 表示。逗号分隔元素

列表的输出以及访问列表元素（列表元素索引从 0 开始，而非 1 开始）

打印列表长度函数：`len()`

```

1 English = ['a','b','c','d','e']
2
3 #输出列表:
4 print(English)
5
6 #输出列表各个元素
7 print(English[0])
8 print(English[1])
9 print(English[2])
10 print(English[3])
11 print(English[4])
12
13
14 print(len(English))

```

(4) for 循环，打印列表中所有程序：

```

1 English = ['a','b','c','d','e']
2
3 for eng in English:
4     print(eng)

```

For 循环中，python 从列表中取出一个元素，放到变量 `eng` 中，通过 `print()` 函数将每一次变量中存储的值打印出来

(5) if 语句：

```

1 age = input("输入你的年龄: ")
2
3 Age = int(age)
4
5 if Age >= 18:
6     print("恭喜你，你已经拥有投票的权力!")
7 else:
8     print("不好意思，你还小!")

```

`Input()` 函数表示输入数值，程序运行之后可在程序输出框中输入你的年龄。

(6) 定义函数：

```

1  #定义函数greet_user(),表示为问候语
2  def greet_user():
3      """显示问候语"""
4      print("Hello!")
5
6  #调用函数
7  greet_user()
8

```

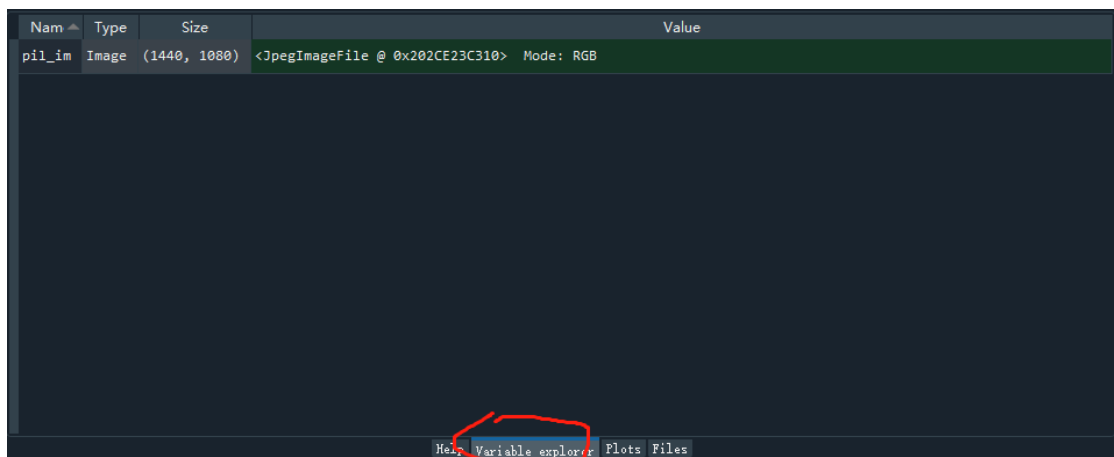
(7) 读取图像和打开图像

其中图像 “a1.jpg” 是与程序处于同一个文件夹内，否则程序会报错。

```

1  #调用PIL图像处理库中的Image模块
2  from PIL import Image
3
4  #通过Image.open('图像名称')读取图片，读取图像命名为pil_im
5  pil_im = Image.open('a1.jpg')
6
7  #打开图像
8  pil_im.show()

```



(1) 读取图像并转化为灰度图像

`Pil_im1 = Image.open(' ').convert('L')`

```

1  #调用PIL图像处理库中的Image模块
2  from PIL import Image
3
4  #转化为灰度图像，读取图像命名为pil_im1
5  pil_im1 = Image.open('a1.jpg').convert('L')
6
7  #打开图像
8  pil_im1.show()

```

(2) 转换图像格式

将文件夹内图片转化为‘png’格式，将无法转化的文件名输出。

```
1  from PIL import Image
2  #os库用于调用文件路径path
3  import os
4
5  """ 获取文件名列表地址，
6  将文件路径中文件存入的filelist文件名列表中
7  文件夹路径需要用反斜杠 '/' 代替 '\\'
8  防止类似'\n' 字符的出现，导致文件报错
9  """
10 filelist = os.listdir("E:/file/pypractice")
11
12
13 # 开始遍历文件，将路径中每一个文件一个一个存入infile中
14 for infile in filelist:
15     try:
16
17         """
18         将文件名和文件类型分开，
19         文件名存入outfile_name中，
20         文件格式存入outfile_type中
21         """
22         outfile_name,outfile_type = os.path.splitext(infile)
23
24         #如果图片格式为'png',继续返回for循环
25         if outfile_type == 'png':
26             continue
27         else:
28             #img.save("%s.jpg"%(name[0:(-type_length-1)]), 'jpeg')
29             #修改：改用下面这行代码：
30             Image.open(infile).save("%s.png"%(outfile_name), 'png')
31
32         #非图像文件报错，并输出文件名
33     except IOError:
34         print ("error",outfile_name)
35
```