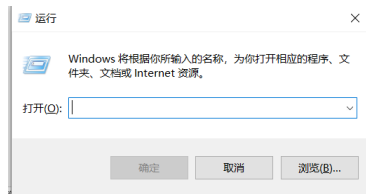
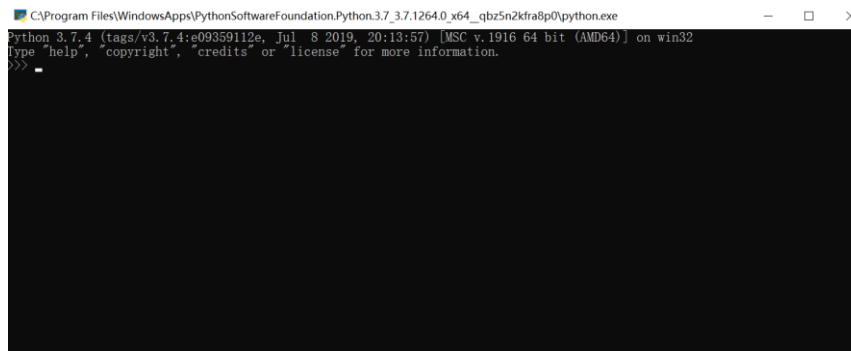


打开 python 终端

首先同时按住键盘上的 win+R 键，打开命令行窗口。



在搜索框中输入 cmd，回车，
然后在 cmd 中输入 python3，回车，
这样就打开了 python 终端。



1. 依次在 python 终端中输入以下表达式，并观察结果（注意：>>>是不需要输入的）

```
>>> 3
>>> 12.5
>>> True
>>> 7 / 4
>>> (2 + 6) / 4
>>> 7 // 4          # Floor division (rounding down)
>>> 7 % 4           # Modulus (remainder of 7 // 4)
>>> a = (100 + 50) // 2
>>> a
```

2. 在不运行以下代码的情况下，计算出每个表达式的值，然后再输入终端中，检验自己是否计算正确。

```
>>> 10 + 2
```

```
_____
```

```
>>> 7 / 2
```

```
_____
```

```
>>> 7 // 2

_____

>>> 7 % 2 #7 modulo 2, equivalent to the remainder of 7// 2

_____

>>> x = 20

>>> x + 2

_____

>>> x

_____

>>> y = 5

>>> y += 3                # Equivalent to y = y + 3

>>> y * 2

_____

>>> y //= 4                # Equivalent to y = y // 4
```

3. 编写函数

```
def twenty_nineteen():

    """Come up with the most creative expression that evalu
ates to 2019,

    using only numbers and the +, *, and - operators.

    >>> twenty_nineteen()

    2019

    """

    return _____
```

函数中用"""包围的字符串是用来解释函数作用的，称为 **docstring**

用`>>>`开头的称为 **doctests**，在本函数中，当我们在 `python` 终端中完成函数定义之后，再输入 `twenty_nineteen()`，会输出 2019。你的任务是用任意一个值为 2019 的表达式替换“**return** _____”中的下划线。

4. 比较三种除法相关的操作，依次输入以下表达式，注意观察结果。

True Division: / (decimal division)	Floor Division: // (integer division)	Modulo: % (remainder)
<pre>>>> 1 / 5 0.2 >>> 25 / 4 6.25 >>> 4 / 2 2.0 >>> 5 / 0 ZeroDivisionError</pre>	<pre>>>> 1 // 5 0 >>> 25 // 4 6 >>> 4 // 2 2 >>> 5 // 0 ZeroDivisionError</pre>	<pre>>>> 1 % 5 1 >>> 25 % 4 1 >>> 4 % 2 0 >>> 5 % 0 ZeroDivisionError</pre>

`x % y == 0` 可以用来判断 `x` 是否可以被 `y` 整除。

5. 依次输入函数定义和函数调用的代码，熟悉函数定义和调用操作。（注意 `python` 函数的对齐问题）

```
def foo(x):
    return x * 3 + 2
```

```
>>> foo(1)
5
>>> foo(2)
8
>>> foo(1000)
3002
```

6. `print`, `return` 和 `None`

`print` 函数是用来在终端中打印数值的，而在终端中调用函数会打印出函数返回值。在终端中输入以下代码，观察运行结果。

```
def what_prints():
```

```
print('Hello World!')

return 'Exiting this function.'

print('61A is awesome!')

>>> what_prints()
Hello World!
```

直接在 `python` 中输入 `None` 或者值为 `None` 的变量，`None` 不会输出到屏幕，我们可以用 `print(None)` 来输出 `None`

```
>>>None

>>> a = None

>>> a

>>> print(None)

>>> print(print(None), print(None))

>>> print(print(1), print(2))
```