

实用python编程 第2讲

# python基础

2017-09-18

## 本节内容

- 编程工具
- 语言基础

实用python编程-2017下

工欲善其事，必先利其器

python编程工具

实用python编程-2017下

## 代码编辑器

- sublime text

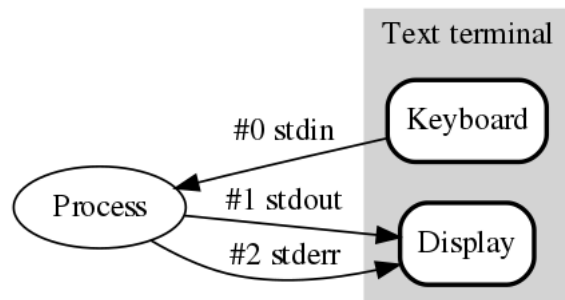
<https://github.com/ruc-python/2017fall/blob/master/sublime-intro.pdf>



实用python编程-2017下

## 运行python

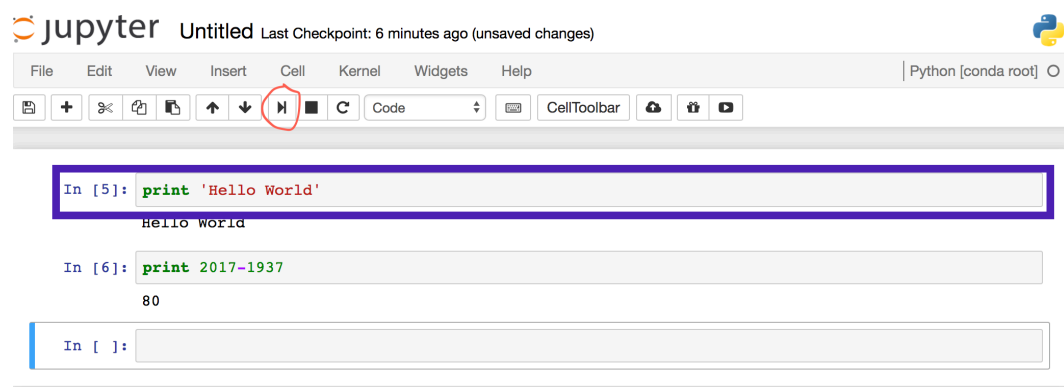
- 终端 Terminal



实用python编程-2017下

## 运行python

- jupyter notebook 交互式编程更方便



实用python编程-2017下

# python语言

实用python编程-2017下

python是一个计算器



实用python编程-2017下

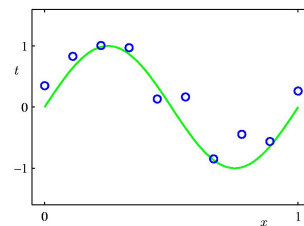
## 用jupyter notebook做加减乘除

- $1+1$
- $1*2$
- $1+2*3-4$
- $1 / 2$
- $1 / 2.0$
- `"1" + "1"`
- $1 + "1"$
- $5**4$
- 对5开根号怎么表示？

实用python编程-2017下

## 高级一点的运算

- 幂函数  $f(x)=x^a \Rightarrow x**a$ 
  - 多项式回归分析
- 指数函数  $f(x)=e^x \Rightarrow \text{math.exp}(x)$
- 对数函数  $f(x)=\log_b(x) \Rightarrow \text{math.log}(x, b)$ 
  - 二进制对数  $\ln(x) \Rightarrow \text{math.log}(x, 2)$
  - 自然对数  $\ln(x) \Rightarrow \text{math.log}(x)$



$$\ln(e^x)=x$$

实用python编程-2017下

## python的变量是一个容器

- 可以装各种类型的数据
- 一般形式  
变量名 = 表达式

```
x = 1
y = 2
print (x,y)
y = x
x = 5
print (x,y)
```

动态展示python运行过程 <http://www.pythontutor.com/visualize.html>

实用python编程-2017下

## 不能用python保留字作为变量名

- python保留字

```
import keyword
print keyword.kwlist
```

```
['and', 'as', 'assert', 'break', 'class', 'continue',
'def', 'del', 'elif', 'else', 'except', 'exec',
'finally', 'for', 'from', 'global', 'if', 'import',
'in', 'is', 'lambda', 'not', 'or', 'pass', 'print',
'raise', 'return', 'try', 'while', 'with', 'yield']
```

实用python编程-2017下

## 课堂练习1（用jupyter notebook）

- 假设一个城市的人口P和年份存在如下的关系

$$P = 250,000 e^{0.012t}$$

t=0表示2017年, t=1表示2018年, ...

- 问题
  - 2020年的人口数是多少?
  - 什么时间人口数达到400,000?

实用python编程-2017下

## python函数是一个积木组件

- 一个函数负责一个功能
- 一次定义，无限次使用



实用python编程-2017下

## 一个例子

- $f(x)=x^a$

```
def power_function(x, a):  
    return x**a
```

`def` python保留字，表明这是一个函数  
`power_function` 函数名称，命名规则同变量  
`x, a` 传给函数的参数名（形式参数）  
`return` python保留字，用于返回计算结果

实用python编程-2017下

## 在函数中调用其他函数

```
def power_function(x, a):  
    return x**a  
  
def square_function(x, a):  
    return power_function(x, 2)
```

实用python编程-2017下



这个函数干什么用的？

```
def xxx(yyy):  
    return yyy/0.62
```

实用python编程-2017下

使用有意义的函数名

```
def to_miles(kilometers):  
    return kilometers/0.62
```

实用python编程-2017下

## 课堂练习2 (用sublime text)

- 用函数的形式解决课堂练习1的问题

```
def compute_population(t):  
    # your code goes here
```

```
def which_year(P):  
    # your code goes here
```

实用python编程-2017下

## 控制语句

- 条件控制
  - if
  - if ... else
  - if ... elif
  - if ... elif ... elif
  - if ... elif .. elif ... else
- 循环控制
  - for
  - while

实用python编程-2017下

## 条件控制

**if** 条件表达式:

语句1

**else:**

语句2

**if** 条件1:

语句1

**elif** 条件2:

语句2

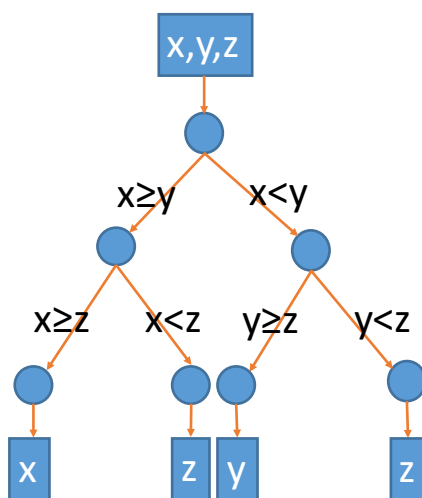
.....

**else:**

语句n

实用python编程-2017下

## 条件控制



如何求三个数中最大的数？

```

def my_max(x,y,z):
    if x >= y:
        if x >= z:
            return x
        else:
            return z
    else:
        if y >= z:
            return y
        else:
            return z
  
```

实用python编程-2017下

## 循环控制：for循环

**for** 变量 **in** 循环范围：  
    循环语句

在jupyter notebook中观察以下代码的运行结果

```
for i in range(10):  
    print i
```

实用python编程-2017下

## 循环控制：for循环的例子

- 如何求  $1+2+\dots+99+100$ ？

```
def my_sum(n):  
    if n <= 0:  
        print 'n must be greater than 0'  
        return  
    s = 0  
    for i in range(1,n+1):  
        s = s + i  
    return s  
print my_sum(100)
```

实用python编程-2017下

## 循环控制：while循环

while 条件：

    循环语句

在jupyter notebook中观察以下代码的运行结果

```
i = 0
```

```
while i < 10:
```

```
    print i
```

```
    i = i + 1
```

没有最后一行会发生什么？

实用python编程-2017下

## 循环控制：while循环的例子

- 满足n的阶乘大于1000的最小整数n怎么求？

```
def my_factorial (max_fac):
```

```
    fac = 1
```

```
    i = 0
```

```
    while fac <= max_fac:
```

```
        i = i + 1
```

```
        fac = fac * i
```

```
    return i
```

```
print my_factorial(1000)
```

实用python编程-2017下