

Python 基本语法

常量

常量是编程中固定不变的量，包括数字、字符串、布尔值

使用 `type(*)` 查看类型

变量

Python中变量不需要声明类型，命名方式同其他语言，由字符、数字、下划线组成，第一个字符不可以是数字

运算符

算数运算符：+ - * / // % **

逻辑运算符：> < >= <= == != and or not

位运算符：>> << & | ^ ~

函数

函数的定义方式：

```
def 函数名(形参):  
    statement  
    return *
```

python的函数：

- 形参不需要声明类型
- 没有大括号，只有冒号，依靠缩进体现函数内部
- 语句后没有分号

局部变量和全局变量：

函数内部定义的变量是局部变量

函数内部要修改全局变量时需要声明

```
# 定义全局变量  
counter = 0  
  
def increment_counter():  
    global counter # 声明使用全局变量  
    counter += 1 # 修改全局变量  
  
increment_counter()  
print(counter) # 输出: 1
```

Python 控制流

1. if-else 语句

```
if ***:  
    statement  
else:  
    statement
```

2. while 语句

```
while ***:  
    statement
```

3. for-in 语句

```
for * in ***:  
    statement
```

4. break 语句

5. continue 语句

Python 数据结构

列表、元组、字典、集合，见具体笔记

Python 面向对象编程

对象的三个特性：

1. 每个对象都有一个独特的名字以区别于其他对象。
2. 有属性来描述它的某些特征。
3. 有一组操作，每个操作决定对象的一种行为。

类：类是一组具有相同数据和操作的对象的模板集合

面向对象编程的几个基本特征：抽象、封装、继承、多态

如何定义类：

```
class 类名():
    # 初始化函数
    def __init__(self,*,*):
        statement
    ## 定义类函数
    def 函数名(self,*,*):
        statement
    # 定义类变量/常量
```

例如：

```
class student():
    def __init__(self, name, Math_score, Chinese_score):
        self.name = name
        self.Math_score = Math_score
        self.Chinese_score = Chinese_score

    ## repr 函数用于定义对象被输出时的输出结果
    def __repr__(self):
        return str((self.name, self.Math_score, self.Chinese_score))

    def change_score(self, course_name, score):
        if course_name == 'Math':
            self.Math_score = score
        elif course_name == 'Chinese':
            self.Chinese_score = score
        else:
            print(course_name, " course is still not in current system")

    def print_name(self,):
        print(self.name)

name = 'Undefined'
Math_score = None
Chinese_score = None
```

Python 文件与模块

文件

1. open() 函数

`open(file,mode)`: `file` 是读写文件的路径, `mode` 是读取文件的模式, 常用模式有以下几种:

- `r`: 以字符串形式读取文件
- `rb`: 以二进制形式读取文件

在 `r` 与 `rb` 模式下:

- `read()`: 读取整个文件
- `readline()`: 读取文件的一行
- `readlines()`: 读取文件的所有行
- `w`: 写入文件

- `a`: 追加写入文件

在 `w` 与 `a` 模式下:

- `write()`
- `writelines()`

2. `close()` 函数

模块

```
from model_name import xxx
```