

编程语言

# 01 语言结构与数据

# 教学内容与要求

课次	教学内容	教学要求
01 编程语言：语言结构与数据	1.语言要素规范	1. 能编写代码并运行代码；
	2.python解释器与开发工具	
	3.数据定义	
	4.数据运算与运算符	

语言是用来做运算的，也用于软件工程，掌握一门语言，首先要掌握语言规范，其次是语法规则。语言规范是一个程序是怎么组织的规范，包含：目录，文件，段，语句，字的规范。语法规则是一个语句是怎么组织的规范，包含：数据定义，数据访问，数据计算，数据存储的规范。

本节课程首先讲清楚python程序的目录，文件，段，语句，字的规范，然后以语句为基础讲述数据定义与数据运算的语法。为了能体验语法规则能很好的达到计算目的，我们还要讲述python工具的使用。

我们讲述python语言的最终目的不是为了掌握python语言，而是需要具有利用python语言编写各种计算程序的目的。通过本次课程，能学会编写python代码，并运行python代码。

# 1. 语言要素规范 - 包与模块

Python程序的源代码都是由文本文件构成，并且可以用不同目录分类存储。

## 目录

在Python语言规范中，目录称为**包 (Package)**，包的分隔使用点“.”，多级目录构成的路径，在Python中称为**包路径**。

## 文件

在Python语言规范中，源代码文件称为模块Module，模块文件名使用.py扩展名。

说明：包与模块的命名遵循Python的命名规范。

- 开头：\_，字母（字母包含扩展字母：汉字）
- 组成：\_，字母，数字

```
E:\01AI\06YOLO\02CODES\YOLOV4
├── app
│   ├── test.py
│   └── train.py
├── data
│   ├── data.py
│   ├── imglist.py
│   └── mosaic.py
├── __pycache__
│   ├── data.cpython-36.pyc
│   ├── imglist.cpython-36.pyc
│   └── mosaic.cpython-36.pyc
├── model
│   ├── configs.py
│   └── yolov5.py
├── net
│   ├── yolov4-pacsp-mish.cfg
│   ├── yolov4-pacsp-s-mish.cfg
│   ├── yolov4-pacsp-s.cfg
│   ├── yolov4-pacsp-x-mish.cfg
│   ├── yolov4-pacsp-x.cfg
│   ├── yolov4-pacsp.cfg
│   ├── yolov4-paspp.cfg
│   └── yolov4-tiny.cfg
├── __pycache__
│   └── configs.cpython-36.pyc
```

# 1. 语言要素规范 - 段, 句, 字

Python程序的源代码文件内容由段构成, 段由语句构成, 语句由字构成。

## 段

缩格对齐的就是段, C,Java,SQL等语言使用的是{}或者其他分界标记。  
段可以嵌套。

## 句

在Python语言中句的分界使用的是换行符。一行表示一个语句, 也可以使用分号。

## 字

在Python语言中字使用空格, 制表符, 运算符等作为分界。主要还是空格。

```
41 # \r\n\s : whitespace characters. Will not be handled.
42 # re_han_default = re.compile("([\u4E00-\u9FD5a-zA-Z0-9+#&\._%]+)", re.U)
43 # Adding "-" symbol in re_han_default
44 re_han_default = re.compile("([\u4E00-\u9FD5a-zA-Z0-9+#&\._%\ -]+)", re.U)
45
46 re_skip_default = re.compile("(\r\n|\s)", re.U)
47
48
49 def setLogLevel(log_level):
50     default_logger.setLevel(log_level)
51
52
53 class Tokenizer(object) (parameter) dictionary: Any | None
54
55     def __init__(self, dictionary=DEFAULT_DICT):
56         self.lock = threading.RLock()
57         if dictionary == DEFAULT_DICT:
58             self.dictionary = dictionary
59         else:
60             self.dictionary = _get_abs_path(dictionary)
61         self.FREQ = {}
62         self.total = 0
63         self.user_word_tag_tab = {}
64         self.initialized = False
65         self.tmp_dir = None
66         self.cache_file = None
```

# 1. 语言要素规范 - 关键字与标识字

Python语言中的字是最小单位，在源代码中使用的字一般两种情况：（1）编译器内置的字，也称**关键字**。（2）用户命名的字，也称**标识字**。Python程序就是由关键字与标识字构成。

## 关键字

关键字也称保留字，是编译器使用的字。关键字在编译器文档中都有说明（见左）。

## 标识字

标识字是程序员命名的。其命名遵循Python的命名规则。

False	await	else	import	pass
None	break	except	in	raise
True	class	finally	is	return
and	continue	for	lambda	try
as	def	from	nonlocal	while
assert	del	global	not	with
async	elif	if	or	yield

说明：Python语言还有一种特殊的字：字面字，主要用来表示各种数据，比如：数字、字符等。

# 1. 语言要素规范 - 字面字

字面字主要用来直接表示数据，常见的数据是整数，小数、字符与逻辑值。逻辑值在Python中是关键字True与False。

## 整数字面字

整数字面字主要有四种方式：

(1) 二进制：0b11 (2) 八进制：0o77 (3) 十进制：123 (4) 十六进制：0xAB  
可以使用+-前缀表示整数的正负

## 小数字面字

小数：23.56

可以使用+-前缀表示整数的正负

## 字符字面字

字符：“ 字符” , ‘ 字符’

字符串：‘ 字符串’ , “ 字符串” ; ‘ ‘ 字符串’ ‘ ‘

字符串前面可以使用前缀表示特殊的字符串：u (unicode编码字符串)、r (原生字符串, 取消转义符)、b (表示使用字节码表示字符串)、f (格式化字符串)



# 1. 语言要素规范 - 句

Python执行的基本单位就是句。一般语言中的句分成三类：

- (1) 特殊语句：比如注释语句；
- (2) 数据申明语句
- (3) 功能语句：数据运算语句；

## 注释语句

#开头的语句表示注释： # 注释；python没有多行注释。

注释语句中有一个特殊的语句，用来指明源代码的编码：#encoding=编码

## 声明语句

因为Python中不需要显式声明数据类型，所以声明语句不常用，但是在语法上是存在的。  
比如：a:int

## 功能语句

输出语句：print( "Hello Python" )

## 空语句

```
pass
```

## 包引入语句

```
import  
import as  
from import
```



## 2. python解释器与开发工具

Python开发的工具很多，其中主要是解释器：python，安装工具pip。

### 解释器工具

### python --help 获取帮助

python 源代码

python -m 模块

Python的源代码与模块搜索机制

- (1) 源代码：window的通用搜索机制（相对/绝对路径）。
- (2) 模块：使用PYTHONPATH环境变量。

python官网：

<https://pypi.org/project/pip/>

pip官网：

<https://docs.python.org/3/>

### 安装工具

### pip --help 获取帮助

pip install 安装包名

pip install 安装包名 -i 代理镜像 （可以使用代理镜像加速，在百度使用“pip国内镜像源”查询）

推荐的国内镜像站

[个人推荐清华大学pypi镜像站(<https://mirrors.tuna.tsinghua.edu.cn/help/pypi/>),

清华大学：<https://pypi.tuna.tsinghua.edu.cn/simple>

阿里云：<http://mirrors.aliyun.com/pypi/simple/>

豆瓣：<http://pypi.douban.com/simple/>



## 3. 数据定义

数据定义一般两个步骤：

1. 数据声明（申请内存空间）。
2. 数据赋值（存储数据）

### 数据声明

在很多语言中，数据声明是必须显式声明，只有声明后才能使用，声明空间必须指定空间大小（数据类型）。Python语言中没有显式的数据类型，所有空间声明不是必须的，但是语法是存在的。

变量名：类型

### 数据赋值

数据赋值使用赋值运算符：=

Python在赋值的同时，可以自动缺省声明数据，所有变量赋值前不用声明数据。

注意：数据赋值还可以使用键盘的输入赋值：变量 = input(“提示字符串”)



## 4. 数据运算与运算符

数据运算都是通过运算符实现，基本上所有语言中的数据运算包含：

- (1) 算术运算：+ (加) - (减) \* (乘) / (除) // (整除) \*\* (幂) %(求余)
- (2) 关系运算：== (相等) != (不等) <= (小于等于) >= (大于等于) < (小于) > (大于)
- (3) 逻辑运算：and (且) or (或) not (否)
- (4) 位运算：&(且)| (或) ^ (异或) ~ (取反) >> (右移) << (左移)
- (5) 复合运算符：+=, -=, \*=, /=, //=, \*\*=, %=, &=, |=, ^=, <<=, >>=

运算的几个关键点：

- (1) 优先级，使用 () 强制指定优先级。
- (2) 运算符参与运算的数据类型要求，比如：逻辑运算只能对逻辑数据。
- (3) 运算后，数据类型改变规则：变大。

## 答疑联系

13338629985(微信/钉钉)