

python的介绍

1. python是解释性语言
2. python是面向对象的编程语言
3. python是比较容易学习的语言

python安装和环境搭建

1. 下载:<https://www.python.org/downloads/windows/>
2. 看安装文档

PyCharm的安装和配置

- 1 可以写代码并且可以运行代码的软件,我们一般叫IDE
- 2 #下面的代码表示打印我的名字
- 3 `print("刘志山")`
- 4
- 5 常用的快捷方式:
- 6 注释行 `Ctrl + /`
- 7 在下方新建行 `Shift + Enter`
- 8 上下移动代码行 `Alt + Shift + ↑ ↓`
- 9 查看参数列表信息 `Ctrl + p`
- 10

编译与解释(重点)

1. 编译:有个单独把代码编译成可执行程序的过程
2. 解释:直接运行成机器语言,然后显示出来
- 3

交互式编程

- 1 交互式编程:
- 2 1.使用python解释器进行编写代码
- 3 2.即用即得
- 4 如:

```
5 Python 3.6.5 (v3.6.5:f59c0932b4, Mar 28 2018, 17:00:18) [MSC v.1900 64 bit (AMD64)]
  on win32
6 Type "copyright", "credits" or "license()" for more information.
7 >>> print("hello")
8 hello
9 >>>
10
11
12
13 运行脚本
14 1.python 脚本名.py
15 2.IDE运行,如:pycharm
16
17 案例:交互式编程,打印你的姓名
18
19 案例:创建一个文本文档.txt,然后写上python代码,输入'明天会更好,加油!',然后后缀改为.py,用python解
  释器运行.
20
```

编码(重点)

```
1 在 windows 下可以不写第一行注释
2     1.#!/usr/bin/env python    #centos下,使用/usr/bin/env目录下的python解释器运行脚本
3     2.# coding:utf-8          #windows下,指定文件编码格式
4     3.# -*- coding:utf-8 -*-   #windows下,推荐,因为支持的编辑器更多
5
6 编码方式对中文有影响,如果gbk编码的中文放到utf8编码下,可能会出现乱码
7 大部分除了微软(gbk编码方式),其他的平台用的都是utf8编码方式
8
```

标识符

```
1 标识符是用来给python语言中一些变量,函数,类名的命名的字符
2 取名字有规则:
3 1.只能用字母,数字,下划线
4 2.第一个字符不能是数字
5 3.对大小写敏感
6 4.不能使用关键字(保留字)
```

保留字

```
1 查看保留字:
2 >>>import keyword
3 >>>keyword.kwlist
4 ['False', 'None', 'True', 'and', 'as', 'assert', 'break', 'class', 'continue',
  'def', 'del', 'elif', 'else', 'except', 'finally', 'for', 'from', 'global', 'if',
  'import', 'in', 'is', 'lambda', 'nonlocal', 'not', 'or', 'pass', 'raise', 'return',
  'try', 'while', 'with', 'yield']
```

注释和缩进(重点)

```
1 注释的作用:就是起到解释说明的作用
2 单行注释使用#
3 多行注释使用三个单引号对,或三个双引号对
4 '''
5 sdfsaf
6 sfasf
7 fdsaf
8 '''
9
10 """
11 fsfsadf
12 sfdafsdaf
13 asfdasf
14 """
15
16 python最具特色的就是使用缩进来表示代码块,不需要使用大括号({}).
17 缩进的空格数是可变的,但是同一个代码块的语句必须包含相同的缩进空格数
18 缩进最好使用四个空格.而且要注意缩进要一致,使用空格就全都用空格,使用Tab就都使用Tab
19
20 如:
21 if True:
22     print("hello")
23     print("xxxxx")
24 else:
25     print("false")
26
27 以下代码最后一行语句缩进数的空格数不一致,会导致运行错误
28 错误:
29 if False:
30     print("hello")
31 else:
32     print("aaa") #这里报错
33     print("false")
```

多行和空行

```
1  多行:
2  当一行代码写不下时,可以使用多行
3
4
5  空行:
6  一般是函数之间或控制语句之间会有空行,这样便于阅读代码
7
8  空行与代码缩进不同,空行并不是Python语法的一部分。书写时不插入空行,Python解释器运行也不会出错。但是空行的作用在于分隔两段不同功能或含义的代码,便于日后代码的维护或重构
9      注意:空行也是程序代码的一部分
10
11
```

输入输出(重点)

```
1  1.输出:
2  print("sdfsdf")
3  print输出是默认换行
4  消除默认换行:
5  print("xxxx",end='')
6  如果写一个print(),希望打印两行或多行
7  print("我是\n两行")
8
9
10 2.输入:
11 msg=input("提示信息")
12 会阻塞,等待输入
13 得到的数据都是字符串
14
15
16 案例:
17 1.从屏幕接收两个整数,并打印其加减的结果
18 a=input("请输入第一个数字:")#这里a存储的信息是字符串
19 b=input("请输入第二个数字:")#这里b存储的信息是字符串
20 #把字符串转换为整型,int(字符串)
21 s=int(a)+int(b)
22 s1=int(a)-int(b)
23 print(s)
24 print(s1)
25
26 2.一个print显示以下内容:
27 姓名:葫芦娃
28 年龄:12
29 性别:男
30 地址:葫芦山
31 print("姓名:葫芦娃\n年龄:12\n性别:男\n地址:葫芦山")
32 3,使用两个print打印下面一句话
33 '我们都是葫芦娃,葫芦娃啊葫芦娃'
34 print("我们都是",end='')
35 print("葫芦娃,葫芦娃啊葫芦娃")
```

占位符

```
1  顾名思义, 占位符就是先占住一个固定的位置, 等着你再往里面添加内容的符号。格式占位符(%)是在Python语言
2  中格式输入函数, 如print( )等函数中使用。其意义就是起到格式占位的意思, 表示在该位置有输入或者输出
3  %d, 整数, 比如1, 2, 3, 4等就是整数, 没有小数点
4  %f, 浮点, 比如20.22, 30.33, 有小数点的, 就叫浮点数
5  %s, 字符串, 比如"我是字符串", '我也是字符串'
6
7
8  #下面这句代码是从终端获取用户输入的数字字符串, 然后转换为整型, 再赋值给a
9  # a=int(input("请输入你的数字:"))
10
11 #当代码运行后, 用户在终端输入字符串10之后, a存储到了10这个整型
12 #然后print这里, 把%d替换为10, 然后打印出来
13 # print("你输入的数字是:%d"%a)
14
15
16 # a=float(input("请输入你的有小数的数字:"))
17 # #0.2表示只保留小数点后两位
18 # print("你输入的有小数的是:%0.2f"%a)
19
20 a=input("请输入你的信息:")
21 print("你输入的信息是:%s"%a)
22
23 案例: 定义两个变量, 一个是整型, 一个是字符串类型, 赋值你的年龄和姓名, 用占位符打印出来
24 a=int(input("请输入你的年龄:"))
25 b=input("请输入你的姓名:")
26 print("你的姓名是:%s, 你的年龄是:%d"%(b, a))
27
28 #案例: 已知A网站苹果和橘子两种水果单价(具体如下), 用户根据自己的需求输入斤数, 系统计算总价并打印结果
29 # 水果单价
30 apple_price = 6.6
31 orange_price = 5
32 apple_price = 6.6
33 orange_price = 5
34 a=float(input("请输入苹果的斤数:"))
35 o=float(input("请输入橘子的斤数:"))
36 s=a*apple_price+o*orange_price
37
38 print("应该付钱:%0.2f"%s)
39
40 #案例: 用户登录系统: 用户输入用户名和密码, 并控制台格式化输出用户输入的用户名和密码。
41 name=input("请输入你的用户名:")
42 password=input("请输入你的密码:")
43 print("你的用户名是:%s, 你的密码是:%s"%(name, password))
44
```

help函数

1. 功能: 可以查看别的内置函数的用法
2. 使用: help(内置函数的名字)

变量(重点)

1. 什么是变量?
变量是可以存储信息
变量是从内存中获取一块位置, 用来存储信息, 这块位置的信息可以改变, 就叫变量
 2. 定义变量的规则:
变量使用前都必须有值
可以同时定义多个变量, 并给相同的值
可以同时定义多个变量, 并给不同的值
- 案例:
1. 给三个变量同时赋值20, 并打印出来
- ```
a=b=c=20
print(a)
print(b)
print(c)
```
2. 把以下数据赋值同时赋值到变量中, 并打印
- ```
'公司:华为', 8, '地址:深圳'
name, age, adds='公司:华为', 8, '地址:深圳'
print(name)
print(age)
print(adds)
```
- 定义变量和给变量赋值的原理
- 一般说变量的类型是说变量里存储的数据是什么类型
- ```
a=10
b=20
#打印变量存储的地址, 这地址是执行内存空间的, 这个内存空间就是存储的数据
print(id(a))
print(id(b))
```
- 案例:
- 定义两个变量a和c, 分别赋值10, 20, 然后打印它们的地址, 之后把a赋值给c, 然后打印a的地址
- 变量的类型:
1. 用type(变量名)可以查出变量里存储的数据类型
- ```
a=10
print(type(a))
a="hello"
print(type(a))
a=20.22
print(type(a))
```

