

Python概述

解释形语言, 交互式语言, 面向对象语言, 初学者的语言

特点: 易于学习, 易于阅读, 易于维护, 一个广泛的标准库, 交互模式, 可移植, 可扩展, 数据库, GUI编程, 可嵌入
缺点: 运行速度慢, 代码不能加密

数据存储

- 1.内存是与CPU沟通的桥梁
- 2.八位一个字节, 8bit==1byte,1024byte==1k
- 3.原码, 反码, 补码

数据类型

- 整数, 浮点数, 复数
- 字符串string: python中无字符概念
- 布尔值Boolean, 空值None, 列表List, 元组tuple, 字典dict, 集合set

标识符

- 是一串字符串(注意: 字符串未必是标识符)
- 1)只能由字母, 数字, 下划线组成
- 2)开头不能是数字
- 3)不能是python的关键字
- 4)区分大小写
- 5)见名知意
- 6)遵循驼峰原则 *首单词是正常的, 从第二个单词开始首字母大写

作用: 给变量, 函数等命名的 (注意: 在python3中, 非ASCII标识符也能识别)

变量和常量

- 变量
 - 1>概述: 程序可操作的存储空间名称, 程序运行期间能改变的数据, 每个变量都能特定的类型
 - 2>作用: 将不同类型的数据存储到内存
 - 3>定义变量: 变量名=初始值 (为了确定变量的类型)
 - 4>数据的存储: 变量名=数据值 (注意: 变量在使用之前必须定义, 否则会报错) input输入的是字符串, 应该使用int来转换为数字 int(input("请输入一个数字"))
 - 5>删除变量: del 变量名 (删除后变量无法引用)
 - 6>查看变量的类型: type(变量名)
 - 7>查看变量地址: id (变量名)
- 常量
 - 程序运行期间不能改变的数据

数学功能

- abs(x) #求绝对值
- max(a,b) #求最大值
- pow(2,5) #求2的5次方
- round(3.3432,2) #求小数点后两位, 四舍五入
- sqrt(a) #开方
- math.ceil(a) #向上取整
- math.floor(a) #向下取整
- math.modf(a) #返回整数部分与小数部分
- random.choice([1,2,3,4]) #取随机数
- random.choice(range(5)) #生成一个5以内的数
- random.randrange([start,stop[,step]]) #生成一个指定范围内的数, step为递增基数
- random.random() #随机生成[0,1]之间的浮点数
- random.shuffle(list) #将序列的所有元素随机排序
- random.uniform(a,b) #随机生成一个实数
- sin,cos,tan #三角函数

import math

常用功能

- eval(str)#功能: 将字符串当成有效的表达式来求值并返回计算结果
- len(str)#返回字符串的长度
- lower(str)#转换字符串中大写字母为小写字母
- upper(str)#转换字符串中小写字母为大写字母
- swapcase()#将字符串中的大写变为小写, 小写变为大写
- capitalize()#首字母大写, 其他小写
- title()#每个单词的首字母大写
- center(width[,fillchar])#返回一个指定宽度的居中字符串, fillchar为填充的字符
- ljust(width[,fillchar])#返回一个指定宽度的左对齐字符串, fillchar为填充字符, 默认空格填充
- rjust(width[,fillchar])#返回一个指定宽度的右对齐字符串, fillchar为填充字符, 默认空格填充
- zfill(width)#返回一个长度为width的字符串, 原字符串右对齐, 前面补零
- count(str[start:end])#返回字符串中str出现的次数, 可以指定一个范围, 默认从头到尾
- find(str[start:end])#检测字符串中是否包含在字符串中, 可以指定范围, 默认从头到尾
- index(str,start=0,end=len(str))#和find一样, 只不过str不存在时返回一个异常
- lstrip(str)#截掉字符串左侧指定的字符, 默认为空格
- rstrip(str)#截掉字符串右侧指定的字符, 默认为空格
- strip(str)#截取左右两端指定的字符串
- ord 参数是字符, 返回值是是精致的整数, 代表点是该字在ASCII中的文字 ord('a')
- chr 参数可以是十进制, 也可以是16进制的形式的数字, 返回值是ASCII字符 chr(str)
- 字符串比较: 从第一个字符开始比较, 谁的ASCII值大谁就大, 如果相等会比较下一个字符的ASCII值大小, 那么谁的谁就大

运算符表达式

- 算术运算符: + - * /(除) %(取模) **(求幂) //(取整)
- 由变量, 常量和运算符组成的式子
 - 算术运算表达式
 - 赋值运算符和赋值运算符表达式
 - 变量 = 表达式
 - 复合运算符
 - + = - = * =
 - 按位运算符是把数字看做二进制数来进行计算
 - 位运算符
 - & 按位与运算(符号为1时则为1)
 - | 按位或运算(两个二进制位有一个位1时, 结果为1)
 - ^ 按位异或运算(相同则0, 不同为1)
 - ~ 按位取反运算符 (每个二进制数数据位取反, 1变0,0变1)
 - << 左移运算符 (各二进制位全部左移若干位, 有<<右侧的数字决定, 高位丢失, 低位补0)
 - >> 右移运算符 (各二进制位全部右移若干位)
 - 关系运算符和关系运算符表达式
 - = == != > < >= <=
 - 表达式1 and 表达式2
 - 表达式1 or 表达式2
 - not 表达式
 - 逻辑运算符
 - in: 如果在指定的序列中找到了值返回True, 否则返回False
 - not in: 如果在指定的序列中没有找到值返回True, 否则返回False
 - is: 判断两个标识符是不是应用同一个对象
 - is not: 判断两个标识符是不是引用不同的对象
 - 身份运算符
 - **求幂
 - ~(取反码)
 - + -
 - * / % //
 - > < <= >=
 - & |
 - ^
 - <= < > >=
 - = !=
 - %= += -= //
 - is is not
 - in not in
 - not or and
 - 运算符优先级

字符串: 字符串不可变

- 是以单引号和双引号括起来的任意文本
- 中文在内存中占2-4个字节, 英文在内存中占1个字节(8位一个字节)

占位符

- %d 可以代替代替整数
- %s 可以代替字符串
- %f 可以代替浮点数
- %.3f 表示保留小数点后三位, 但会四舍五入

转义字符 \

- \n 占用一个字节
- \t 制表符 默认四个空格, 或八个空格
- r #python允许用r表示内部的字符串默认不转义