

第 1 章 基础知识

1. 安装

安装就不多讲，而且很多 **linux** 系统都自带 **python**。

对于初学者来说，**IDE** 可以选 **IDLE**，标准的 **IDE** 对于初学者已经够了。

2. 交互式解释器

我们更多的是通过文件来执行。

第一个程序：**print "hello world"**

3. 算法是什么

4. 数字和表达式

交互式的 **python** 解释器可以当计算器使用。

整数除法问题：两个整数相除-->截断，只保留整数

处理方法：使用浮点数

使用普通除法 **from __future__ import division**

整除：**//**

5. 变量

变量名可以包括字母，数字和下划线，变量不能以数字开头

6. 语句

表达式和语句的区别：

2*2: 某事

print 2*2: 做某事 语句改变了事物 这里改变了显示器的显示的内容

7. 获取用户输入

input 函数

8. 函数

内建函数 **pow abs round floor(math)**

9. 模块

import math

math.floor(32.9) 小于等于的最大整数

math.ceil 大于等于的最小整数

导入模块的另一种方法：

from math import sqrt （先确保多个模块函数不重名） 可直接使用 **sqrt** 函数

nan: 特殊的值 表示 非数值 比如：**sqrt(-1)** （复数可以计算此表达式 需 **cmath** 模块）

10. 保存并执行程序

pound bang: `#!/usr/bin/env python`

11. 字符串

灵活使用单引号和双引号

str, repr 和反引号的区别

str: 把值转换成合理形式的字符串,以便用户理解

str 是一种类型

repr: 创建一个字符串,它以合法的 **python** 表达式的形式来表示

repr 只是函数

反引号功能相同(**python3** 没有反引号)

input 和 **raw_input** 的区别

input: **input** 假设用户输入的都是合法的 **python** 表达式

raw_input: **raw_input** 把所有的输入当做原始数据,放入字符串尽量 **raw_input**

长字符串,原始字符串和 **Unicode**

长字符串: `'''`

原始字符串: `r"c:\xxg"` 不能在原始字符串末尾输入反斜线

处理这种问题: `print r'd:\xxx' '\\'` 可以实现末尾\的输出

Unicode 字符串: `u'hello,world'`

python3 全部都是 **Unicode** 字符串