# 第二天讲课大纲

1. Python安装，区分2.X , 3.X
2. 使用IDLE编码
3. 弱类型语言
4. 变量及命名规则
   1. 第一个字符必须是字母表中字母或下划线 \_ 。
   2. 标识符的其他的部分由字母、数字和下划线组成。
   3. 标识符对大小写敏感。
5. 数值类型
   1. 整型
   2. 浮点型
   3. 复数
6. 字符串
   1. 字符串及转义字符

|  |  |
| --- | --- |
| **转义字符** | **描述** |
| \(在行尾时) | 续行符 |
| \\ | 反斜杠符号 |
| \' | 单引号 |
| \" | 双引号 |
| \a | 响铃 |
| \b | 退格(Backspace) |
| \000 | 空 |
| \n | 换行 |
| \v | 纵向制表符 |
| \t | 横向制表符 |
| \r | 回车 |
| \f | 换页 |
| \oyy | 八进制数，yy代表的字符，例如：\o12代表换行 |
| \xyy | 十六进制数，yy代表的字符，例如：\x0a代表换行 |
| \other | 其它的字符以普通格式输出 |

* 1. 更新单个字符
  2. 字符串拼接，重复输出字符串
  3. 使用input()进行字符串输入
  4. 格式化字符串

整数输出

%o —— oct 八进制

%d —— dec 十进制

%x —— hex 十六进制

>>> print('%o' % 20)

24

>>> print('%d' % 20)

20

>>> print('%x' % 20)

14

浮点型输出

%f ——保留小数点后面六位有效数字

　　%.3f，保留3位小数位

%e ——保留小数点后面六位有效数字，指数形式输出

　　%.3e，保留3位小数位，使用科学计数法

%g ——在保证六位有效数字的前提下，使用小数方式，否则使用科学计数法

　　%.3g，保留3位有效数字，使用小数或科学计数法

>>> print('%f' % 1.11) # 默认保留6位小数

1.110000

>>> print('%.1f' % 1.11) # 取1位小数

1.1

>>> print('%e' % 1.11) # 默认6位小数，用科学计数法

1.110000e+00

>>> print('%.3e' % 1.11) # 取3位小数，用科学计数法

1.110e+00

>>> print('%g' % 1111.1111) # 默认6位有效数字

1111.11

>>> print('%.7g' % 1111.1111) # 取7位有效数字

1111.111

>>> print('%.2g' % 1111.1111) # 取2位有效数字，自动转换为科学计数法

1.1e+03

字符串输出

%s

%10s——右对齐，占位符10位

%-10s——左对齐，占位符10位

%.2s——截取2位字符串

%10.2s——10位占位符，截取两位字符串

>>> print('%s' % 'hello world') # 字符串输出

hello world

>>> print('%20s' % 'hello world') # 右对齐，取20位，不够则补位

hello world

>>> print('%-20s' % 'hello world') # 左对齐，取20位，不够则补位

hello world

>>> print('%.2s' % 'hello world') # 取2位

he

>>> print('%10.2s' % 'hello world') # 右对齐，取2位

he

>>> print('%-10.2s' % 'hello world') # 左对齐，取2位

He

字符串位置匹配

>>> print('{} {}'.format('hello','world')) # 不带字段

hello world

>>> print('{0} {1}'.format('hello','world')) # 带数字编号

hello world

>>> print('{0} {1} {0}'.format('hello','world')) # 打乱顺序

hello world hello

>>> print('{1} {1} {0}'.format('hello','world'))

world world hello

>>> print('{a} {tom} {a}'.format(tom='hello',a='world')) # 带关键字

world hello world

* 1. 字符串截取：变量[头下标:尾下标:步长]
  2. 原始字符串：print（r“ok”）
  3. in / not in

1. bool类型
2. 运算符

|  |  |  |
| --- | --- | --- |
| **运算符** | **描述** | **实例** |
| + | 加 - 两个对象相加 | a + b 输出结果 31 |
| - | 减 - 得到负数或是一个数减去另一个数 | a - b 输出结果 -11 |
| \* | 乘 - 两个数相乘或是返回一个被重复若干次的字符串 | a \* b 输出结果 210 |
| / | 除 - x 除以 y | b / a 输出结果 2.1 |
| % | 取模 - 返回除法的余数 | b % a 输出结果 1 |
| \*\* | 幂 - 返回x的y次幂 | a\*\*b 为10的21次方 |
| // | 取整除 - 向下取接近除数的整数 | >>> 9//2  4  >>> -9//2  -5 |

|  |  |  |
| --- | --- | --- |
| **运算符** | **描述** | **实例** |
| == | 等于 - 比较对象是否相等 | (a == b) 返回 False。 |
| != | 不等于 - 比较两个对象是否不相等 | (a != b) 返回 True。 |
| > | 大于 - 返回x是否大于y | (a > b) 返回 False。 |
| < | 小于 - 返回x是否小于y。所有比较运算符返回1表示真，返回0表示假。这分别与特殊的变量True和False等价。注意，这些变量名的大写。 | (a < b) 返回 True。 |
| >= | 大于等于 - 返回x是否大于等于y。 | (a >= b) 返回 False。 |
| <= | 小于等于 - 返回x是否小于等于y。 | (a <= b) 返回 True。 |

|  |  |  |
| --- | --- | --- |
| **运算符** | **描述** | **实例** |
| = | 简单的赋值运算符 | c = a + b 将 a + b 的运算结果赋值为 c |
| += | 加法赋值运算符 | c += a 等效于 c = c + a |
| -= | 减法赋值运算符 | c -= a 等效于 c = c - a |
| \*= | 乘法赋值运算符 | c \*= a 等效于 c = c \* a |
| /= | 除法赋值运算符 | c /= a 等效于 c = c / a |
| %= | 取模赋值运算符 | c %= a 等效于 c = c % a |
| \*\*= | 幂赋值运算符 | c \*\*= a 等效于 c = c \*\* a |
| //= | 取整除赋值运算符 | c //= a 等效于 c = c // a |

1. 列表
   1. 定义
   2. 使用
   3. 截取
   4. 拼接
   5. in运算，len()，max()， min()
2. 元祖，同列表，但不可变
3. 字典
4. If分支语句
5. while循环
6. for - in循环
7. pass语句