

# Python 基本命令与操作

田雅夫

July 21, 2014

1. `help()` 查看一个对象的帮助文档
2. `type()` 查看变量/对象/一切事物的类型
3. `global` 在函数中使用外部变量的话用, 例: `global a`
4. `def` 定义一个函数, 注意冒号, 缩进
5. `print` 输出
6. `input()` 输入, 接受一个字符串参数
7. `and or not` 逻辑与, 逻辑或, 逻辑非
8. `int(),str()` 强制类型转换
9. `if()`: 判断语句, 注意冒号, 缩进
10. `else`: 流程控制, 注意冒号, 缩进
11. `elif()`: 流程控制, 代替 `switch`
12. `while()`: 循环, 注意冒号, 缩进
13. `for i in range(n)`: 产生一个 `n` 次的循环, 循环中 `i` 取值为 0 到 `n-1`
14. `range()` 产生列表, 接受一个, 两个和三个参数
15. `list.append()` 在列表后面附加元素
16. `list.pop()` 删除指定下标的元素, 返回被删除元素
17. `list.remove()` 删除指定元素, 没有返回值
18. `del list[]` 删除指定下标的元素, 没有返回值
19. `for i,j in enumerate(list)`: 或 `for j in list`:  
迭代一个列表,`i` 为列表中每个元素的下标,`j` 为列表中的元素
20. `list.count()` 统计指定元素在列表中的出现次数
21. `list(str)` 字符串转列表
22. `str.join(list)` 列表在给定字符串下结合字符串

- 23. `str.split(list)` 字符串
- 24. `lambda parameters:function` 匿名函数例 `lambda x,y,z:(2*x+y-z)`
- 25. `map(func,iterable)` 对可迭代对象 (列表, 元组, 字典) 中每个元素 `a` 用 `func(a)` 代替.`func` 要求为单参数单返回值函数
- 26. `reduce(func,iterable)` 对可迭代对象中元素顺次使用 `func()` 使得列表缩减为单个元素
- 27. `filter(func,iterable)` `func` 要求返回值只能为 `True` 或 `False`, 返回一个能使 `func()` 返回值为真的对象
- 28. `len(list/str)` 返回列表/字符串长度

- 1. 忘记在 `if` , `elif` , `else` , `for` , `while` , `class`,`def` 声明末尾添加: (导致 “`SyntaxError : invalid syntax`”)
- 2. 3) 错误的使用缩进量。(导致 “`IndentationError: unexpected indent`”、“`IndentationError: unindent does not match any outer indetation level`” 以及 “`IndentationError: expected an indented block`”)
- 3. 在字符串首尾忘记加引号 (导致 “`SyntaxError: EOL while scanning string literal`”)
- 4. 交互模式下忘记添加新行 (导致 “`SyntaxError : invalid syntax`”), 交互模式下每一行以 “`»>`” 开头
- 5. 方法名拼写错误 (导致 “`AttributeError: 'str' object has no attribute 'appppppppend'`”)
- 6. 引用超过 `list` 最大索引 (导致 “`IndexError: list index out of range`”)
- 7. 尝试使用 `Python` 关键字作为变量名 (导致 “`SyntaxError: invalid syntax`”)
- 8. 在定义局部变量前在函数中使用局部变量 (此时有与局部变量同名的全局变量存在) (导致 “`UnboundLocalError: local variable 'foobar' referenced before assignment`”)