

python 基础语法

T1:

变量:

视作英语单词? 可以使用英文字母 (小写)、下划线、数字的组合 (不能单独下划线和数字)

用法就是“=”

一般情况下不加(int)等情况时, 视为字符串

例如:

```
synx=66666  
fuck_you=int(114514)
```

函数:

和变量命名规则相同, 只是前面加个def, 后面加个括号 (里面表示引入的变量) ,括号后面有":" (经常忘awa)

例如:

```
def fuck(you):  
    return 114514  
def refuck(you):  
    return 1919810
```

类:

和以上两个类似, 但是最好是首字母大写, 后面不接括号但是有":"

例如

```
class Animal:  
class Student:
```

规范命名好处:

第一点是可以更容易的区分各个变量, 很清楚的就能分清类, 变量和函数, 从而更容易在后续打码中理清思路。

好像就这些? 感觉想不出来了 (个人平时打码也没有太注意这些.....)

T3:

常见容器:

1.列表: 有序、可变、允许重复元素

```
# 创建
my_list = [1, 2, 3, 'a', 'b']

# 常见操作
my_list.append(4)          # 添加元素
my_list.insert(1, 'x')     # 插入元素
my_list.remove('a')        # 删除元素
my_list.pop()              # 弹出末尾元素
my_list.sort()              # 排序
len(my_list)                # 获取长度
```

2.元组: 有序、不可变、允许重复元素

```
# 创建
my_tuple = (1, 2, 3, 'a')

# 常见操作
my_tuple.count(2)           # 计数
my_tuple.index('a')         # 查找索引
len(my_tuple)                # 获取长度
# 元组不可修改, 无添加删除操作
```

3.字典:键值对、无序、键必须唯一

```
# 创建
my_dict = {'name': 'Alice', 'age': 25}

# 常见操作
my_dict['city'] = 'Beijing' # 添加/修改
my_dict.get('name')         # 获取值
my_dict.keys()               # 所有键
my_dict.values()             # 所有值
my_dict.items()              # 所有键值对
my_dict.pop('age')           # 删除键
```

4.集合: 无序、不重复、元素必须可哈希

```
# 创建
my_set = {1, 2, 3, 4}

# 常见操作
my_set.add(5)                # 添加元素
my_set.remove(2)              # 删除元素
set1 | set2                   # 并集
set1 & set2                   # 交集
set1 - set2                   # 差集
len(my_set)                   # 元素个数
```

5.字符串：有序、不可变、字符序列

```
# 创建
my_str = "Hello World"

# 常见操作
my_str.upper()      # 转大写
my_str.lower()      # 转小写
my_str.split(' ')   # 分割
my_str.replace('H', 'h') # 替换
my_str.find('world') # 查找
len(my_str)         # 长度
```

6.队列：先进先出

```
from collections import deque

# 创建
my_queue = deque()

# 常见操作
my_queue.append('a') # 入队
my_queue.popleft()   # 出队
len(my_queue)        # 队列大小
```