

课程：字典

目标

- 字典的应用场景
- 创建字典的语法
- 字典常见操作
- 字典的循环遍历

一. 字典的应用场景

思考1：如果有多个数据，例如：'Tom', '男', 20，如何快速存储？

答：列表

```
1 list1 = ['Tom', '男', 20]
```

思考2：如何查找到数据'Tom'？

答：查找到下标为0的数据即可。

```
1 list1[0]
```

思考3：如果将来数据顺序发生变化，如下所示，还能用 list1[0] 访问到数据'Tom'吗？。

```
1 list1 = ['男', 20, 'Tom']
```

答：不能，数据'Tom'此时下标为2。

思考4：数据顺序发生变化，每个数据的下标也会随之变化，如何保证数据顺序变化前后能使用同一的标准查找数据呢？

答：字典，字典里面的数据是以键值对形式出现，字典数据和数据顺序没有关系，即字典不支持下标，后期无论数据如何变化，只需要按照对应的键的名字查找数据即可。

二. 创建字典的语法

字典特点：

- 符号为大括号
- 数据为键值对形式出现
- 各个键值对之间用逗号隔开

```
1 # 有数据字典
2 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
3
4 # 空字典
5 dict2 = {}
6
7 dict3 = dict()
```

注意：一般称冒号前面的为键(key)，简称k；冒号后面的为值(value)，简称v。

三. 字典常见操作

3.1 增

写法：字典序列[key] = 值

注意：如果key存在则修改这个key对应的值；如果key不存在则新增此键值对。

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2
3 dict1['name'] = 'Rose'
4 # 结果: {'name': 'Rose', 'age': 20, 'gender': '男'}
5 print(dict1)
6
7 dict1['id'] = 110
8
9 # {'name': 'Rose', 'age': 20, 'gender': '男', 'id': 110}
10 print(dict1)
```

注意：字典为可变类型。

3.2 删

- del() / del：删除字典或删除字典中指定键值对。

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2
3 del dict1['gender']
4 # 结果: {'name': 'Tom', 'age': 20}
5 print(dict1)
```

- clear()：清空字典

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2
3 dict1.clear()
4 print(dict1) # {}
```

3.3 改

写法：字典序列[key] = 值

注意：如果key存在则修改这个key对应的值；如果key不存在则新增此键值对。

3.4 查

3.4.1 key值查找

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 print(dict1['name']) # Tom
3 print(dict1['id']) # 报错
```

如果当前查找的key存在，则返回对应的值；否则则报错。

3.4.2 get()

- 语法

```
1 字典序列.get(key, 默认值)
```

注意：如果当前查找的key不存在则返回第二个参数(默认值)，如果省略第二个参数，则返回None。

- 快速体验

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 print(dict1.get('name')) # Tom
3 print(dict1.get('id', 110)) # 110
4 print(dict1.get('id')) # None
```

3.4.3 keys()

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 print(dict1.keys()) # dict_keys(['name', 'age', 'gender'])
```

3.4.4 values()

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 print(dict1.values()) # dict_values(['Tom', 20, '男'])
```

3.4.5 items()

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 print(dict1.items()) # dict_items([('name', 'Tom'), ('age', 20), ('gender', '男')])
```

四. 字典的循环遍历

4.1 遍历字典的key

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 for key in dict1.keys():
3     print(key)
```

```
C:\Users\黑马程序员\AppData\Local\Programs\Python\Python37\python3.exe
name
age
gender

Process finished with exit code 0
```

4.2 遍历字典的value

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 for value in dict1.values():
3     print(value)
```

```
C:\Users\黑马程序员\AppData\Local\Programs\Python\Python37\python3.exe
Tom
20
男

Process finished with exit code 0
```

4.3 遍历字典的元素

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 for item in dict1.items():
3     print(item)
```

```
C:\Users\黑马程序员\AppData\Local\Programs\Python\Python37\python3.exe
('name', 'Tom')
('age', 20)
('gender', '男')

Process finished with exit code 0
```

4.4 遍历字典的键值对

```
1 dict1 = {'name': 'Tom', 'age': 20, 'gender': '男'}
2 for key, value in dict1.items():
3     print(f'{key} = {value}')
```

```
C:\Users\黑马程序员\AppData\Local\Programs\Python\Python37\python3.exe
name = Tom
age = 20
gender = 男

Process finished with exit code 0
```

五. 总结

- 定义字典

```
1 dict1 = {'name': 'Python', 'age': 30}
2
3 dict2 = {}
4
5 dict3 = dict()
```

- 常见操作
 - 增/改

```
1 字典序列[key] = 值
```

- 查找
 - 字典序列[key]
 - keys()
 - values()
 - items()