

作业 8

在本次作业中，选择 **redis** 非关系型数据库。为了让代码及结果呈现得更加清楚，选择在 **python** 中接入 **redis** 数据库并进行操作。在 **redis** 数据库中有字符串、列表、集合、有序集合和哈希几种数据结构，下面将逐一对几种数据结构进行基础得增、删、改、查得操作。

一、字符串

代码部分：

```
# 字符串中的增删改查
r.set( name: 'name', value: 'Alice')
name = r.get('name')
print(name)
r.set( name: 'name', value: 'Bob')
name = r.get('name')
print(name)
r.delete('name')
name = r.get('name')
print(name)
```

结果部分：

```
b'Alice'
b'Bob'
None
```

二、列表

代码部分：

```
# 列表中的增删改查
r.lpush( name: 'list', *values: 'Alice')
r.rpush( name: 'list', *values: 'Bob')
r.lpush( name: 'list', *values: 'Ann')
lst = r.lrange( name: 'list', start: 0, -1)
lst1 = r.lrange( name: 'list', start: 0, end: 0)
print(lst)
print(lst1)
r.lset( name: 'list', index: 0, value: 'Ken')
lst = r.lrange( name: 'list', start: 0, -1)
print(lst)
r.lpop('list')
r.rpop('list')
lst = r.lrange( name: 'list', start: 0, -1)
print(lst)
r.lrem( name: 'list', count: 1, value: 'Alice')
```

结果部分:

```
[b'Ann', b'Alice', b'Alice']  
[b'Ann']  
[b'Ken', b'Alice', b'Alice']  
[b'Alice', b'Alice', b'Alice']
```

三、集合

代码部分:

```
# 集合中的增删改查  
r.sadd( name: 'fruits', *values: 'apple')  
r.sadd( name: 'fruits', *values: 'banana')  
fruits = r.smembers('fruits')  
print(fruits)  
r.sadd( name: 'fruits', *values: 'orange')  
fruits = r.smembers('fruits')  
print(fruits)  
r.srem( name: 'fruits', *values: 'banana')  
fruits = r.smembers('fruits')  
print(fruits)
```

结果部分:

```
{b'apple', b'banana', b'orange'}  
{b'apple', b'banana', b'orange'}  
{b'apple', b'orange'}
```

四、有序集合

代码部分:

```
# 有序集合中的增删改查  
r.zadd( name: 'scores', mapping: {'Alice': 90, 'Bob': 85})  
top_scores = r.zrevrange( name: 'scores', start: 0, end: 1, withscores=True)  
print(top_scores)  
score_range = r.zrangebyscore( name: 'scores', min: 80, max: 90, withscores=True)  
print(score_range)  
r.zincrby( name: 'scores', amount: 5, value: 'Bob')  
r.zrem( name: 'scores', *values: 'Alice')
```

结果部分:

```
[(b'Alice', 90.0), (b'Bob', 85.0)]  
[(b'Bob', 85.0), (b'Alice', 90.0)]
```

五、哈希

代码部分:

```

# 哈希中的增删改查
r.hset('user:1', 'name', 'Alice')
r.hset('user:1', 'age', 30)
name = r.hget(name: 'user:1', key: 'name') # 获取指定字段的值
print(name)
user_info = r.hgetall('user:1') # 获取所有字段和值
print(user_info)
r.hset('user:1', 'age', 31)
name = r.hget(name: 'user:1', key: 'name')
print(name)
user_info = r.hgetall('user:1')
print(user_info)
r.hdel(name: 'user:1', *keys: 'age')
name = r.hget(name: 'user:1', key: 'name')
print(name)
user_info = r.hgetall('user:1')
print(user_info)

```

结果部分:

```

b'Alice'
{b'name': b'Alice', b'age': b'30'}
b'Alice'
{b'name': b'Alice', b'age': b'31'}
b'Alice'
{b'name': b'Alice'}

```