

# Làm quen với Redis – cài đặt và code minh họa

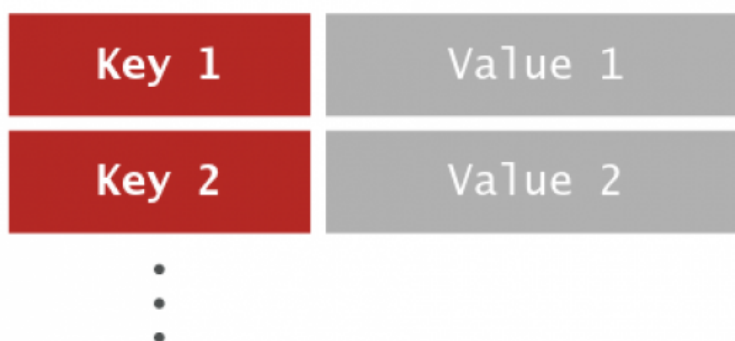
## *Giới thiệu chung*

Redis (REmote DIctionary Server) là một open source (BSD licensed), in-memory **data structure store** (đại ý là cơ chế xử lý dữ liệu trên memory (nhằm tăng tốc) và lưu trữ trong disk như một database, các bạn có thể xem thêm về in-memory database [tại đây](#)). Qua đặc điểm trên thì Redis có khả năng được sử dụng như một database, cache và message broker (cơ chế pub/sub – xem thêm về cơ chế pub/sub [tại đây](#)). Nhờ cung cấp nhiều tính năng như Pipelining (truyền nhiều command cùng lúc), Pub/Sub, Partitioning (cơ chế master slave), ưu điểm về tốc độ và dễ sử dụng, nên hiện tại Redis đang được sử dụng rộng rãi trong các ứng dụng, game với yêu cầu gọn nhẹ và tốc độ xử lý cao.

## *Đặc điểm*

Lưu trữ data dưới dạng key – value, hỗ trợ lưu trữ các dạng dữ liệu:

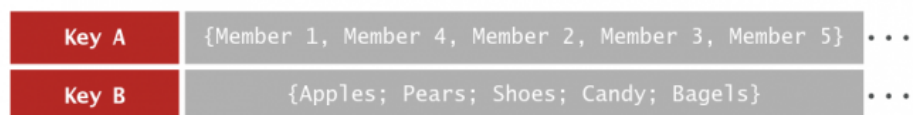
*String:*



*List:*



*Set:*



*Sort Set:*

Key A	
Member R	Score=300
Member X	Score=500
Member P	Score=1000
⋮	

*Hash:* có thể sử dụng để lưu object

Key A		Key B		Key C	
Field 1	Value A	Field 1	Value X	Field 1	Value P
Field 2	Value B	Field 2	Value Y	Field 2	Value Q
Field 3	Value C	⋮		Field 3	Value R
Field 4	Value D			⋮	
⋮					


### ***Hướng dẫn cài đặt Redis trên hệ điều hành Ubuntu***

- sudo apt-get update
- sudo apt-get install redis-server

Sau khi cài đặt thành công, Redis sẽ được cài đặt trong thư mục /etc/redis. Ta có thể xem cấu hình cài đặt của Redis trong file *.conf* hoặc gõ *redis-cli ->INFO* để xem thông tin cấu hình. Cấu hình mặc định là server localhost và port 6379

***Start server Redis:*** redis-server

Start sever thành công:



```
thachlp@thachlp-Precision-Tower-3620: ~
thachlp@thachlp-Precision-Tower-3620:~$ redis-server
16014:C 01 Nov 15:48:52.325 # Warning: no config file specified, using the default config. In order to specify a config file use redis-server /path/to/redis.conf
16014:M 01 Nov 15:48:52.325 * Increased maximum number of open files to 10032 (it was originally set to 1024).

Redis 3.0.6 (00000000/0) 64 bit

Running in standalone mode
Port: 6379
PID: 16014

http://redis.io
```

### Code minh họa sử dụng Redis

Class *Account*, dùng cấu trúc Hash để lưu trữ

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package thach.le.redis.sample;

/**
 *
 * @author thachlp
 */
public class Account {
    private String userName;
    private String passWord;
    public Account(String userName, String passWord) {
        this.userName = userName;
        this.passWord = passWord;
    }
    public String getUser_name() {
        return userName;
    }
    public void setUser_name(String userName) {
        this.userName = userName;
    }
    public String getPassWord() {
        return passWord;
    }
    public void setPassWord(String passWord) {
        this.passWord = passWord;
    }
}
```

```
    }
}
```

Class *RedisSample* demo cách sử dụng các kiểu dữ liệu trên Redis:

```
/*
 * To change this license header, choose License Headers in Project Properties.
 * To change this template file, choose Tools | Templates
 * and open the template in the editor.
 */
package thach.le.redis.sample;

import java.util.ArrayList;
import java.util.HashMap;
import java.util.List;
import java.util.Map;
import java.util.Set;
import java.util.concurrent.CountDownLatch;
import redis.clients.jedis.Jedis;
import redis.clients.jedis.JedisPubSub;
/**
 *
 * @author thachlp
 */
public class RedisSample {
    public static Map<String, String> putoMap(Account account) {
        Map<String, String> hash = new HashMap<>();
        hash.put("username", account.getUserName());
        hash.put("password", account.getPassWord());
        return hash;
    }
    public static void main(String[] args) throws InterruptedException {
        //Connecting to Redis server on localhost
        Jedis jedis = new Jedis("localhost");

        System.out.println("Connection to server sucessfully");
        //check whether server is running or not
        System.out.println("Server is running: " + jedis.ping());

        //redis string
        jedis.set("name", "Thach Le");
        System.out.println("String: " + jedis.get("name"));

        //redis list
        jedis.lpush("fruits", "Banana");
        jedis.lpush("fruits", "Guava");
```

```

jedis.lpush("fruits", "Apple");

// Get the stored data and print it
List list = jedis.lrange("fruits", 0, 3);
for (int i = 0; i < list.size(); i++) {
    System.out.println("List: " + list.get(i));
}

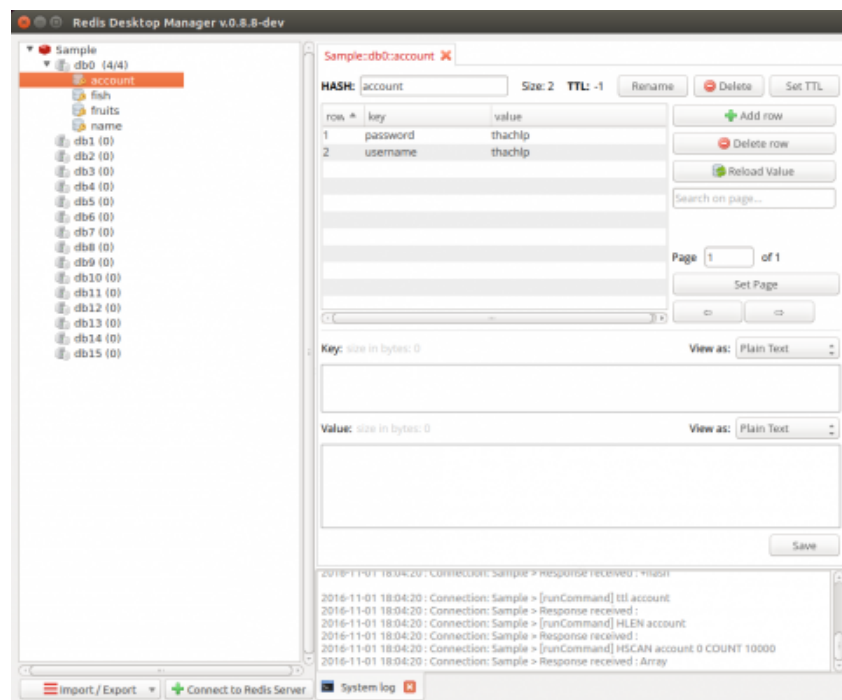
//redis hash
Account account = new Account("thachlp", "thachlp");
jedis.hmset("account", putoMap(account));
Map<String, String> properties = jedis.hgetAll("account");
System.out.println("Account: " + properties.get("username") + "/" + properties.get("password"));

//get data
System.out.println("fish: " + jedis.get("fish"));
jedis.pfmerge("fish", "temp");
System.out.println("temp: " + jedis.get("temp"));
System.out.println("HyperLoglog Count: " + jedis.pfcount("fish"));
}
}

```

Ta có thể dùng command để xem data đã được insert và lưu trữ trong Redis hoặc xem bằng Redis Desktop Manager (xem cách cài đặt [tại đây](#))

**Kết quả data sau khi đã insert:**



Vừa rồi là bài giới thiệu của mình về Redis, mình sẽ tiếp tục chia sẻ những kinh nghiệm, vấn đề trong quá trình sử dụng (nếu có :D), cảm ơn các bạn đã theo dõi.

Rất mong nhận được ý kiến đóng góp của tất cả các bạn.

Xin cảm ơn

Còn rất rất nhiều các tính năng của Redis, các bạn có thể tham khảo thêm tại:

<https://redislabs.com/Downloads/Redis-Labs-Do-You-Really-Know-108-Web.pdf>

<http://redis.io/>

Bài viết gốc được đăng trên Blog cá nhân của mình: [http://thachleblog.com/lam-quen-voi-redis/#codesyntax\\_2](http://thachleblog.com/lam-quen-voi-redis/#codesyntax_2)