Redis是一个key-value存储系统。和Memcached类似，它支持存储的value类型相对更多，包括string(字符串)、list(链表)、set(集合)、zset(sorted set --有序集合)和hash（哈希类型）。这些数据类型都支持push/pop、add/remove及取交集并集和差集及更丰富的操作，而且这些操作都是原子性的。在此基础上，redis支持各种不同方式的排序。与memcached一样，为了保证效率，数据都是缓存在内存中。区别的是redis会周期性的把更新的数据写入磁盘或者把修改操作写入追加的记录文件，并且在此基础上实现了master-slave(主从)同步。

Redis 是一个高性能的key-value数据库。 redis的出现，很大程度补偿了memcached这类key/value存储的不足，在部 分场合可以对关系数据库起到很好的补充作用。它提供了Java，C/C++，C#，PHP，JavaScript，Perl，Object-C，Python，Ruby，Erlang等客户端，使用很方便。

Redis支持主从同步。数据可以从主服务器向任意数量的从服务器上同步，从服务器可以是关联其他从服务器的主服务器。这使得Redis可执行单层树复制。存盘可以有意无意的对数据进行写操作。由于完全实现了发布/订阅机制，使得从数据库在任何地方同步树时，可订阅一个频道并接收主服务器完整的消息发布记录。同步对读取操作的可扩展性和数据冗余很有帮助。

以上是百度百科上面关于Redis的一些介绍，现在就来学习一下具体Redis的安装和使用。

Redis的下载链接为一下的地址，选择自己合适的版本。https://github.com/MSOpenTech/redis/releases

下载完成后，直接解压到指定的文件夹，在修改完文件夹名称后，直接在文件夹内部运行cmd命令redis-server.exe redis.windows.conf。如果你修改了默认的配置路径，就可以直接敲redis-server.exe。运行成功后的界面如下：



运行成功后，现在Redis的客户端敲一些命令，熟悉一下。原来的cmd窗口不要关闭，在开启一个cmd窗口。命令为：redis-cli.exe -h 127.0.0.1 -p 6379。主机号和端口号大家应该都知道，不再介绍。成功界面如下：

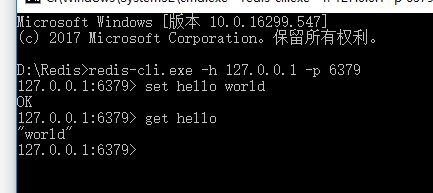


现在开始存储key value。命令如下：

set Hello World

get Hello

成功界面如下：



剩下就是java代码了。编写代码需要jar包，我这里把依赖放上去，jar包会在我的GitHub上面的项目里面。我这里的代码使用的是jar包。依赖如下：

<!-- https://mvnrepository.com/artifact/redis.clients/jedis -->

<dependency>

<groupId>redis.clients</groupId>

<artifactId>jedis</artifactId>

<version>2.9.0</version>

</dependency>

剩下就是代码了，代码没有什么难度，我在代码中的注释也比较详细。没有什么可以讲的，就是一个工具，大家会使用就可以了。在使用的过程中去学习也是可以的。代码如下：

**package** RedisTrain;

**import** java.util.List;

**import** java.util.Set;

**import** redis.clients.jedis.Jedis;

**public** **class** RedisTrainDemo1 {

**public** **static** **void** main(String[] args) {

/\*\*

\* 特点

\* Redis支持数据的持久化，可以将内存中的数据保存在磁盘中，重启的时候可以再次加载进行使用。

\* Redis不仅仅支持简单的key-value类型的数据，同时还提供list，set，zset，hash等数据结构的存储。

\* Redis支持数据的备份，即master-slave模式的数据备份。

\*

\* 优点

\* 性能极高 – Redis能读的速度是110000次/s,写的速度是81000次/s 。 丰富的数据类型 – Redis支持二进制案例的

\* Strings, Lists, Hashes, Sets 及 Ordered Sets 数据类型操作。 原子 –

\* Redis的所有操作都是原子性的，意思就是要么成功执行要么失败完全不执行。单个操作是原子性的。多个操作也支持事务，即原子性，通过MULTI和EXEC指令包起来。

\* 丰富的特性 – Redis还支持 publish/subscribe, 通知, key 过期等等特性。

\*

\*/

// 连接本地的Redis服务

//redis.clients.jedis.exceptions.JedisConnectionException这个是Redis数据库没有启动的异常，重新启动即可

Jedis jedis = **new** Jedis("localhost");

System.***out***.println("本地Redis连接服务成功");

// 查看本地Redis服务状态

System.***out***.println("本地Redis服务状态:" + jedis.ping());

//存储字符串数据

jedis.set("baidu", "www.baidu.com");

System.***out***.println("存储的字符串数据为:"+jedis.get("baidu"));

//存储list数据

jedis.lpush("list","haha1");

jedis.lpush("list","haha2");

jedis.lpush("list","haha3");

jedis.lpush("list","haha4");

//取出list数据

List<String> list = jedis.lrange("list", 0, 4);

System.***out***.print("存储的list集合内的数据为:");

**for**(String zfc:list) {

System.***out***.print(zfc+" ");

}

System.***out***.println();

//获取

//jedis.keys该方法返回的是一个set，必须要使用set。可以只用强转

Set<String> keys = jedis.keys("\*");

System.***out***.print("Redis中的key有:");

**for**(String key:keys) {

System.***out***.print(key+" ");

}

}

}

剩下就没有什么难点了，主要还是在项目中的使用。另外，我的学习都是在这个网站上面学习的http://www.runoob.com/redis/redis-conf.html