

主要代码

笔记本： 0.Redis

创建时间： 2018/12/14 15:26

更新时间： 2018/12/14 15:34

作者： 2363655324idjzb

主要代码

- Redis 作业

```
// stringRedisTemplate.opsForValue().set("test","100",60*10,TimeUnit.SECONDS);//向redis里存入数据和设置缓存时间
// stringRedisTemplate.boundValueOps("test").increment(-1);//val做-1操作
// stringRedisTemplate.opsForValue().get("test");//根据key获取缓存中的val
// stringRedisTemplate.boundValueOps("test").increment(1);//val +1
// stringRedisTemplate.getExpire("test");//根据key获取过期时间
// stringRedisTemplate.getExpire("test",TimeUnit.SECONDS)//根据key获取过期时间并换算成指定单位
// stringRedisTemplate.delete("test");//根据key删除缓存
// stringRedisTemplate.hasKey("546545");//检查key是否存在，返回boolean值
// stringRedisTemplate.opsForSet().add("red_123", "1","2","3");//向指定key中存放set集合
// stringRedisTemplate.expire("red_123",1000, TimeUnit.MILLISECONDS);//设置过期时间
// stringRedisTemplate.opsForSet().isMember("red_123", "1");//根据key查看集合中是否存在指定数据
// stringRedisTemplate.opsForSet().members("red_123");//根据key获取set集合
// 获取当前时间戳
// //方法 一
// System.currentTimeMillis();
// //方法 二
// Calendar.getInstance().getTimeInMillis();
// //方法 三
// new Date().getTime();
```

```
@RequestMapping("/sendMessage")
public String sendMessage() {
    String startTime = Long.toString(System.currentTimeMillis());
    int count;
    String MobileMssage = "看短信";
    redisTemplate.opsForValue().set("start", startTime, 30 * 60L, TimeUnit.SECONDS);
    if (!redisTemplate.hasKey(MobileMssage)) {
        count = 1;
        redisTemplate.opsForValue().set(MobileMssage, String.valueOf(count), 60L, TimeUnit.SECONDS);
        String result = MobileMssage + "已发送" + redisTemplate.opsForValue().get(MobileMssage)
+ "条";
        return result;
    }
    String times = redisTemplate.opsForValue().get(MobileMssage);
    long start = Long.parseLong(redisTemplate.opsForValue().get("start"));
    count = Integer.parseInt(times);
    count++;
    if (count > 3 || (System.currentTimeMillis() - start) < 30) {
        return "发送请求过多，请稍后再试";
    } else {
        redisTemplate.opsForValue().set(MobileMssage, String.valueOf(count), 60L, TimeUnit.SECONDS);
        String result = MobileMssage + "已发送" + redisTemplate.opsForValue().get(MobileMssage)
+ "条";
        return result;
    }
}
```

- mongodb 作业
 - 两种方法

```
//method1
```

```
// @RequestMapping("/withinPolygon")
// public List<DBObject> withinPolygon(){
//     DBObject query = new BasicDBObject();
//     DBObject field = new BasicDBObject();
//     int limit = 3;//
//     List<double[]> list = new ArrayList<>();
//     list.add(new double[]{110,40});
//     list.add(new double[]{110,0});
//     list.add(new double[]{120,0});
//     list.add(new double[]{120,40});
//     list.add(new double[]{110,40});

//     List<List<double[]>> list2 = new ArrayList<>();
//     list2.add(list);
//     list2.add(list1);

//     List<DBObject> listfinal = userService.withinPolygon("point.test", "loc",
//     list2, field, query, limit);
//     for(DBObject obj : listfinal)
//         System.out.println(obj);
//     return listfinal;
// }
```

```
/ method2
@RequestMapping("/withinPolygon1")
public List<DBObject> withinPolygon1(){
    DBObject query = new BasicDBObject();
    DBObject field = new BasicDBObject();
    int limit = 2;//
    List<double[]> list = new ArrayList<>();
    list.add(new double[]{110,40});
    list.add(new double[]{110,0});
    list.add(new double[]{120,0});
    list.add(new double[]{120,40});
    list.add(new double[]{110,40});

    List<double[]> list1 = new ArrayList<>();
    list1.add(new double[]{200,40});
    list1.add(new double[]{210,0});
    list1.add(new double[]{220,0});
    list1.add(new double[]{220,40});
    list1.add(new double[]{210,40});

//     List<List<double[]>> list2 = new ArrayList<>();
//     list2.add(list);
//     list2.add(list1);
//

    List<DBObject> listfinal = userService.withinPolygon("point.test", "loc",
        list,list1, field, query, limit);
    for(DBObject obj : listfinal)
        System.out.println(obj);

    return listfinal;
}
```