编号：

2020 - 2021学年第 二 学期



**实 验 报 告**

实验课程名称 软件架构与应用开发

专 业 班 级 计算机1803

学 生 姓 号 31801150

学 生 姓 名 张帅

实验指导教师 彭彬

一. 实验目的和要求

1.学习本节示例L10JpaArticleRedisDemo

2.模仿示例L10RedisCURD在学生信息系统中添加缓存功能

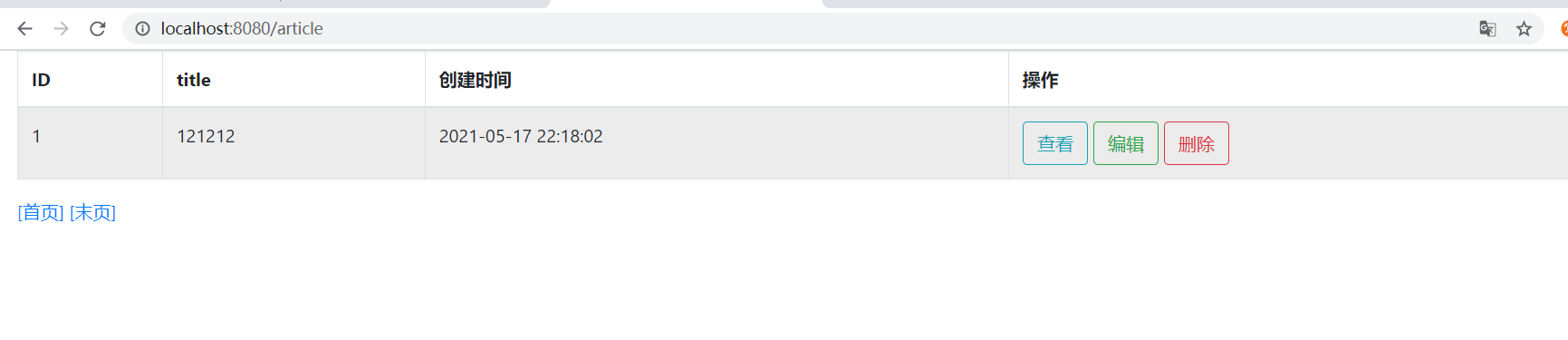
1）基于Redis实现对学生信息系统中学生信息的缓存机制

2）为学生信息系统增加学生绩点信息，并且提供对绩点的增加、更新接口

3）基于Redis的有序列表，实现学生的绩点排名的机制，基于Redis缓存实现接口从Redis快速返回绩点排名的前N名同学

二. 实验内容

1.学习本节示例L10JpaArticleRedisDemo

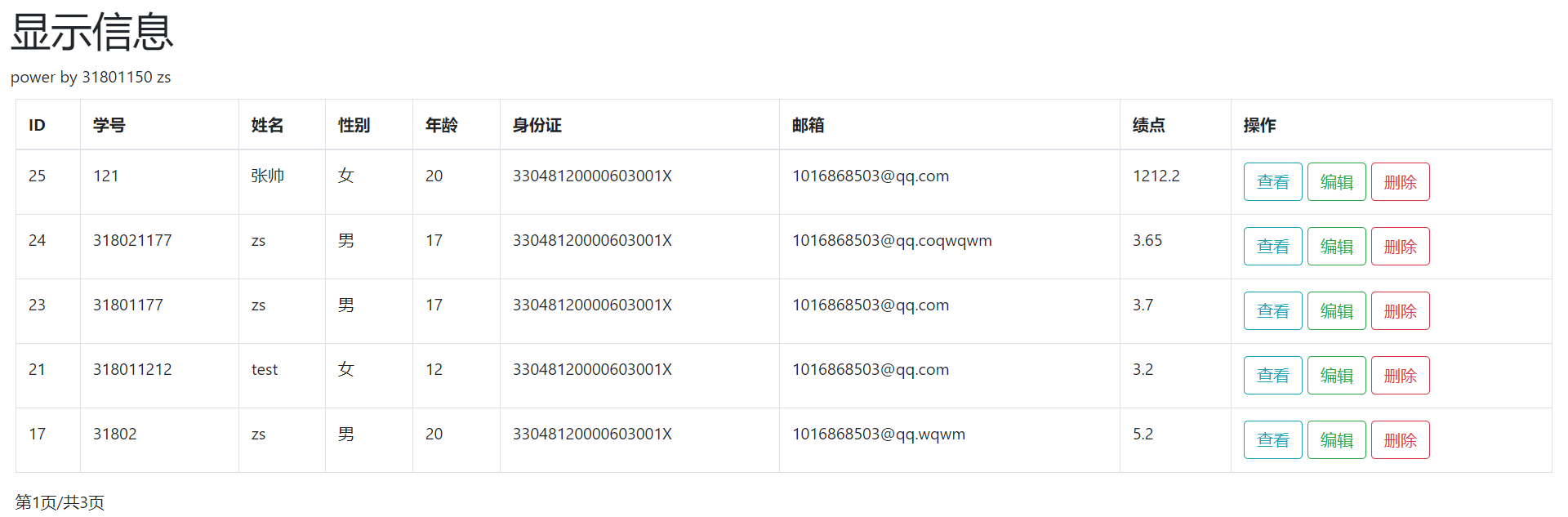


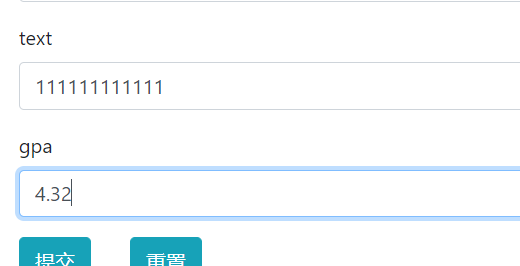
2.模仿示例L10RedisCURD在学生信息系统中添加缓存功能

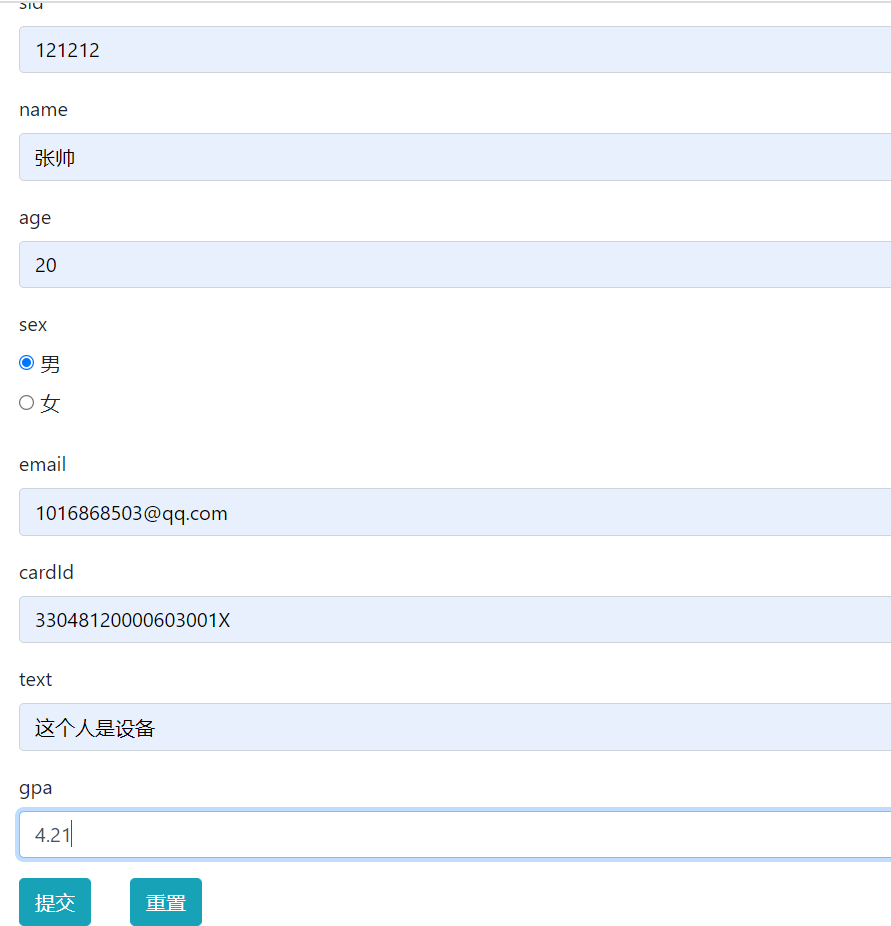
1）基于Redis实现对学生信息系统中学生信息的缓存机制

@Service  
@CacheConfig(cacheNames = "student")  
public class StuService {  
 @Autowired  
 StuMapper stuMapper;  
  
 @Cacheable(key ="#p0")  
 public Stu findbyid(Integer id){  
  
 return stuMapper.queryById(id);  
 }  
  
 @CachePut(key = "#p0")  
 public void updataById(Stu stu){  
  
 stuMapper.updateById(stu);  
 }  
  
  
 @CacheEvict(key ="#p0",allEntries=true)  
 public void deleteById(Integer id){  
  
 stuMapper.delById(id);  
 }  
  
 @CachePut(key ="#p0")  
 public void add(Stu stu){  
  
 stuMapper.add(stu);  
 }  
  
  
  
 public List<Stu> findall(){  
  
 return stuMapper.queryAll();  
 }  
  
  
  
  
}

2）为学生信息系统增加学生绩点信息，并且提供对绩点的增加、更新接口







3）基于Redis的有序列表，实现学生的绩点排名的机制，基于Redis缓存实现接口从Redis快速返回绩点排名的前N名同学@Autowired  
RedisTemplate redisTemplate;  
  
  
//private ZSetOperations zSetOperations;  
  
@Cacheable(key ="#p0")  
public Set rank(int n){  
 List<Stu> re = stuMapper.queryAll();  
 for(int i=0;i<re.size();i++){  
 redisTemplate.opsForZSet().add("stu", re.get(i).getId(),re.get(i).getGpa());  
 }  
  
 return redisTemplate.opsForZSet().reverseRangeWithScores("stu",0,n-1);  
}

