**作业名称：**课后作业 2024.6.6

**文档命名：课后作业-2021214975-黑昀熙-2024.6.6**

**作业内容：**

一、开发接口实现从分数表scores和users表查询用户姓名、考试科目名称、分数并按成绩从高到低排序，通过POSTMAN 展示最终结果截图

1. 构建用户分数（UserScores）数据实体，继承User实体



参考：

|  |
| --- |
| package com.example.entity;  /\*\*  \* SpringBootProj - UserScores  \* @author HeiYunxi 黑昀熙  \* @version 1.0  \*/  public class UserScores extends User {  private String course;  private String score;  public String getCourse() {  return course;  }  public void setCourse(String course) {  this.course = course;  }  public String getScore() {  return score;  }  public void setScore(String score) {  this.score = score;  }  @Override  public String toString() {  return "UserScores{" +  "course='" + course + '\'' +  ", score='" + score + '\'' +  '}';  }  } |

1. 映射文件，包含数据库查询语句

参考：

|  |
| --- |
| <?xml version="1.0" encoding="UTF-8" ?>  <!DOCTYPE mapper  PUBLIC "-//mybatis.org//DTD Mapper 3.0//EN"  "http://mybatis.org/dtd/mybatis-3-mapper.dtd">  <mapper namespace="com.example.mapper.UserScoresMapper">  <select id="SelectUserScoers" resultType="com.example.entity.UserScores">  select u.\*, s.course, s.score from scores s  join users u on s.studentid = u.id  order by s.score desc;  </select>  </mapper> |

1. 构建数据库持久化层，定义其查数据的函数操作

参考：

|  |
| --- |
| package com.example.mapper;  import com.example.entity.UserScores;  import org.springframework.stereotype.Component;  import java.util.List;  /\*\*  \* SpringBootProj - UserScores  \* @author HeiYunxi 黑昀熙  \* @version 1.0  \*/  @Component  public interface UserScoresMapper {  List<UserScores> SelectUserScoers();  } |

1. 构建业务服务层，定义其查数据的函数操作，并完成内部逻辑

参考：

|  |
| --- |
| package com.example.service.impl;  import com.example.entity.UserScores;  import com.example.mapper.UserScoresMapper;  import com.example.service.IUserScoresService;  import org.springframework.beans.factory.annotation.Autowired;  import org.springframework.stereotype.Service;  import org.springframework.transaction.annotation.Transactional;  import java.util.List;  /\*\*  \* SpringBootProj - UserScoresImpl  \* @author HeiYunxi 黑昀熙  \* @version 1.0  \*/  @Service  @Transactional  public class UserScoresServiceImpl implements IUserScoresService {  @Autowired  UserScoresMapper userScoresMapper;  @Override  public List<UserScores> SelectUserScoers(){  return userScoresMapper.SelectUserScoers();  }  } |

1. 构建前端控制层，完成对前端请求的数据的查询请求的响应

参考：

|  |
| --- |
| package com.example.controller;  import com.example.entity.UserScores;  import com.example.service.IUserScoresService;  import org.springframework.beans.factory.annotation.Autowired;  import org.springframework.stereotype.Controller;  import org.springframework.web.bind.annotation.RequestMapping;  import org.springframework.web.bind.annotation.ResponseBody;  import java.util.HashMap;  import java.util.List;  import java.util.Map;  /\*\*  \* SpringBootProj - UserScoresController  \* @author HeiYunxi 黑昀熙  \* @version 1.0  \*/  @Controller  @RequestMapping("user\_scores")  public class UserScoresController {  @Autowired  IUserScoresService iUserScoresService;  @RequestMapping(value = "/SelectUserScoers")  @ResponseBody  public Map<String, Object> SelectUserScoers() {  List<UserScores> list = iUserScoresService.SelectUserScoers();  Map<String, Object> dataMap = new HashMap<>();  dataMap.put("list", list);  return dataMap;  }  } |

6、POSTMAN 展示最终结果截图





