

实验四集合与函数式编程实验.xls

返回

姓名: 李全欣 班级: 计算机2018级4-7 成绩: 100分

一.简答题 (共3题,100.0分)

1

- 一、实验目的：
- 1. 掌握集合的基本操作方法
 - 2. 掌握Lambda表达式的声明编写规范
 - 3. 掌握基于集合Stream的过滤/映射/聚合等操作
 - 4. 掌握基于函数式编程集合元素的移除方法
- 二、实验环境

(5.0分)

正确答案：

我的答案：
操作系统： windows 10 64位
JDK版本 jdk-11.0.6
编译器： eclipse-inst-win64

2

三、实验内容、代码及结果截图

初始化
理解实验的目的，不要浪费时间在初始化的代码，直接复制使用

在com.experiment04.entity下，创建Student类，直接从github复制使用实验代码
在com.experiment04.resource下，创建DatabaseUtils类，直接从github复制使用实验代码，模拟添加若干对象
在com.experiment04.service下，创建StudentService接口，直接从github复制使用实验代码

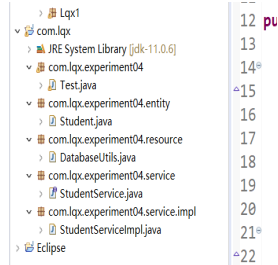
在com.experiment04.service.impl下，创建StudentService接口的实现类StudentServiceImpl，基于集合stream等操作，按接口方法注释需求实现

在com.experiment04下，创建Test类，
为每个接口方法创建静态测试方法，在main()主函数调用
面向接口编程，测试StudentService接口中方法的实现
<https://github.com/bwhyman/java-course/tree/master/java-experiments>

(80.0分)

正确答案：

我的答案：



```
package com.lqx.experiment04.service.impl;
```

```
import java.util.List;
```

```
import java.util.Map;
```

```
import java.util.stream.Collectors;
```

```
import com.lqx.experiment04.entity.Student;
```

```
import com.lqx.experiment04.entity.Student.Sex;
```

```
import com.lqx.experiment04.resource.DatabaseUtils;
```

```
import com.lqx.experiment04.service.StudentService;
```

```
public class StudentServiceImpl implements StudentService {
```

```
    public List<Student> addStudent(Student student) { //向集合添加一个学生，返回当前全部学生
```

```
        List<Student> s = DatabaseUtils.getStudents();
```

```
        s.add(student);
```

```
        return s;
```

```
    }
```

```
    @Override
```

```
    public List<Student> listStudentsByYear(int year) { //返回指定届的全部学生
```

```
        return DatabaseUtils.getStudents()
```

```
            .stream()
```

```
            .filter(s -> s.getYear() == year)
```

```
            .collect(Collectors.toList());
```

```
    }
```

```
    public List<String> listStudentsNames(int year, Sex sex) { //返回指定届，指定性别的全部学生的姓名
```

```
        return DatabaseUtils.getStudents()
```

```
            .stream()
```

```
            .filter(s -> s.getYear() == year)
```

```
            .filter(s -> s.getSex() == sex) //不需要用equals判断相等
```

```
            .map(s -> s.getName())
```

```
            .collect(Collectors.toList());
```

```
    }
```

```
    public Map<Sex, List<Student>> mapStudentsBySex() { //将所有学生，以性别分组
```

```
        return DatabaseUtils.getStudents()
```

```
            .stream()
```

```
            .collect(Collectors.groupingBy(s -> s.getSex()));
```

```
    }
```

```
    public boolean removeStudent(int id) { //删除指定学号的学生，返回是否成功删除
```

```
        return DatabaseUtils.getStudents()
```

```
            .removeIf(s -> s.getId() == id);
```

```
    }
```

```
    }
```

下面为Test里的方法

```
public class Test {
```

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

```
        List<Student> students = DatabaseUtils.getStudents();
```

```
        StudentServiceImpl imple = new StudentServiceImpl();
```

```
        System.out.println("TEST 1: 加入一个学生后的集合: ");
```

```
        imple.addStudent(new Student(201307, Student.Sex.MALE, "王五", 2013));
```

```
        students.forEach(s -> {
```

```
            System.out.println("学号:" + s.getId() + "\t" + "性别:" + s.getSex() + "\t" + "姓名:" + s.getName() + "\t" + "年份:" + s.getYear());
```

```
        });
```

```
    }
```

}

```

TEST 1: 加入一个学生后的集合:
学号:201001    性别:FEMALE    姓名:赵阳阳    年份2010
学号:201002    性别:MALE     姓名:邵鹏     年份2010
学号:201103    性别:MALE     姓名:高学斌   年份2011
学号:201104    性别:MALE     姓名:张杨     年份2011
学号:201205    性别:FEMALE   姓名:吕惠玲   年份2012
学号:201206    性别:MALE     姓名:曾志优   年份2012
学号:201307    性别:MALE     姓名:王五     年份2013

```

```

public class Test {
    public static void main(String[] args) {
        List<Student> students=DatabaseUtils.getStudents();
        StudentServiceImpl imple=new StudentServiceImpl();

        System.out.println("TEST 2: 年份为2012的学生集合: ");
        imple.listStudentsByYear(2012)
        .forEach(s->{
            System.out.println("学号:"+s.getId()+"\t"+"性别:"+s.getSex()+"\t"+"姓名:"+s.getName()+"\t"+"年份"+s.getYear());
        });
    }
}

```

```

TEST 2: 年份为2012的学生集合:
学号:201205    性别:FEMALE   姓名:吕惠玲   年份2012
学号:201206    性别:MALE     姓名:曾志优   年份2012

```

```

public class Test {
    public static void main(String[] args) {
        List<Student> students=DatabaseUtils.getStudents();
        StudentServiceImpl imple=new StudentServiceImpl();

        System.out.println("TEST 3: 年份为2012, 性别为女, 学生姓名集合: ");
        imple.listStudentsNames(2012,Sex.FEMALE)
        .forEach(s->{
            System.out.printf("姓名:%s\n",s);
        });
    }
}

```

```

TEST 3: 年份为2012, 性别为女, 学生姓名集合:
姓名:吕惠玲

```

```

public class Test {
    public static void main(String[] args) {
        List<Student> students=DatabaseUtils.getStudents();
        StudentServiceImpl imple=new StudentServiceImpl();

        System.out.println("TEST 4: 将所有学生, 以性别分组: ");
        imple.mapStudentsBySex()
        .forEach((sex,student)->{
            System.out.println("性别是:    "+sex);
            student.forEach(s->System.out.println("学号:"+s.getId()+"\t"+"性别:"+s.getSex()+"\t"+"姓名:"+s.getName()+"\t"+"年份"+s.getYear()));
        });
    }
}

```

```

TEST 4: 将所有学生, 以性别分组:
性别是:    MALE
学号:201002    性别:MALE     姓名:邵鹏     年份2010
学号:201103    性别:MALE     姓名:高学斌   年份2011
学号:201104    性别:MALE     姓名:张杨     年份2011
学号:201206    性别:MALE     姓名:曾志优   年份2012
学号:201307    性别:MALE     姓名:王五     年份2013
性别是:    FEMALE
学号:201001    性别:FEMALE   姓名:赵阳阳   年份2010
学号:201205    性别:FEMALE   姓名:吕惠玲   年份2012

```

```

public class Test {
    public static void main(String[] args) {
        List<Student> students=DatabaseUtils.getStudents();
        StudentServiceImpl imple=new StudentServiceImpl();

        System.out.println("TEST 5: 删除学号为201205的学生, 返回是否删除成功: ");
        System.out.println(imple.removeStudent(201205));
    }
}

```

```

TEST 5: 删除学号为201205的学生, 返回是否删除成功:
true

```

3

四、实验过程分析与讨论

(15.0分)

正确答案:

我的答案:

```
System.out.println("TEST 3: 年份为2012, 性别为女, 学生姓名集合: ");
imple.listStudentsNames(2012, Sex.FEMALE)
    .forEach(s->{
        System.out.printf("姓名:%s\n", s);
    });
```

这里的s不能用s.getName因为前面map了s.getName()
