

华侨大学工学院

实验报告

课程名称: **JAVA 实验**

实验项目名称: Java 的类和对象实验

学 院： 工学院

专业班级: 物联网工程

姓 名： 李昊唐

学 号: 1995131017

指导教师:

2021 年 12 月 13 日

预 习 报 告

一、 实验目的

熟悉 **Java** 的类和对象的概念。

二、 实验仪器

掌握 **Java** 类和对象的使用。

三、 实验原理

1. 设计一个课程类，包括课程的名称、编号、成绩；课程类能够实现的操作包括：
 - (1) 显示课程的所有信息；
 - (2) 统计课程成绩的平均分；
 - (3) 统计课程成绩的最低分；
 - (4) 统计课程成绩的最高分。
2. 设计一个学生类，包括学生的学号、姓名、性别、年龄、身高、生源地、所在学校、专业、及所选的课程（含：课程编号 **v**、课程名称、课程成绩）；学生类能够实现的操作包括：

预 习 报 告

- (1) 显示学生的所有信息;
- (2) 当挂科课程数量高于总课程数 40% 时, 自动对该学生提出警告;
- (3) 根据学生的姓名或学号来查询学生的成绩。

四、实验内容及步骤

1、 课程类

```
package lei.houtong.exp;

import java.sql.*;

public class Course {
    /* 设计一个课程类, 包括课程的名称、编号、成绩 */
    String course, id;
    float max, min, avg;

    Connection conn;
    Statement stmt;
    ResultSet rs;

    public Course(Connection conn) throws SQLException {
        this.conn = conn;
        this.stmt = this.conn.createStatement();
    }

    public void save_db() throws SQLException {
        this.stmt.executeQuery("INSERT INTO courses VALUES(" + id +
            "," + course + ")");
    }

    public String find(String id) throws SQLException {
```

实验报告

五、实验原始数据

```
/* 显示课程的所有信息 */
this.rs = this.stmt.executeQuery("SELECT * FROM courses
WHERE id=" + id);
String name = null;
while (this.rs.next()) name = this.rs.getString("course");
return name;
}

public float avg(String id) throws SQLException {
    /* 统计课程成绩的平均分 */
    this.rs = this.stmt.executeQuery("SELECT AVG(score) FROM
electives WHERE id=" + id);
    while (this.rs.next()) this.avg = this.rs.getFloat(1);
    return this.avg;
}

public float min(String id) throws SQLException {
    /* 统计课程成绩的最低分 */
    this.rs = this.stmt.executeQuery("SELECT MIN(score) FROM
electives WHERE id=" + id);
    while (this.rs.next()) this.min = this.rs.getFloat(1);
    return this.min;
}

public float max(String id) throws SQLException {
    /* 统计课程成绩的最高分 */
    this.rs = this.stmt.executeQuery("SELECT MAX(score) FROM
electives WHERE id=" + id);
    while (this.rs.next()) this.max = this.rs.getFloat(1);
    return this.max;
}
}
```

2、学生类

```
package lei.houtong.exp;

import java.sql.*;
```

```
public class Student {
    /* 设计一个学生类，包括学生的学号、姓名、性别、年龄、身高、生源地、所在学
校、专业、及所选的课程（含：课程编号、课程名称、课程成绩） */
```

实验报告

六、 数据处理

```
String no, name, gender;
int age;
float height, score;
String source, school, major;
boolean postgraduate = false;

Connection conn;
Statement stmt;
ResultSet rs;

public Student(Connection conn) throws SQLException{
    this.conn = conn;
    this.stmt = this.conn.createStatement();
}

public void save_db() throws SQLException{
    /* 输入学生信息 */
    PreparedStatement stmt = this.conn.prepareStatement("INSERT
INTO students VALUES(?,?,?,?,?,?,?,?,?,?,?)");
    stmt.setString(1, this.no);
    stmt.setString(2, this.name);
    stmt.setString(3, this.gender);
    stmt.setInt(4, this.age);
    stmt.setFloat(5, this.height);
    stmt.setString(6, this.source);
    stmt.setString(7, this.school);
    stmt.setString(8, this.major);
    stmt.setBoolean(9, this.postgraduate);
    stmt.setNull(10, Types.NULL);
    stmt.setNull(11, Types.NULL);
    stmt.executeUpdate();
}

public ResultSet find(String no) throws SQLException {
    /* 显示学生的所有信息 */
    return this.stmt.executeQuery("SELECT * FROM students WHERE
`no`=" + no);
}

public boolean fail(String no) throws SQLException {
    /* 当挂科课程数量高于总课程数 40%时, 自动对该学生提出警告 */
```

实验报告

七、实验结论及分析讨论

```
this.rs = this.stmt.executeQuery("SELECT COUNT(*) / (SELECT  
COUNT(*) FROM electives WHERE `no`=" + no + ") FROM electives  
WHERE score<60 AND `no`=" + no);  
float fail_rate = 0;  
while (rs.next()) { fail_rate = rs.getFloat(1); }  
return fail_rate > 0.4;  
}  
  
public void save_score(String no, String id, Float score)  
throws SQLException{  
    this.stmt.executeQuery("INSERT INTO electives VALUES(" + no  
+ "," + id + "," + score + ")");  
}  
  
/* 根据学生的姓名或学号来查询学生的成绩 */  
  
public String find_id(String name) throws SQLException {  
    this.rs = this.stmt.executeQuery("SELECT `no` FROM students  
WHERE name=\"" + name + "\"");  
    while (rs.next()) { this.no = rs.getString(1); }  
    return this.no;  
}  
  
public String find_score(String field, String id, boolean Name)  
throws SQLException {  
    if (Name) this.no = find_id(field); else this.no = field;  
    this.rs = this.stmt.executeQuery("SELECT score FROM  
electives WHERE id=" + id + " AND `no`=" + this.no);  
    while (this.rs.next()) { this.score =  
this.rs.getFloat(1); }  
    return String.valueOf(this.score);  
}  
}
```

预习报告成绩	实验报告成绩	实验操作成绩	总成绩