

# 山西工程技术学院

## 实验报告

( 2024- 2025 学年第一学期 )

课程名称: 数据库系统原理

专业班级:

学 号:

学生姓名:

任课教师:

2024 年 9 月

# 实验报告

实验名称	数据查询			指导教师	李素萍
实验类型	验证型	实验学时	4	实验时间	
<b>一、实验目的与要求</b> 1. 了解查询的概念和方法； 2. 掌握查询分析器的使用方法； 3. 掌握 SELECT 语句在单表查询中的应用； 4. 掌握复杂查询的使用方法； 5. 掌握多表连接的方法； 6. 掌握 SELECT 语句在多表查询中的应用。					
<b>二、实验环境</b> 操作系统：win-7 编程环境：Microsoft SQL Server 2008					
<b>三、实验内容和步骤</b> <b>一、表的创建</b> 1、写出并执行语句创建教工表，条件如下：					
列名	数据类型	允许为空	约束	IDENTITY 属性	
教工号	Int	否	主键	无	
姓名	Varchar(20)	是	无	无	
家庭地址	Char(30)	是	无	无	
性别	Char(2)	是	默认为“男”	无	
电话	Int	是	无	无	
工资	Int	是	无	无	
系别	Varchar(16)	是	必须是（计算机软 件, 计算机体系结构, 计算机网络）之一	无	
职称	Varchar(10)	是	无	无	
课程名称	Varchar(10)	是	必须在（语文, 数学, 英语，化学，物理） 中	无	

CREATE TABLE Teacher (

教工号 INT PRIMARY KEY, -- 教工号，主键

姓名 VARCHAR(20) NULL,

家庭地址 CHAR(30) NULL,

性别 CHAR(2) DEFAULT '男',

电话 VARCHAR(15) NULL,

工资 INT NULL,

系别 VARCHAR(16) CHECK (系别 IN ('计算机软件', '计算机体系结构', '计算机网络')),

职称 VARCHAR(10) NULL,

课程名称 VARCHAR(10) CHECK (课程名称 IN ('语文', '数学', '英语', '化学', '物理'))

);

“电话”字段用于存储电话号码，使用 VARCHAR 类型，而不是 INT，因为电话号码可能包含前导零或其他非数字字符（如+）

2. )插入数据（不少于 5 条记录）

INSERT INTO Teacher (教工号, 姓名, 家庭地址, 性别, 电话, 工资, 系别, 职称, 课程名称)

VALUES

(1001, '张三', '北京市朝阳区', '男', 13812345678, 8000, '计算机软件', '教授', '数学'),

(1002, '李四', '上海市浦东新区', '女', 13998765432, 7500, '计算机体系结构', '副教授', '英语'),

(1003, '王五', '广州市天河区', '男', 13654321098, 7000, '计算机网络', '讲师', '化学'),

(1004, '赵六', '深圳市南山区', '女', 13512349876, 6500, '计算机软件', '助理教授', '物理'),

(1005, '孙七', '杭州市西湖区', '男', 13456781234, 7200, '计算机网络', '教授', '语文');

## 二、单表查询

1. 基本查询

打开查询分析器，从 teacher 表中分别检索出教师的所有信息，以及教工号、姓名和职称。

```
select * from teacher
```

```
select 教工号, 姓名 from teacher
```

2. 查询时改变列标题的显示

从 teacher 表中分别检索出教师教工号、姓名、家庭住址信息并分别加上‘教师姓名’、‘教工号’、‘家庭住址’等标题信息。

3. 基于 WHERE 语句进行数据查询

(1) 基于比较条件。从 teacher 表中查询出教工号小于 2130 的教师资料。

```
select * from teacher where 教工号<2130
```

(2) 基于 BETWEEN 语句。从 teacher 表中查询出教工号介于 2100 和 2130 之间的教师资料。

```
select * from teacher where 教工号 between 2100 and 2130
```

(3) 基于 IN 子句的数据查询。从 teacher 表中查询出职称为“教授”或“副教授”的教师的教工号、教师姓名、职称及家庭住址。

```
select 教工号,姓名,家庭住址,职称 from teacher  
where 职称 in ('教授','副教授')
```

(4) 基于 LIKE 子句的查询。从 teacher 表中分别检索出姓伍的教师的资料，或者姓名的第 2 个字是寿或立的教师的资料。

```
select * from teacher where 姓名 like '张%'  
select * from teacher where 姓名 like '_[寿，立]%'
```

(5) 使用 TOP 关键字查询。分别从 teacher 中检索出前 2 条及前面 67% 的教师的信息。

```
select top 2 * from teacher  
select top 67 percent * from teacher
```

(6) 使用 DISTINCT 关键字查询。从 teacher 表中检索出教师的职称并且要求显示的职称不重复。

```
select distinct 职称 from teacher
```

(7) 使用 ORDER BY 语句对查询的结果进行排序 ASC、DESC 分别是升序和降序排列的关键字，系统默认的是升序排列。从 teacher 表中查询工资大于 2800 的教师的教工号、姓名，并按升序排列，

```
select 教工号,姓名 from teacher  
where 工资>2800 order by 工资 ASC
```

### (3)多表查询

数据库的各个表中存放着不同的数据，用户经常需要用多个表中的数据来组合提炼出所需要的信息，如果一个查询需要对多个表进行操作，就称为联表查询，联表查询的结果集或结果表称为表之间的连接。联表查询实际上是通过各表之间共同列的关联来查询数据的，它是关系数据库查询最基本的特征  
按照下表所示,分别在数据库 test 中构造 student、course 和 student\_course 三张表，并写入记录。

Student 表

列名称	类型	宽度	允许为空	缺省值	主键
学号	Char	8	否		是
学生姓名	Nvarchar	8	否		
性别	Bit		否		
出生年月	Smalldatetime		否		
班级号	Char	6	否		
入学时间	Smalldatetime		否		
家庭住址	Nvarchar	40	是		

Course 表

列名称	类型	宽度	允许为空	缺省值	主键
课程号	char	10	否		是
课程名称	Nvarchar	20	否		
书标识	Char	13	否		
课程总学时	Tinyint		是		
周学时	Tinyint		是		
课程学分	Tinyint		是		

Student\_course 表

列名称	类型	宽度	允许为空	缺省值	主键
课程号	Char	10	否		是
学号	Char	8	否		是
成绩	Tinyint		否		
学分	Tinyint		否		
学期	Bit		是		
学年	Char	9	是		

SQL 2008 兼容 2 种连接形式：用于 FROM 子句的 ANSI 连接语法和用于 WHERE 子句的 SQL SERVER 连接语法形式。

#### 1. 进行多表查询

从 student、course 和 student\_course 三张表中检索学生的学号、姓名、学习课程号、学习课程名及课程成绩。

```
select student.学号, student_course.学号, student_course.课程号, course.课程名称,
student_course.成绩 from student, student_course, course
where student.学号 = student_course.学号
and course.课程号 = student_course.课程号
```

## 2. 使用 UNION 子句进行查询

使用 UNION 子句可以将一个或者多个表的某些数据类型相同的列显示在同一列上。如从 teacher 表中列出教工号、姓名并从 student 表中列出学号及学生姓名。

```
SELECT CAST(学号 AS VARCHAR(50)) AS 代码, 学生姓名 AS 姓名 FROM student  
UNION
```

```
SELECT CAST(教工号 AS VARCHAR(50)) AS 代码, 姓名 AS 姓名 FROM teacher;
```

遇到**字段类型不匹配**的问题:学号 和 教工号 的类型不同, 强制转换字段的类型: **AS VARCHAR(50)**学号 和 教工号 都会被转换为 VARCHAR 类型, 确保类型一致。

## 3.用 GROUP 子句进行查询

如果要在数据检索时对表中数据按照一定条件进行分组汇总或求平均值, 就要在 SELECT 语句中与 GROUP BY 子句一起使用集合函数。使用 GROUP BY 子句进行数据检索可得到数据分类的汇总统计、平均值或其他统计信息。

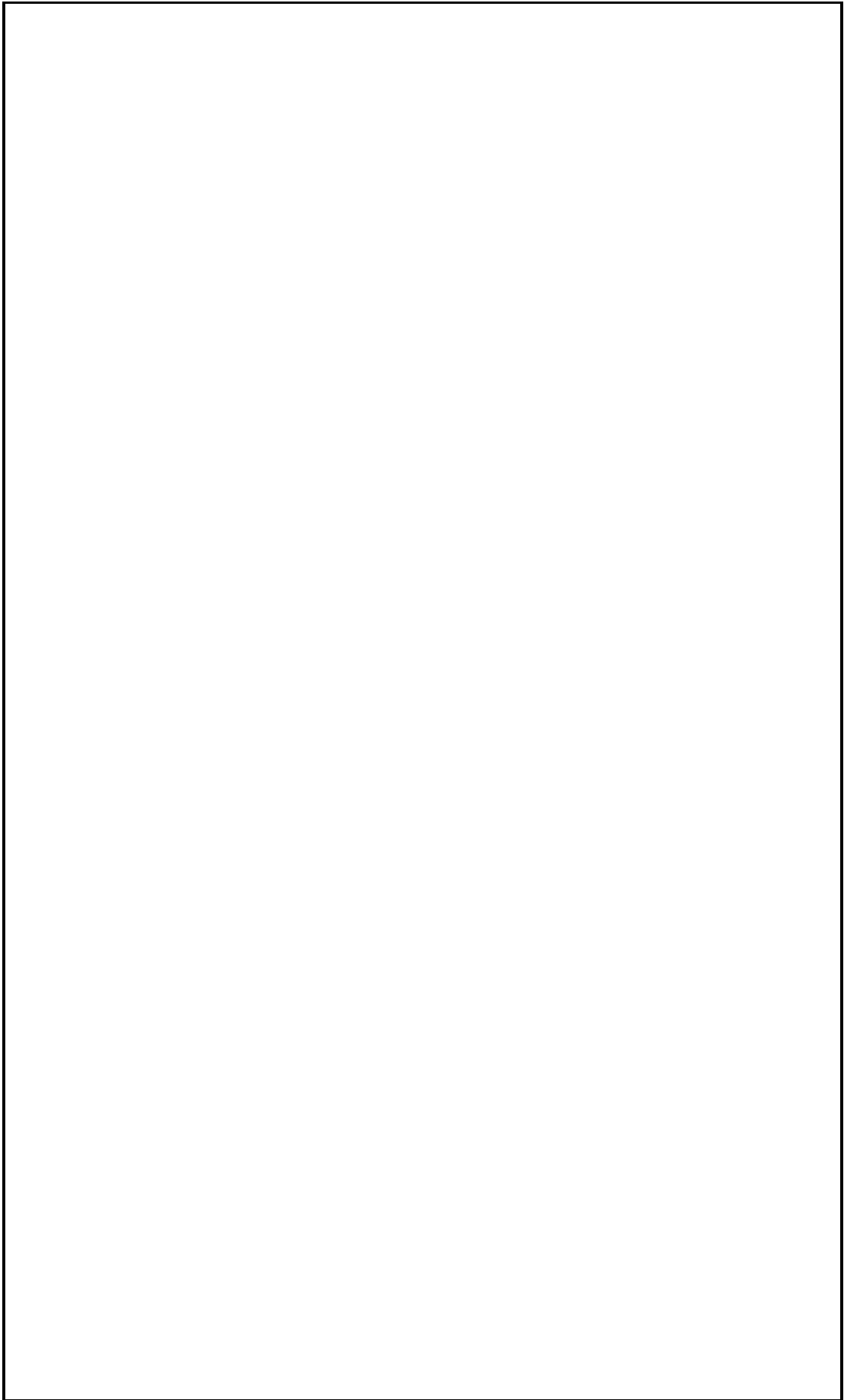
(1) 使用不带 HAVING 的 GROUP BY 子句。使用不带 HAVING 的 GROUP BY 子句汇总出 student\_course 表中的学生的学号及总成绩。

```
select '学号' =student.学号, '总成绩' =SUM(成绩)from student_course  
group by 学号
```

(2) 使用带 HAVING 的 GROUP BY 子句。使用带 HAVING 的 GROUP BY 子句汇总出 student\_course 表中总分大于 450 分的学生的学号及总成绩,

```
SELECT 学号, SUM(成绩) AS 总成绩  
FROM student_course  
GROUP BY 学号  
HAVING SUM(成绩) > 450;
```

```
SELECT 学号, SUM(成绩) AS 总成绩  
FROM student_course  
GROUP BY 学号  
HAVING SUM(成绩) > 100;
```



### 三、进行嵌套查询

#### 1. 使用 IN 或 NOT IN 关键字

使用 IN 关键字查询出 j10011 班所有男生的学号、课程号及相应的成绩。

```
select student_course.学号, student_course.课程号, student_course.成绩
FROM student_course
WHERE 学号 IN
    (SELECT 学号 FROM student
     WHERE 班级= 'j10011' AND 性别=1)
```

#### 2. 使用 EXISTS 或 NOT EXISTS 关键字。使用 EXISTS 关键字查询出 'j10011' 班的学生学号、课程号及相应的成绩

```
SELECT student_course.学号, student_course.课程号, student_course.成绩
FROM student_course
WHERE EXIST
    (SELECT * FROM student
     WHERE student_course.学号=student.学号 AND student.班级= 'j10011' )
```

### 四、实验小结和思考

通过这次实验，我深刻认识到数据库查询的语句的使用。SQL 不仅仅是用来查询数据，它还是数据库管理和操作的核心工具。使用不同类型的语句可以达到不同的效果，在过程中，我遇到了很多问题，如超出范围，数据类型不同等，后续我通过资料查询解决了问题，学会了使用强制类型转换，以及超出范围时的报错判断。

实验成绩		批阅日期		批阅人	
------	--	------	--	-----	--



附件（实验结果）

一、表的创建

1、写出并执行语句创建教职工表

列

教工号 (PK, int, not null)

姓名 (varchar(20), null)

家庭地址 (char(30), null)

性别 (char(2), null)

电话 (int, null)

工资 (int, null)

系别 (varchar(16), null)

职称 (varchar(10), null)

课程名称 (varchar(10), null)

100 %

消息

命令已成功完成。

完成时间： 2024-11-

2、插入数据（不少于 5 条记录）

消息

命令已成功完成。

完成时间： 2024-11-27T13:58:19.3583709+08:00

二、单表查询

1. 基本查询

结果 消息

	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称
1	1001	张三	北京市朝阳区	男	13812345678	8000	计算机软件	教授	数学
2	1002	李四	上海市浦东新区	女	13998765432	7500	计算机体系结构	副教授	英语
3	1003	王五	广州市天河区	男	13654321098	7000	计算机网络	讲师	化学
4	1004	赵六	深圳市南山区	女	13512349876	6500	计算机软件	助理教授	物理
5	1005	孙七	杭州市西湖区	男	13456781234	7200	计算机网络	教授	语文

	教工号	姓名
1	1001	张三
2	1002	李四
3	1003	王五
4	1004	赵六
5	1005	孙七

2.查询时改变列标题的显示

结果 消息				
	教工号	姓名	家庭地址	
1	1001	张三	北京市朝阳区	
2	1002	李四	上海市浦东新区	
3	1003	王五	广州市天河区	
4	1004	赵六	深圳市南山区	
5	1005	孙七	杭州市西湖区	

---

	教工号	姓名	家庭地址	
1	1001	张三	北京市朝阳区	
2	1002	李四	上海市浦东新区	
3	1003	王五	广州市天河区	
4	1004	赵六	深圳市南山区	
5	1005	孙七	杭州市西湖区	

### 3. 基于 WHERE 语句进行数据查询

(1) 基于比较条件。

	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称
1	1001	张三	北京市朝阳区	男	13812345678	8000	计算机软件	教授	数学
2	1002	李四	上海市浦东新区	女	13998765432	7500	计算机体系结构	副教授	英语
3	1003	王五	广州市天河区	男	13654321098	7000	计算机网络	讲师	化学
4	1004	赵六	深圳市南山区	女	13512349876	6500	计算机软件	助理教授	物理
5	1005	孙七	杭州市西湖区	男	13456781234	7200	计算机网络	教授	语文

(2) 基于 BETWEEN 语句。

结果 消息									
	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称

当语句为 `select * from teacher where 教工号 between 2100 and 2130` 时，无法查询出结果，应为插入的“教工号”只有“1001-1005”，查询已超出范围。

可以改为 `select * from teacher where 教工号 between 1000 and 1002`，即可查询到以下数据：

结果 消息									
	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称
1	1001	张三	北京市朝阳区	男	13812345678	8000	计算机软件	教授	数学
2	1002	李四	上海市浦东新区	女	13998765432	7500	计算机体系结构	副教授	英语

(3) 基于 IN 子句的数据查询。

结果 消息				
	教工号	姓名	家庭地址	职称
1	1001	张三	北京市朝阳区	教授
2	1002	李四	上海市浦东新区	副教授
3	1005	孙七	杭州市西湖区	教授

(4) 基于 LIKE 子句的查询。

结果 消息									
	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称
1	1001	张三	北京市朝阳区	男	13812345678	8000	计算机软件	教授	数学

语句“select \* from teacher where 姓名 like '张%’” 它的作用是查找所有姓“张”的教师

语句“select \* from teacher where 姓名 like '\_[寿,立]%'” 这个查询会返回所有姓名中第二个字符是“寿”或“立”的教师，但插入数据中无，故没有查询结果

4. 使用 TOP 关键字查询。

结果 消息									
	教工号	姓名	家庭地址	性别	电话	工资	系别	职称	课程名称
1	1001	张三	北京市朝阳区	男	13812345678	8000	计算机软件	教授	数学
2	1002	李四	上海市浦东新区	女	13998765432	7500	计算机体系结构	副教授	英语

5.使用 DISTINCT 关键字查询。

结果 消息	
	职称
1	副教授
2	讲师
3	教授
4	助理教授

## 6.用计算列查询。

结果 消息				
	教工号	姓名	(无列名)	
1	1001	张三	7600.00	
2	1002	李四	7125.00	
3	1003	王五	6650.00	
4	1004	赵六	6175.00	
5	1005	孙七	6840.00	

	教工号	姓名	预法工资
1	1001	张三	7600.00
2	1002	李四	7125.00
3	1003	王五	6650.00
4	1004	赵六	6175.00
5	1005	孙七	6840.00

## 7. 使用 ORDER BY 语句对查询的结果进行排序

	教工号	姓名
1	1004	赵六
2	1003	王五
3	1005	孙七
4	1002	李四
5	1001	张三

## 二、多表查询

### 1. 进行多表查询

结果 消息				
	学号	课程号	课程名称	成绩

未设置查询条件，故无结果，改为以下语句

```
SELECT student.学号, student_course.课程号, course.课程名称, student_course.成绩
FROM student
student_course ON student.学号 = student_course.学号
course ON student_course.课程号 = course.课程号
WHERE student.学号 = '20230101';
```

结果 消息				
	学号	课程号	课程名称	成绩
1	20230101	C000000001	数学分析	90
2	20230101	C000000002	计算机科学导论	85

## 2. 使用 UNION 子句进行查询

结果 消息		
	代码	姓名
1	20230101	张三
2	20230102	李四
3	20230103	王五
4	20230104	赵六
5	20230105	钱七

## 3. 用 GROUP 子句进行查询

(1) 使用不带 HAVING 的 GROUP BY 子句。

结果 消息		
	学号	总成绩
1	20230101	175
2	20230102	92
3	20230103	88
4	20230104	75

(2) 使用带 HAVING 的 GROUP BY 子句。

结果 消息		
	学号	总成绩

由 3 (1) 的结果可知，最大的成绩总分为 175 分，故语句“select ‘学号’=学号，‘总成绩’=SUM(成绩) FROM student\_course GROUP BY 学号 HAVING SUM(成绩)>450”无法查到数据，若改成绩擦查询为“>100”可查询到以下结果：

结果 消息		
	学号	总成绩
1	20230101	175

## 三、进行嵌套查询

1. 使用 IN 或 NOT IN 关键字

结果 消息			
	学号	课程号	成绩
1	20230101	C000000001	90
2	20230101	C000000002	85
3	20230103	C000000004	88

2. 使用 EXISTS 或 NOT EXISTS 关键字。

结果 消息			
	学号	课程号	成绩
1	20230102	C000000003	92