**深 圳 大 学**

**实 验 报 告**

**课程名称： 数据库系统**

**实验序号： 实验1**

**实验名称：**  **SQL的DDL语言和单表查询**

**学 号： 2022155028**

**姓 名： 黄亮铭**

**实验完成日期： 2024年 9 月 21 日**

**一、实验目的：**

1. 了解DBMS系统的功能、软件组成；

2、掌握利用SQL语句定义、和简单操纵数据库的方法。

**二、实验要求：**

1、在课外安装相关软件并浏览软件自带的帮助文件和功能菜单，了解DBMS的功能、结构；  
 2、数据库、关系表定义；

3、学习定义关系表的约束(主键、外键、自定义)；

4、了解SQL的数据定义功能；

5、了解SQL的操纵基本功能；

6、了解视图的概念；

**三、实验设备：**

计算机、数据库管理系统如MYSQL,DB2，Oracle 等软件。

**四、实验内容**

1、使用SQL DDL语句**建立关系数据库模式，并用DML数据**如下；

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| EMPNO | ENAME | JOB | MGR | HIREDATE | SAL | COMM | DEPTNO |
| 7369 | SMITH | CLERK | 7902 | 17-Dec-90 | 13750 |  | 20 |
| 7499 | ALLEN | SALESMAN | 7698 | 20-FEB-89 | 19000 | 6400 | 30 |
| 7521 | WARD | SALESMAN | 7698 | 22-FEB-93 | 18500 | 4250 | 30 |
| 7566 | JONES | MANAGER | 7839 | 02-APR-89 | 26850 |  | 20 |
| 7654 | MARTIN | SALESMAN | 7698 | 28-SEP-97 | 15675 | 3500 | 30 |
| 7698 | BLAKE | MANAGER | 7839 | 01-MAY-90 | 24000 |  | 30 |
| 7782 | CLARK | MANAGER | 7839 | 09-JUN-88 | 27500 |  | 10 |
| 7788 | SCOTT | ANALYST | 7566 | 19-APR-87 | 19500 |  | 20 |
| 7839 | KING | PRESIDENT |  | 17-NOV-83 | 82500 |  | 10 |
| 7844 | TURNER | SALESMAN | 7698 | 08-SEP-92 | 18500 | 6250 | 30 |
| 7876 | ADAMS | CLERK | 7788 | 23-MAY-96 | 11900 |  | 20 |
| 7900 | JAMES | CLERK | 7698 | 03-DEC-95 | 12500 |  | 30 |
| 7902 | FORD | ANALYST | 7566 | 03-DEC-91 | 21500 |  | 20 |
| 7934 | MILLER | CLERK | 7782 | 23-JAN-95 | 13250 |  | 10 |
| 3258 | GREEN | SALESMAN | 4422 | 24-Jul-95 | 18500 | 2750 | 50 |
| 4422 | STEVENS | MANAGER | 7839 | 14-Jan-94 | 24750 |  | 50 |
| 6548 | BARNES | CLERK | 4422 | 16-Jan-95 | 11950 |  | 50 |

**DEPT+(学生自己的学号)**:

|  |  |  |
| --- | --- | --- |
| DEPTNO | DNAME | LOC |
| 10 | ACCOUNTING | LONDON |
| 20 | RESEARCH | PRESTON |
| 30 | SALES | LIVERPOOL |
| 40 | OPERATIONS | STAFFORD |
| 50 | MARKETING | LUTON |

**以下为学生实验填写部分：**

1. **参考课件约束方式，创建emp和dept的DDL语句 （****要有语句和运行结果截屏）**
   1. 使用命令创建数据库，然后使用命令查看数据是否创建成功，如果成功，结果如下图所示。

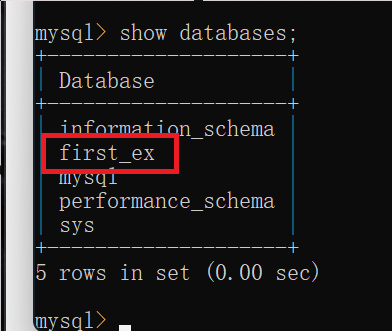
****

图1：数据库创建成功

* 1. 查看两个表中的键，发现EMP表中有DEPT表中的键，即：EMP表中有外键约束。因此，我们需要先创建DEPT表，然后再创建EMP表。
     1. 使用如下命令即可创建DEPT表。

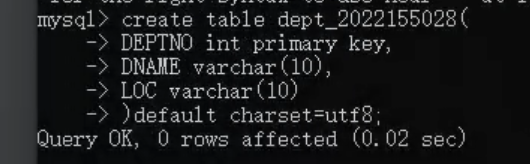
****

图2：创建表dept

* + 1. 检查DEPT表是否创建成功（使用命令），如果成功则结果如下图所示，否则显示为空。

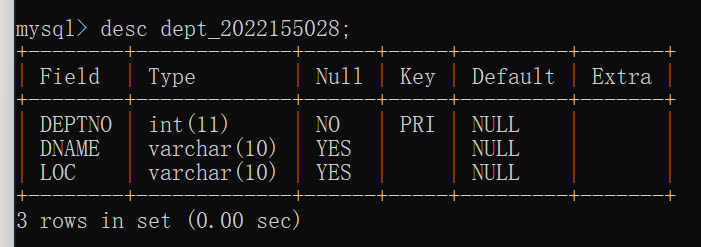
****

图3：创建表dept成功

* + 1. 使用如下命令创建EMP表**。**

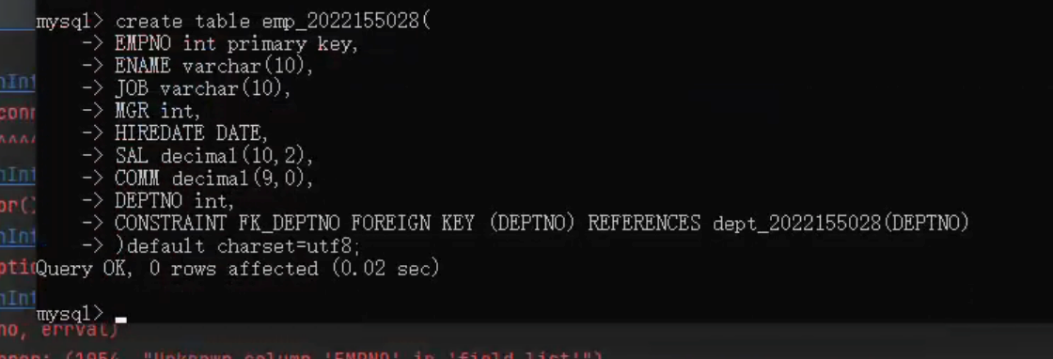
****

图4：创建表emp

* + 1. 检查EMP表是否创建成功（使用命令），如果成功则结果如下图所示，否则显示为空。

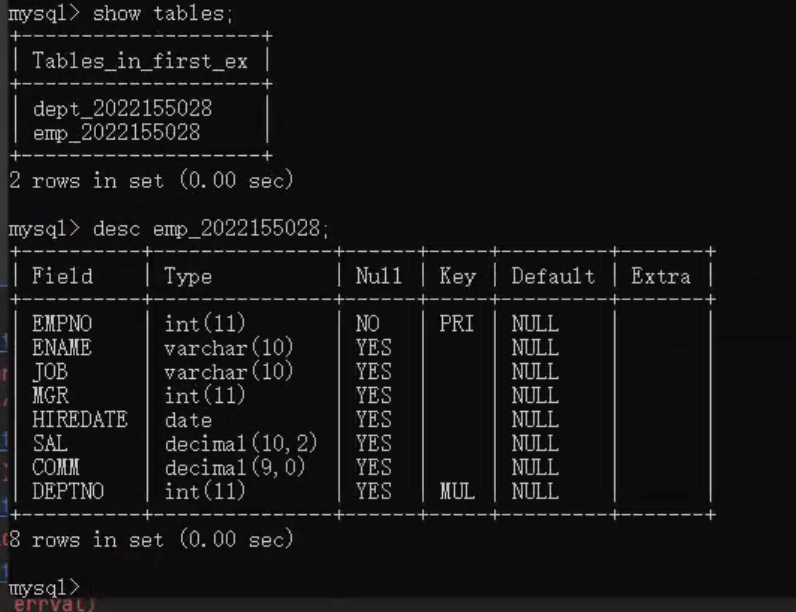
****

图5：创建表emp成功

* + 1. 发现在创建表时忘记将外键DEPTNO设置为非空，因此在这里使用命令将该键设置为非空。

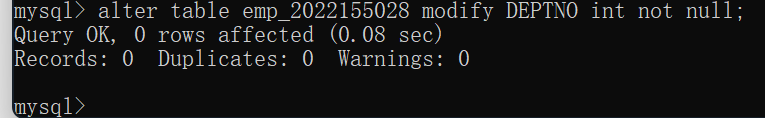
****

图6：修改字段

* + 1. 然后再次使用命令，发现字段已经设置为非空。

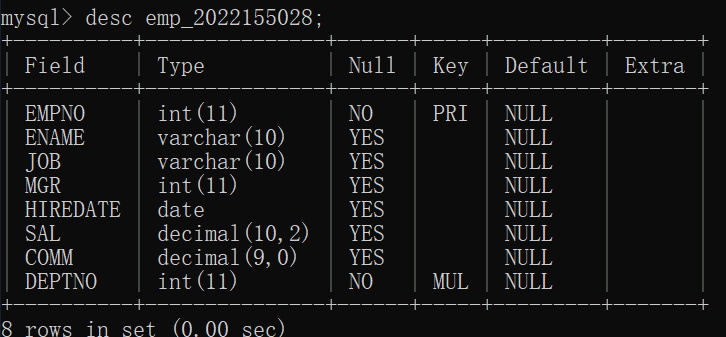
****

图7：修改字段结果

* + 1. 创建表的时候忘记将MGR设置为EMPNO的外键，因此这里使用命令

。

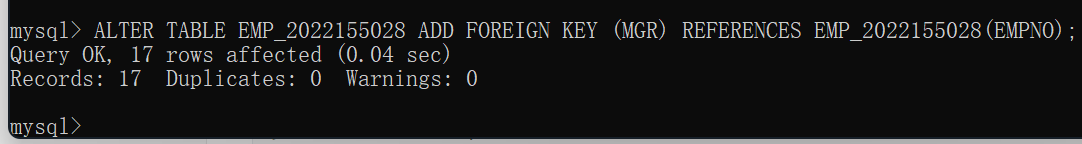
****

图8：修改字段

* + 1. 最终效果展示

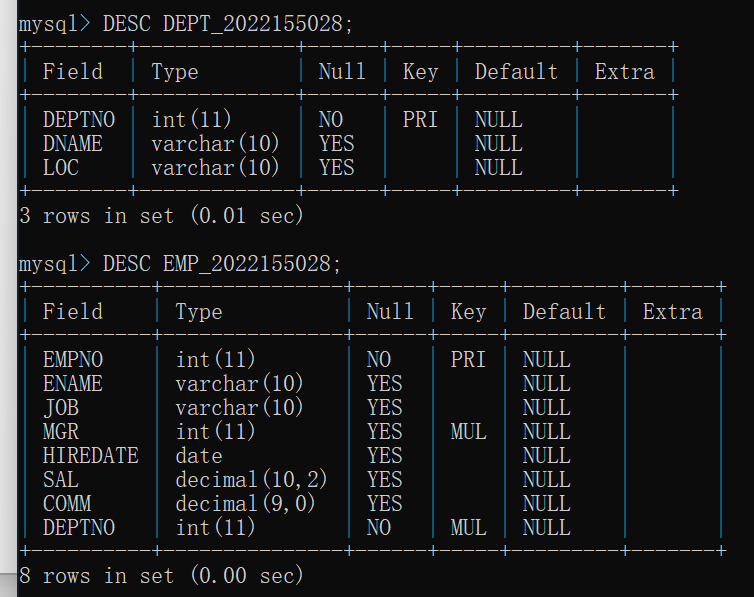


图9：最终效果

1. **插入emp和dept数据的DML语句 （要有语句和运行结果截屏）**
   1. 使用命令*INSERT INTO dept\_2022155028 VALUES (10, 'ACCOUNTING', 'LONDON'), (20, 'RESEARCH', 'PRESTON'), (30, 'SALES', 'LIVERPOOL'), (40, 'OPERATIONS', 'STAFFORD'), (50, 'MARKETING', 'LUTON');*向表中插入数据，然后输入命令查看插入结果（见下图）。

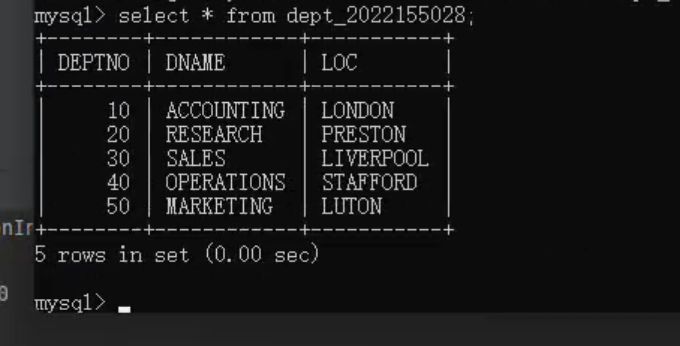
****

图10：表dept\_2022155028插入结果

* 1. 使用命令INSERT INTO emp\_2022155028 VALUE (7369, 'SMITH', 'CLERK', 7902, '1990-12-17', 13750, NULL, 20), (XX,XXX,……,XX);即可向表中插入数据（因为数据太长，这里只以第一条数据的插入作为命令演示，实际上可以像①中多条插入），然后输入命令查看插入结果（见下图）。

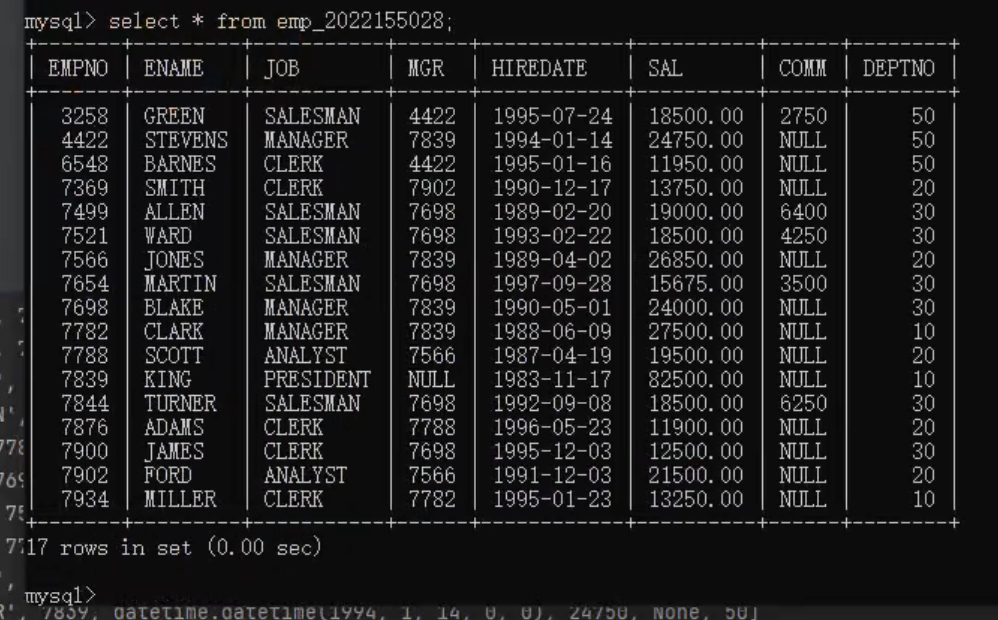
****

图11：emp\_2022155028插入结果

**3．完成实验指导书中对应老师指定的题目.（要有题目语句和运行结果截屏）**

EX1

7. What is the name, job title and employee number of the person in department 20 who earns more than £25000.

输入命令：SELECT ENAME, JOB, EMPNO FROM EMP\_2022155028 WHERE DEPNO = 20 AND SAL > 25000;

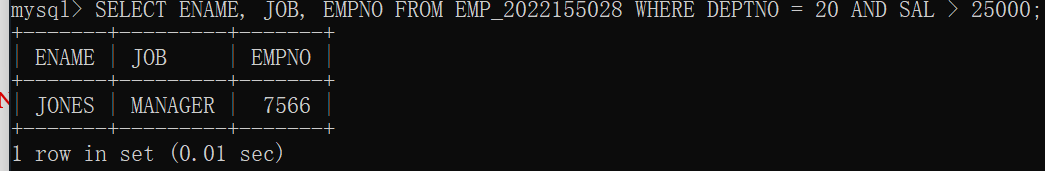


图12：运行结果

9. Find any Clerk who is not in department 10.

输入命令：SELECT \* FROM EMP\_2022155028 WHERE DEPTNO != 10 AND JOB = “CLERK”;

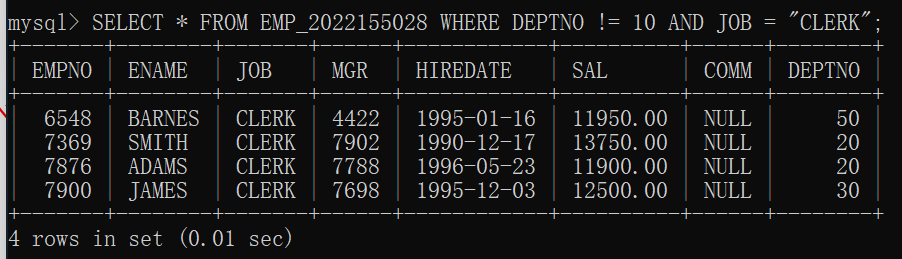


图13：运行结果

12 Find the name of the President.

输入命令：SELECT ENAME FROM EMP\_2022155028 WHERE JOB = “PRESIDENT”;

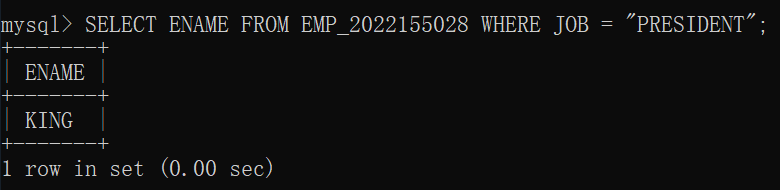


图14：运行结果

14 List the employees whose names have TH or LL in them

输入命令：SELECT \* FROM EMP\_2022155028 WHERE ENAME LIKE “%TH%” OR ENAME LIKE “%LL%”;

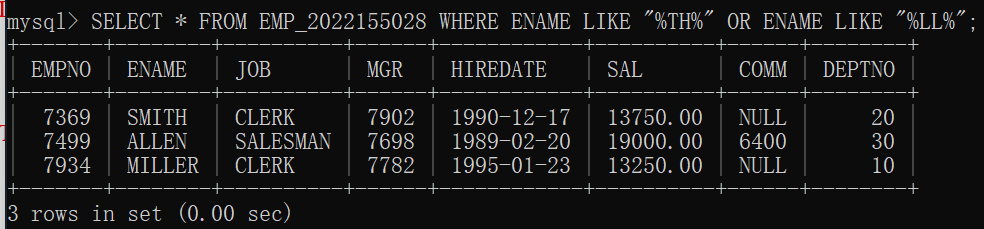


图15：运行结果

17. Find the name, job, salary, hiredate and department number of all employees in ascending order by their salaries.

输入命令：SELECT ENAME, JOB, SAL, HIREDATE, DEPTNO FROM EMP\_2022155028 ORDER BY SAL;

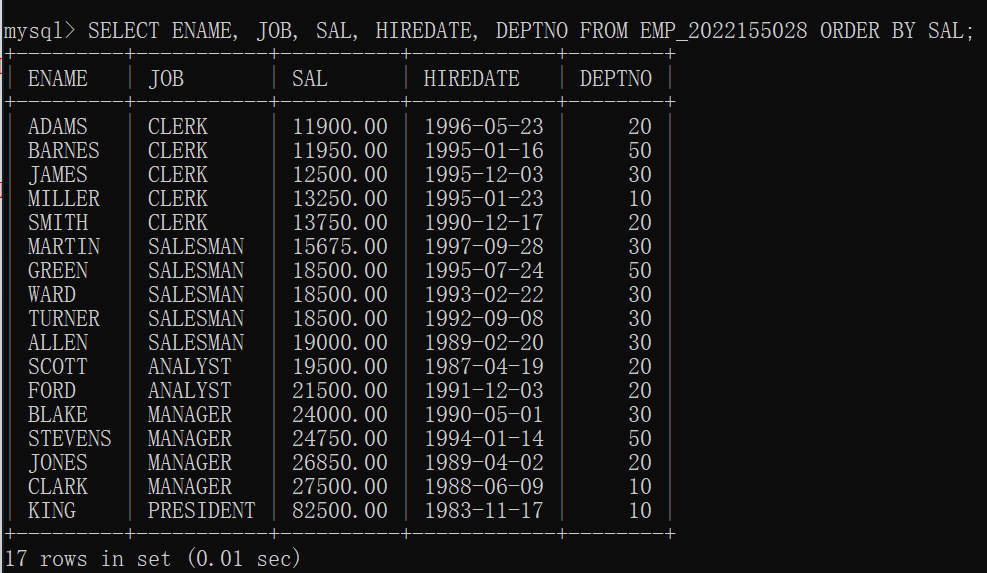


图16：运行结果

18. List all salesmen in descending order by commission divided by their salary.

输入命令：SELECT \* FROM EMP\_2022155028 WHERE JOB = "SALESMAN" ORDER BY (COMM / SAL) DESC;

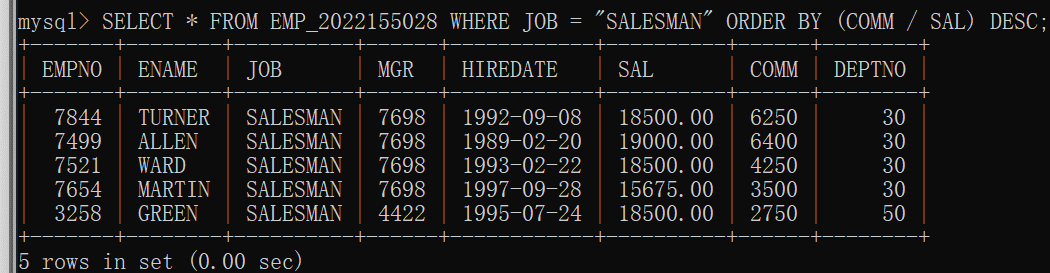


图17：运行结果

19. Order employees in department 30 who receive commision, in ascending order by commission

输入命令：SELECT \* FROM EMP\_2022155028 WHERE DEPTNO = 30 AND COMM IS NOT NULL ORDER BY COALESCE(COMM, 0);

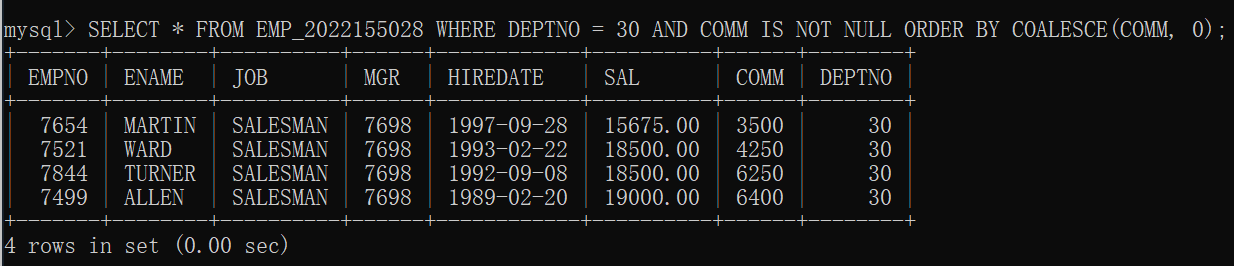


图18：运行结果

20 Find the names, jobs, salaries and commissions of all employees who do not have managers.

输入命令：SELECT ENAME, JOB, SAL, COMM FROM EMP\_2022155028 WHERE MGR IS NULL;

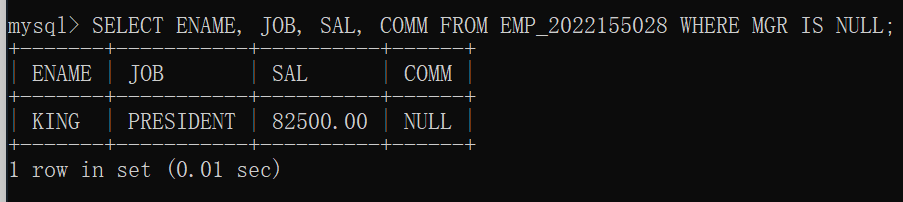


图19：运行结果

EX2

3. List the names of all salesmen who work in SALES

输入命令：SELECT ENAME FROM EMP\_2022155028, DEPT\_2022155028 WHERE EMP\_2022155028.DEPTNO = DEPT\_2022155028.DEPTNO AND DEPT\_2022155028.DNAME = “SALES”;

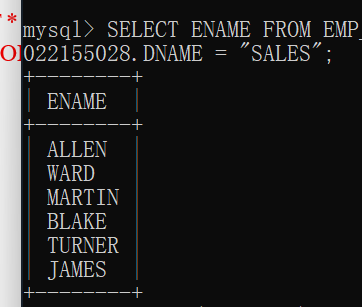


图20：运行结果

4. List all departments that do not have any employees.

输入命令；SELECT DNAME FROM DEPT\_2022155028 WHERE DEPT\_2022155028.DEMPTNO NOT IN (SELECT DEPTNO FROM EMP\_2022155028);

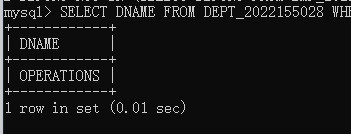


图21：运行结果

5 For each employee whose salary exceeds his manager's salary, list the employee's name and salary and the manager's name and salary.

输入命令；SELECT WORKER.ENAME, WORKER.SAL, MANAGER.ENAME, MANAGER.SAL FROM EMP\_2022155028 WORKER, EMP\_2022155028 MANAGER WHERE WORKER.MGR = MANAGER.EMPNO AND WORKER.SAL > MANAGER.SAL;

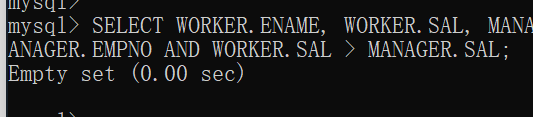


图22：运行结果

6. List the employees who have BLAKE as their manager.

输入命令；SELECT WORKER.\* FROM EMP\_2022155028 WORKER, EMP\_2022155028 MANAGER WHERE WORKER.MGR = MANAGER.EMPNO AND MANAGER.ENAME = "BLAKE";

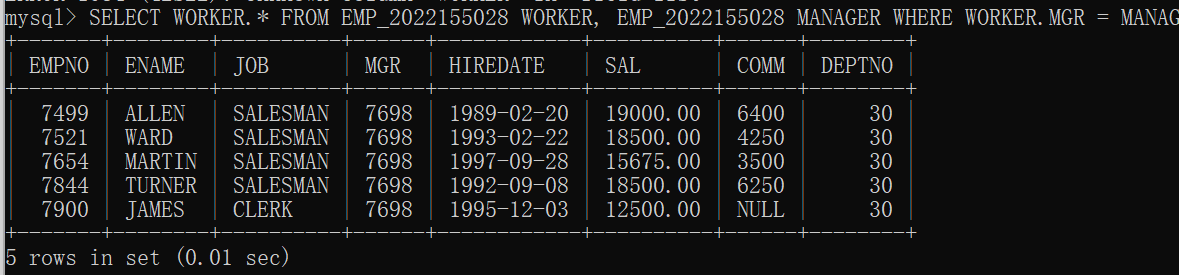


图23：运行结果

开放性题目：索引的创建与优化分析

**1. 你需要设计一个优化查询来找到在指定年份（比如2000年）之后入职的所有员工的姓名和工资。请根据查询的需求设计一个索引，来优化如下查询的执行效率。使用EXPLAIN 分析输出显示索引的有效性。**

使用命令：*EXPLAIN SELECT ENAME, SAL FROM EMP\_2022155028 WHERE HIREDATE > "2000-01-01";*以及使用命令：*EXPLAIN SELECT ENAME, SAL FROM EMP\_2022155028 WHERE HIREDATE > "1990-01-01";*进行测试。

不建立索引的测试结果如下：可以看到，无论查询的数据是否在表中存在，在没有索引的情况下均会遍历整个表的17行数据。

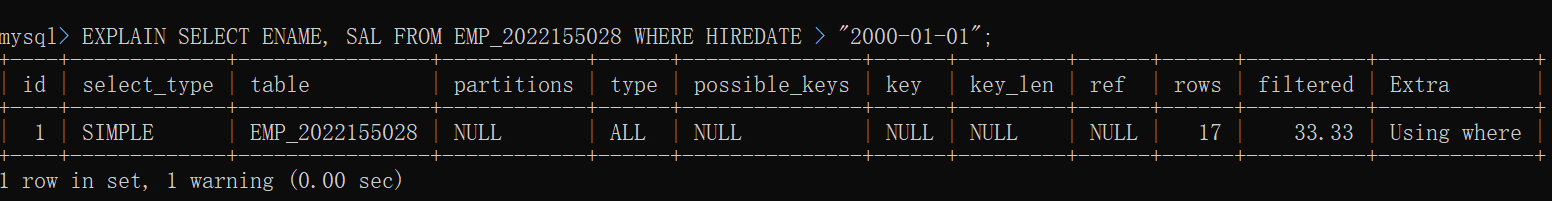


图24：测试结果（1）

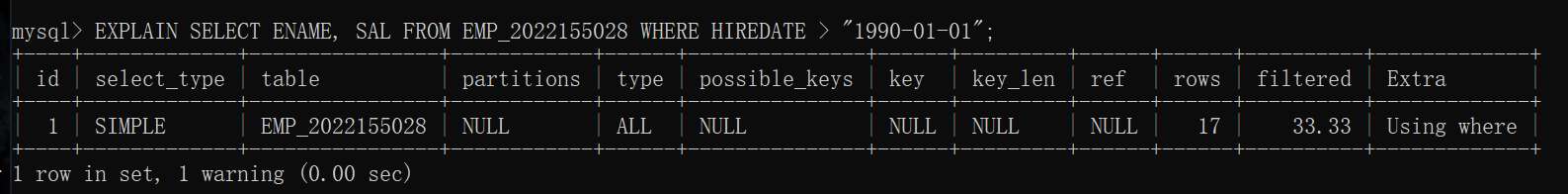


图25：测试结果（2）

因为题目要求涉及查询年份、员工姓名和工资，所以使用命令：*CREATE INDEX YEAR ON EMP\_2022155028(HIREDATE, ENAME, SAL);*建立索引。

建立索引的测试结果如下：可以看到在使用第一条命令的情况下只需要探测一行数据即可，相比于建立索引前的17行，探测量大大减少，由此证明索引的有效性；即使是第二条命令也只需要探测12行数据，相比于17行，探测量也下降了不少，由此也可以证明索引的有效性。

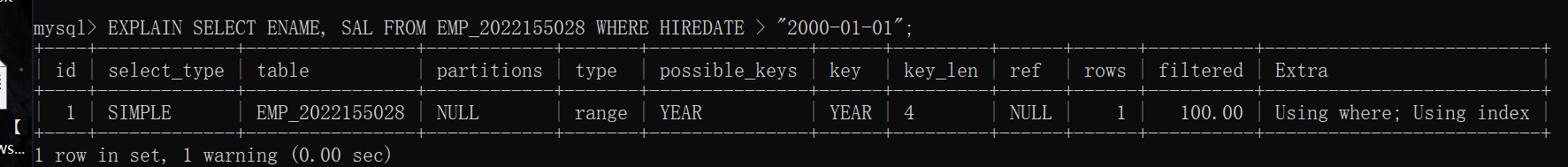


图26：测试结果（1）

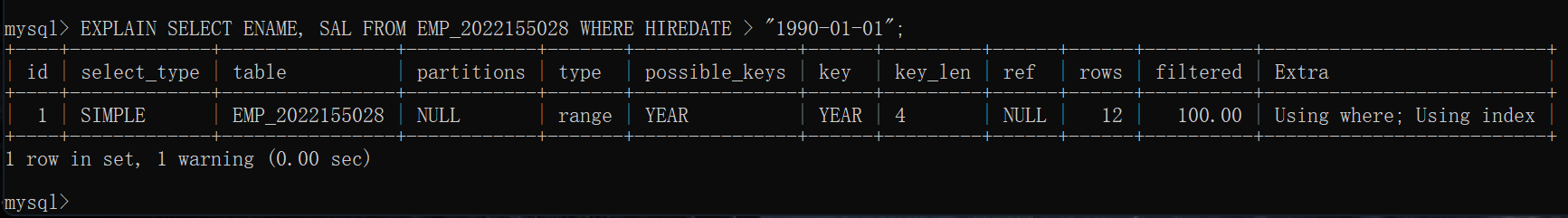


图27：测试结果（2）

**2. 索引回表（Index Back to Table）是 MySQL 中常见的一个概念，通常发生在使用非聚簇索引（即辅助索引）的查询过程中。当查询通过索引检索到相关的索引列数据后，还需要回到数据表中查找非索引列的数据，这一过程就被称为回表。**

**这种回表的操作增加了 IO 操作，尤其是在大量查询数据时，会对性能产生一定的影响。因此，设计合理的索引，可以避免或减少回表操作，提升查询效率。请对此问题，设计相关索引并说明解决方案。**

1. 设置覆盖索引。设置覆盖索引即是指索引包含所有查询所需的列，这样查询可以直接通过索引来获取数据，而不需要回表。

解决方案：如果查询经常访问某些列，而这些列已经存在于某个索引中，那么可以创建一个包含这些列的复合索引。

1. 根据索引的选择性排列索引列顺序。在复合索引中，MySQL 会从左到右使用索引中的列，所以最左边的列应该具有最高的选择性。

解决方案：根据查询模式调整复合索引中列的顺序，使得选择性高的列排在前面。

1. 利用索引合并策略：MySQL 可以使用多个索引来优化查。

解决方案：设计索引时，考虑可能的查询模式，以便 MySQL 可以有效地使用多个索引。

1. 利用索引的选择性：选择性高的索引可以更有效地过滤数据。选择性是指索引列中唯一值的数量与表中行数的比例。

解决方案：选择那些具有高选择性的列来创建索引，这样可以减少查询时需要检查的行数，从而减少回表的可能性。

1. 避免使用函数，因为函数会阻止MySQL使用索引。

解决方案：1）避免在 WHERE 子句中使用函数或表达式，或者创建计算列并为其创建索引。2）将MySQL版本升级为8.x，因为最新版本的MySQL支持使用函数索引。

**五.实验心得**

1. 通过这次数据库系统的实验，我对SQL的DDL语言DML语言有了更深入的了解和实践。
2. 在数据插入和查询的过程中，我体会到了SQL语言的强大和灵活性。通过编写INSERT INTO语句，我成功地向数据库表中插入了数据，而SELECT语句则让我能够根据需要检索数据。此外，我还学习了如何使用ORDER BY子句对查询结果进行排序，以及如何使用LIKE子句进行模式匹配查询。
3. 通过EXPLAIN命令，我观察到了索引对查询效率的影响，这让我认识到了索引在数据库性能优化中的关键作用。此外，我也学习了如何设计合理的索引，以减少查询过程中的回表操作，提高查询效率。

|  |
| --- |
| 指导教师批阅意见：  成绩评定：  指导教师签字：  年 月 日 |
| 备注： |