

**数据库系统**

**实 验 报 告**

学 号：2191110323

学生姓名：张凯文

班 级：1911103

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 计分项 | 出勤 | 程序验收 | 实验报告 | 综合 |
| 成绩 |  |  |  |  |
| 评语 | | | | |
|  | | | | |

**2021-04**

**数据库系统 实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 张凯文 | | 院系 | 软件学院 | | 学号 | 2191110323 |
| 任课教师 | | 王金宝 | | | 指导教师 | 王金宝 | |
| 实验地点 | | 研究院二楼 | | | 实验时间 | 2021年4月24日 | |
| 实验名称 | | 实验一：关系数据库管理系统及SQL语言的使用 | | | | | |
| 实验环境 | | 阿里云开放实验室提供的RDS-MySQL或者PolarDB-MySQL。 | | | | | |
| 实验目标 | | | | | | | |
| 掌握关系数据库管理系统的基本命令，并熟练使用SQL语言管理数据库（阿里云数据库RDS-MySQL或者PolarDB-MySQL）。掌握SQL语言的使用方法，学会使用SQL语言进行关系数据库查询，特别是聚集查询、连接查询和嵌套查询。 | | | | | | | |
| 实验内容（实验过程和相应的实验结果展示） | | | | | | | |
| **创建关系数据库COMPANY，使用SQL语言完成如下查询：**  1：参加了项目名为“SQL Project”的员工名字；  SELECT ENAME  FROM `employee` WHERE ESSN IN  ( SELECT ESSN FROM `works\_on` WHERE PNO IN  ( SELECT PNO  FROM `project` WHERE PNAME = 'SQL Project'));  2：在“Research Department”工作且工资低于3000元的员工名字和地址；  SELECT ENAME,ADDRESS FROM `e2\_employee`  WHERE SALARY < 3000 AND DNO IN (  SELECT DNO  FROM `e2\_department` WHERE  DNAME = 'Research Department') ;  3：没有参加项目编号为P1的项目的员工姓名；  SELECT ENAME FROM `e2\_employee` WHERE ESSN NOT IN  ( SELECT ESSN FROM `e2\_works\_on` WHERE  PNO = 'P1');  4：由张红领导的工作人员的姓名和所在部门的名字；  SELECT ENAME, DNAME FROM `e2\_employee` JOIN `e2\_department` USING(DNO)  WHERE `ESSN` IN( SELECT `ESSN` FROM `e2\_works\_on` WHERE `PNO` IN  ( SELECT `PNO` FROM `e2\_project` WHERE `DNO` IN  ( SELECT `DNO` FROM `e2\_department` WHERE `MGRSSN` IN  ( SELECT `ESSN` FROM `e2\_employee` WHERE `ENAME` = "张红"))));  5：至少参加了项目编号为P1和P2的项目的员工号；  SELECT ESSN FROM `e2\_works\_on` WHERE PNO = 'P1' AND ESSN IN  ( SELECT ESSN FROM `e2\_works\_on` WHERE PNO = 'P2');  6：参加了全部项目的员工号码和姓名；  SELECT ESSN, ENAME FROM `e2\_employee` WHERE NOT EXISTS  ( SELECT PNO FROM `e2\_project` WHERE NOT EXISTS  (SELECT \* FROM `e2\_works\_on` WHERE `e2\_employee`.ESSN = `e2\_works\_on`.ESSN  AND `e2\_project`.PNO = `e2\_works\_on`.PNO));  7：员工平均工资低于3000元的部门名称；  SELECT DNAME FROM `e2\_department` JOIN `e2\_employee` ON (`e2\_department`.DNO = `e2\_employee`.DNO)  GROUP BY DNAME  HAVING 3000 >= avg(SALARY);  8：至少参与了3个项目且每周项目工作总时间不少于20小时的员工名字；  SELECT ENAME FROM `e2\_employee` WHERE ESSN IN  (SELECT ESSN FROM `e2\_works\_on` WHERE `e2\_employee`.ESSN = `e2\_works\_on`.ESSN  GROUP BY ESSN  HAVING COUNT( \*) >3  AND SUM(HOURS\_PER\_WEEK) >=20);  9：假设每个员工每周工作40个小时，查询每个部门的员工小时平均工资；  SELECT avg(SALARY)/40 SALARY\_PER\_HOURS FROM `e2\_employee`  GROUP BY DNO;  10：参加了由张红领导的部门所属的全部项目的员工姓名；  SELECT ENAME FROM `e2\_employee` A  WHERE NOT EXISTS  ( SELECT PNO FROM `e2\_project` B WHERE DNO IN  (SELECT DNO  FROM `e2\_department` WHERE MGRSSN IN  ( SELECT ESSN FROM `e2\_employee` WHERE ENAME = '张红'  ))  AND NOT EXISTS  ( SELECT \*  FROM `e2\_works\_on` WHERE  A.ESSN = `e2\_works\_on`.ESSN  AND B.PNO = `e2\_works\_on`.PNO  ));  11：没有参加过任何由张红领导的部门所属项目的员工姓名；  SELECT ENAME FROM `e2\_employee` A WHERE NOT EXISTS  ( SELECT PNO FROM `e2\_project` B WHERE DNO IN  (SELECT DNO FROM `e2\_department` WHERE MGRSSN IN  ( SELECT ESSN FROM `e2\_employee` WHERE ENAME = '张红'  ))  AND EXISTS  ( SELECT \*  FROM `e2\_works\_on` WHERE  A.ESSN = `e2\_works\_on`.ESSN  AND B.PNO = `e2\_works\_on`.PNO  ));  12：领导项目数量最多的部门负责人姓名；  SELECT ENAME FROM `e2\_employee` WHERE ESSN IN  ( SELECT MGRSSN FROM `e2\_department` WHERE DNO IN  (SELECT DNO FROM `e2\_project`  WHERE `e2\_project`.DNO = `e2\_department`.DNO  GROUP BY DNO  HAVING COUNT( `e2\_project`.PNO) >= ALL (  SELECT COUNT( \*)  FROM `e2\_project`  GROUP BY DNO )));  13：参与项目最多的员工的姓名；  SELECT ENAME FROM `e2\_employee` WHERE ESSN IN  ( SELECT ESSN FROM `e2\_works\_on`  WHERE `e2\_employee`.ESSN = `e2\_works\_on`.ESSN  GROUP BY ESSN  HAVING COUNT(\*) >= ALL (  SELECT COUNT( \*) FROM `e2\_works\_on`  GROUP BY ESSN ));  14：工资总数最高的部门信息；  SELECT \* FROM `department` WHERE DNO IN  ( SELECT DNO FROM `employee`  WHERE `employee`.DNO = `department`.DNO  GROUP BY DNO  HAVING SUM(SALARY) >= ALL (  SELECT SUM(SALARY) FROM `employee`  GROUP BY DNO ));  15：假设每个员工每周工作40个小时，查询每个项目一周的人工费用；  SELECT PNO, (avg(SALARY)/40)\*SUM(`HOURS\_PER\_WEEK`) SALARY\_PER\_WEEK  FROM `employee` JOIN `works\_on` ON ( `employee`.ESSN = `works\_on`.ESSN)  GROUP BY PNO;  **创建触发器，完成以下功能：**  1：修改WORKS\_ON中每周工作时间后，检查员工每周工作总时长是否超过40小时，如果是，则撤销修改；      2：插入新的员工后，如果员工的直接领导信息为空而且员工已分配至某部门，则将员工所属部门的领导作为员工的直接领导；    创建视图，完成以下功能：   1. 为张红创建她所属部门的员工信息视图EMPLOYEE\_ZHONGHONG；   Select `e2\_employee`.`ENAME` AS `ENAME`,`e2\_employee`.`ESSN` AS `ESSN`,`e2\_employee`.`ADDRESS` AS `ADDRESS`,`e2\_employee`.`SALARY` AS `SALARY`,`e2\_employee`.`SUPERSSN` AS `SUPERSSN`,`e2\_employee`.`DNO` AS `DNO` from `e2\_employee` where (`e2\_employee`.`SUPERSSN` = (select `e2\_employee`.`ESSN` from `e2\_employee` where (`e2\_employee`.`ENAME` = '张红')))    2）在EMPLOYEE\_ZHONGHONG中查询工资超过4000的员工信息；  SELECT \* FROM `employee\_zhonghong`  WHERE SALARY> 4000 | | | | | | | |
| 实验结论（结果分析、遇到的困难和解决方法等） | | | | | | 备注 |  |
| 1. 建立触发器进行操作时，触发器可以进行几个表之间的嵌套操作，使插入、更新等操作符合用户需求，省去管理员在多表之间操作。 2. 触发器更新操作时应该是BEFORE而不是AFTER。 | | | | | | | |

**数据库系统 实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 张凯文 | | 院系 | 软件学院 | | 学号 | 2191110323 |
| 任课教师 | | 王金宝 | | | 指导教师 | 王金宝 | |
| 实验地点 | | 研究院二楼 | | | 实验时间 | 2021年 5月 8日 | |
| 实验名称 | | 实验二：使用高级语言操纵阿里云数据库 | | | | | |
| 实验环境 | | 阿里云数据库、Java、C++等 | | | | | |
| 实验目标 | | | | | | | |
| 学会使用高级语言访问阿里云数据库数据库，并进行查询。 | | | | | | | |
| 实验内容（实验过程和相应的实验结果展示） | | | | | | | |
| 1.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的参加项目号员工的项目号："); String PNO = in.nextLine(); PreparedStatement ps1 = connection.prepareStatement("SELECT ENAME FROM `employee`\n" + "WHERE ESSN IN \n" + "(SELECT ESSN FROM `works\_on` WHERE PNO = ?);");  ps1.setString(1,PNO);  resultSet = ps1.executeQuery();  while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME"));  图1 图2  2.  Scanner in = new Scanner(System.in);  System.out.println("请输入你要查询的参加项目员工的项目名：");  String PNAME = in.nextLine();  PreparedStatement ps2 = connection.prepareStatement("SELECT ENAME FROM `employee`" + "WHERE ESSN IN \n" + "(SELECT ESSN FROM `works\_on` WHERE `PNO` IN ( SELECT `PNO` FROM `project` WHERE `PNAME` = ?));"); ps2.setString(1,PNAME);  resultSet = ps2.executeQuery();  while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME")); }  3.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的在部门工作的部门名："); String DNAME = in.nextLine(); PreparedStatement ps3 = connection.prepareStatement("SELECT ENAME, ADDRESS FROM `employee` WHERE `DNO` IN \n" +  "( SELECT `DNO` FROM `department` WHERE `DNAME` = ?);\n"); ps3.setString(1,DNAME); resultSet = ps3.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME"));  System.out.println(resultSet.getString("ADDRESS")); }  图3  图4  4.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的在部门工作的部门名："); String DNAME = in.nextLine(); System.out.println("请输入你要查询者工资的上限："); int SALARY = in.nextInt(); PreparedStatement ps4 = connection.prepareStatement("SELECT ENAME, ADDRESS FROM `employee` WHERE `DNO` IN \n" +  "( SELECT `DNO` FROM `department` WHERE `DNAME` = ? AND `SALARY` < ? );\n"); ps4.setString(1,DNAME); ps4.setInt(2,SALARY); resultSet = ps4.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME"));  System.out.println(resultSet.getString("ADDRESS")); }  5.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的不在项目工作员工的项目名："); String PNO = in.nextLine(); PreparedStatement ps5 = connection.prepareStatement("SELECT ENAME FROM `employee` \n" +  "WHERE `ESSN` NOT IN \n" +  "( SELECT `ESSN` FROM `works\_on` WHERE `PNO` = ?);"); ps5.setString(1,PNO); resultSet = ps5.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME")); }  图5  图6  6.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的领导工作员工的领导名："); String ENAME = in.nextLine(); PreparedStatement ps6 = connection.prepareStatement("SELECT ENAME, DNAME FROM `employee` JOIN `department` USING(DNO)\n" +  "WHERE `ESSN` IN( SELECT `ESSN` FROM `works\_on` WHERE `PNO` IN \n" +  " ( SELECT `PNO` FROM `project` WHERE `DNO` IN \n" +  " ( SELECT `DNO` FROM `department` WHERE `MGRSSN` IN \n" +  " ( SELECT `ESSN` FROM `employee` WHERE `ENAME` = ?)))); "); ps6.setString(1,ENAME); resultSet = ps6.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME"));  System.out.println(resultSet.getString("DNAME")); }  7.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询的参加项目的项目名1："); String PNO1 = in.nextLine(); System.out.println("请输入你要查询的参加项目的项目名2："); String PNO2 = in.nextLine(); PreparedStatement ps7 = connection.prepareStatement("SELECT ESSN FROM `works\_on` WHERE `PNO` = ?\n" +  "AND `ESSN` IN \n" +  "( SELECT `ESSN` FROM `works\_on` WHERE `PNO` = ?)\n"); ps7.setString(1,PNO1); ps7.setString(2,PNO2); resultSet = ps7.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ESSN")); }  图7  图8  8.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询平均部门工资上限："); int SALARY = in.nextInt(); PreparedStatement ps8 = connection.prepareStatement("SELECT DNAME FROM `department` WHERE `DNO` IN \n" +  "( SELECT `DNO` FROM `employee` GROUP BY `DNO` HAVING avg( `SALARY`) < ?); "); ps8.setInt(1,SALARY); resultSet = ps8.executeQuery(); while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("DNAME")); }  图9  9.  Scanner in = new Scanner(System.in); System.out.println("请输入你要查询员工至少参加项目数："); int COUNT\_P = in.nextInt(); System.out.println("请输入你要查询员工工作时间上限："); int TIME = in.nextInt(); PreparedStatement ps9 = connection.prepareStatement("SELECT ENAME FROM `employee` WHERE `ESSN` IN \n" +  "( SELECT `ESSN` FROM `works\_on` GROUP BY `ESSN` HAVING COUNT( `PNO`) > ? AND SUM( `HOURS\_PER\_WEEK`) < ?);");  ps9.setInt(1,COUNT\_P);  ps9.setInt(2,TIME);  resultSet = ps9.executeQuery();  while (resultSet.next()) {  // 输出courses表中的name字段  System.out.println(resultSet.getString("ENAME"));  } | | | | | | | |
| 实验结论（结果分析、遇到的困难和解决方法等） | | | | | | 备注 |  |
| 1. 不了解关于向其中函数传参的具体实现方式，后根据上网查询方式了解了如何具体实现 2. 数据库连接方式不了解，后根据实验报告指导成功实现数据库连接 | | | | | | | |

**数据库系统 实验报告**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 姓名 | 张凯文 | | 院系 | 软件学院 | | 学号 | 2191110323 |
| 任课教师 | | 王金宝 | | | 指导教师 | 王金宝 | |
| 实验地点 | | 研究院二楼 | | | 实验时间 | 2021年 5 月 15日 | |
| 实验名称 | | 实验三：基于阿里云数据库的数据库应用系统开发 | | | | | |
| 实验环境 | | 阿里云数据库、Java、C++等 | | | | | |
| 实验目标 | | | | | | | |
| 在熟练掌握MySQL基本命令、SQL语言以及用高级程序语言编写MySQL操作程序的基础上，学习简单数据库系统的设计方法，包括数据库概要设计、逻辑设计。 | | | | | | | |
| 实验内容（实验过程和相应的实验结果展示） | | | | | | | |
| 1. ER图      1. 关系的完整性约束    1. 主键约束      * 1. 外键约束   部门外键约束：    员工外键约束     * 1. 空值约束      1. 索引   部门索引：    部员索引：    仓库索引：    工作信息索引：     1. 视图   所属A工厂的部门视图     1. 系统功能    1. 插入功能   插入前：    插入时：    插入结果：    插入空值：    插入主键重复：     * 1. 删除功能   若删除的表是包含另一个表外键信息    无外键约束删除前：    无外键约束删除后：     * 1. 更新功能   更新前    对D07部门更新时界面    更新后界面     * 1. 查询功能   查询界面    连接查询代码    嵌套查询    分组查询     1. 触发器   当零件未输入细节信息时输入‘无’     1. 界面   部门界面    员工界面    工作信息界面 | | | | | | | |
| 实验结论（结果分析、遇到的困难和解决方法等） | | | | | | 备注 |  |
| 1. 一开始在图形界面具体代码实现方式上不了解，后来上网寻找教学视频，跟着视频一步一步完成实验 2. 对前端界面传参数的具体工作流程不了解，后上网查找后解决 | | | | | | | |