**实验报告**

**实 验 名 称：**

**基于校运动会的数据库方案设计与实现**

|  |
| --- |
| **实验概述：** |
| **【实验目的及要求】**  **实验目的：**  综合运用课程所学的知识点进行具体背景的数据库方案设计和实现  **实验要求：**  **完成如下设计和实现内容：**  **1、设计部分**  项目背景介绍  ER图设计（概念模型）  数据库库表设计（表头，逻辑模型）  规范化工作  表的详细设计（属性定义，物理模型）  常见查询归纳  完整性约束方案  视图设计方案  数据库安全方案的设计（可选，有加分）  **2、实现部分**  表格创建（包括约束条件的创建）  数据录入  常见查询的例子演示  视图创建  数据库安全（可选）  **3、总结**  组员分工  难点与不足分析  收获或体会  **【实验环境】**   * Oracle10g 或11g |
| **实验内容：** |
| **1.设计部分** |
| **（1）项目背景介绍** |
|  |
| 1. **ER图设计**   **中文版ER图：** |
| **608e0a25630bbbe03d89b6c785dfecd**  **英文版ER图：**  a9df0125e785cebc96906714ba50072 |
| **（3）数据库库表设计（表头、逻辑模型及规范化）** |
| **RULE（3NF）**   |  |  |  | | --- | --- | --- | | RU\_NO | RU\_NAME | CNO |   **COMPETITION（3NF）**   |  |  |  |  |  |  |  | | --- | --- | --- | --- | --- | --- | --- | | CNO | CNAME | SEX | COMPETITION\_TYPE | FNO | ENO | TNO |   **REFEREE\_COMPETITION（3NF）**   |  |  | | --- | --- | | RE\_NO | CNO |   **EVENT\_CATEGORY（3NF）**   |  |  | | --- | --- | | ENO | ENAME |   **REFEREE（3NF）**   |  |  |  |  | | --- | --- | --- | --- | | RE\_NO | RE\_NAME | SEX | PNO |   **FIELD（3NF）**   |  |  | | --- | --- | | FNO | FNAME |   **POST（3NF）**   |  |  | | --- | --- | | PNO | PNAME |   **INDIVIDUAL\_COMPETITION（3NF）**   |  |  | | --- | --- | | CNO | LIMITED\_NUMBER |   **COLLEGE\_TEAM（3NF）**   |  |  |  |  | | --- | --- | --- | --- | | CT\_NO | CL\_NO | CT\_NAME | CNO |   **INDIVIDUAL\_PARTICIPATION（3NF）**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | ANO | CNO | GRADE | RANK | ORDER |   **TEAM\_ATHLETE（3NF）**   |  |  | | --- | --- | | CT\_NO | ANO |   **ATHLETE（3NF）**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | ANO | ANAME | ID | SEX | CL\_NO |   **COLLEGE（3NF）**   |  |  | | --- | --- | | CL\_NO | CL\_NAME |   **TEAM\_PARTICIPATION（3NF）**   |  |  |  |  |  | | --- | --- | --- | --- | --- | | CT\_NO | CNO | GRADE | RANK | ORDER |   **TEAM\_COMPETITION（3NF）**   |  |  | | --- | --- | | CNO | TEAM\_NUMBER |   **TIME（3NF）**   |  |  | | --- | --- | | TNO | SPECIFIC\_TIME | |
| **（4）表的详细设计（属性定义，物理模型）** |
| |  |  |  |  | | --- | --- | --- | --- | | **RULE** | **含义** | **数据类型** | **完整性约束** | | RU\_NO | 规则编号 | 长度为4的可变长度字符串 | 主键 | | RU\_NAME | 规则名称 | 长度为30的可变长度字符串 | 非空 | | CNO | 项目编号 | 长度为4的可变长度字符串 | 外键 | | 主键：RU\_NO | | | | | 外键CNO（被参照关系是COMPETITION） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **EVENT\_CATEGORY** | **含义** | **数据类型** | **完整性约束** | | ENO | 赛事类别编号 | 长度为4的可变长度字符串 | 主键 | | ENAME | 赛事类别名称 | 长度为20的可变长度字符串 | 非空 | | 主键：ENO | | | |  |  |  |  |  | | --- | --- | --- | --- | | **FIELD** | **含义** | **数据类型** | **完整性约束** | | FNO | 场地编号 | 长度为4的可变长度字符串 | 主键 | | FNAME | 场地名称 | 长度为10的可变长度字符串 | 非空 | | 主键：FNO | | | |  |  |  |  |  | | --- | --- | --- | --- | | **POST** | **含义** | **数据类型** | **完整性约束** | | PNO | 职务编号 | 长度为4的可变长度字符串 | 主键 | | PNAME | 职务名称 | 长度为20的可变长度字符串 | 非空 | | 主键：PNO | | | |  |  |  |  |  | | --- | --- | --- | --- | | **REFEREE** | **含义** | **数据类型** | **完整性约束** | | RE\_NO | 裁判编号 | 长度为4的可变长度字符串 | 主键 | | RE\_NAME | 裁判名称 | 长度为10的可变长度字符串 | 非空 | | SEX | 性别 | 长度为10的可变长度字符串 | “男”或“女” | | PNO | 职务编号 | 长度为4的可变长度字符串 | 外键 | | 主键：RE\_NO | | | | | 外键PNO（被参照关系是POST） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **REFEREE\_COMPITITION** | **含义** | **数据类型** | **完整性约束** | | RE\_NO | 裁判编号 | 长度为4的可变长度字符串 | 主键、外键 | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键、外键 | | 主键：RE\_NO,CNO | | | | | 外键RE\_NO（被参考关系是REFEREE）,外键CNO（被参考关系是COMPETITION） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **COMPETITION** | **含义** | **数据类型** | **完整性约束** | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键 | | CNAME | 项目名称 | 长度为20的可变长度字符串 | 非空 | | SEX | 参赛者性别 | 长度为10的可变长度字符串 | “男”或“女” | | COMPETITION\_TYPE | 项目类型 | 长度为10的可变长度字符串 | 非空 | | FNO | 场地编号 | 长度为4的可变长度字符串 | 外键 | | ENO | 赛事类别编号 | 长度为4的可变长度字符串 | 外键 | | TNO | 时间编号 | 长度为4的可变长度字符串 | 外键 | | 主键：CNO | | | | | 外键ENO(被参照关系是EVENT \_CATEGORY),外键TNO(被参照关系是TIME),外键FNO(被参照关系是FIELD) | | | |  |  |  |  |  | | --- | --- | --- | --- | | **TEAM\_COMPETITION** | **含义** | **数据类型** | **完整性约束** | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键、外键 | | TEAM\_NUMBER | 团体人数 | 整型 | >0 | | 主键：CNO | | | | | 外键CNO（被参考关系是COMPETITON） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **TEAM\_PARTICIPATION** | **含义** | **数据类型** | **完整性约束** | | CT\_NO | 队伍编号 | 长度为4的可变长度字符串 | 主键、外键 | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键、外键 | | GRADE | 成绩 | 长度为20的可变长度字符串 |  | | RANK | 名次 | 长度为2的可变长度字符串 | 非空 | | ORDER | 参赛顺序 | 长度为2的可变长度字符串 | 非空 | | 主键：CT\_NO,CNO | | | | | 外键CT\_NO（被参考关系是COLLEGE\_TEAM）,外键CNO（被参考关系是COMPETITION） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **COLLEGE** | **含义** | **数据类型** | **完整性约束** | | CL\_NO | 学院编号 | 长度为4的可变长度字符串 | 主键 | | CL\_NAME | 学院名称 | 长度为10的可变长度字符串 | 非空 | | 主键：CL\_NO | | | |  |  |  |  |  | | --- | --- | --- | --- | | **ATHLETE** | **含义** | **数据类型** | **完整性约束** | | ANO | 运动员编号 | 长度为4的可变长度字符串 | 主键 | | ANAME | 姓名 | 长度为10的可变长度字符串 | 非空 | | ID | 学号 | 长度为10的可变长度字符串 | 非空 | | SEX | 性别 | 长度为10的可变长度字符串 | “男”或“女” | | CL\_NO | 学院编号 | 长度为4的可变长度字符串 | 外键 | | 主键：ANO | | | | | 外键CL\_NO(被参照关系：COLLEGE) | | | |  |  |  |  |  | | --- | --- | --- | --- | | **COLLEGE\_TEAM** | **含义** | **数据类型** | **完整性约束** | | CT\_NO | 队伍编号 | 长度为4的可变长度字符串 | 主键 | | CT\_NAME | 队伍名称 | 长度为30的可变长度字符串 | 非空 | | CL\_NO | 学院编号 | 长度为4的可变长度字符串 | 外键 | | CNO | 项目编号 | 长度为4的可变长度字符串 | 外键 | | 主键：CT\_NO | | | | | 外键CL\_NO(被参照关系：COLLEGE),外键CNO(被参照关系：TEAM\_COMPETITION) | | | | |  | | | |  |  |  |  |  | | --- | --- | --- | --- | | **TEAM\_ATHLETE** | **含义** | **数据类型** | **完整性约束** | | CT\_NO | 队伍编号 | 长度为4的可变长度字符串 | 主键、外键 | | ANO | 运动员编号 | 长度为4的可变长度字符串 | 主键、外键 | | 主键：CT\_NO,ANO | | | | | 外键CT\_NO（被参考关系是COLLEGE\_TEAM），外键ANO（被参考关系是ATHLETE） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **INDIVIDUAL\_COMPETITION** | **含义** | **数据类型** | **完整性约束** | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键、外键 | | LIMITED\_NUMBER | 学院限制人数 | 整型 | >0 | | 主键：CNO | | | | | 外键CNO（被参考关系是COMPETITION） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **INDIVIDUAL\_PARTICIPATION** | **含义** | **数据类型** | **完整性约束** | | ANO | 运动员编号 | 长度为4的可变长度字符串 | 主键、外键 | | CNO | 项目编号 | 长度为4的可变长度字符串 | 主键、外键 | | GRADE | 成绩 | 长度为20的可变长度字符串 |  | | RANK | 名次 | 长度为2的可变长度字符串 | 非空 | | ORDER | 参赛顺序 | 长度为2的可变长度字符串 | 非空 | | 主键：ANO,CNO | | | | | 外键ANO（被参考关系是ATHLETE）,外键CNO（被参考关系是INDIVIDUAL\_COMPETITION） | | | |  |  |  |  |  | | --- | --- | --- | --- | | **CTIME** | **含义** | **数据类型** | **完整性约束** | | TNO | 时间编号 | 长度为4的可变长度字符串 | 主键 | | SPECIFIC\_TIME | 具体时间 | 长度为20的可变长度字符串 | 非空 | | 主键：TNO | | | | |
| **（5）常见查询归纳** |
| ①信息管理与工程学院的参赛运动员信息  ②男子跳远的比赛结果  ③所有计时裁判员的姓名和编号  ④参赛学院的个数  ⑤获得女子4×100米项目第一名的学院编号和学院名称  ⑥参加男子1500米比赛的运动员编号和参赛顺序  ⑦所有获得个人项目的第一名的运动员信息  ⑧赛事类别名称为“田赛（FIELD\_EVENT）”的项目名称和场地名称  ⑨男子跳远项目的规则编号  ⑩参加男子4\*100接力的运动员学号和姓名  ⑪获得女子800米前三的运动员姓名和学号  ⑫男子跳高比赛的学院限制人数  ⑬赛事类别名称为“(径赛)TRACK\_EVENT”的项目名称、规则编号和场地编号  ⑭男子4×100米接力前三名的学院编号和学院名称  ⑭获得女子3000米第二名的运动员姓名和成绩 |
| **（6）完整性约束方案** |
| CREATE TABLE EVENT\_CATEGORY  (ENO VARCHAR(4) PRIMARY KEY,  ENAME VARCHAR(20) NOT NULL);  CREATE TABLE FIELD  (FNO VARCHAR(4) PRIMARY KEY,  FNAME VARCHAR(10) NOT NULL);  CREATE TABLE CTIME  (TNO VARCHAR(4) PRIMARY KEY,  SPECIFIC\_TIME VARCHAR(20) NOT NULL);  CREATE TABLE COLLEGE  (CL\_NO VARCHAR(4) PRIMARY KEY,  CL\_NAME VARCHAR(30) NOT NULL);  CREATE TABLE POST  (PNO VARCHAR(4) PRIMARY KEY,  PNAME VARCHAR(20) NOT NULL);  CREATE TABLE REFERREE  (RE\_NO VARCHAR(4) PRIMARY KEY,  RE\_NAME VARCHAR(10) NOT NULL,  SEX VARCHAR(10),  PNO VARCHAR(4),  CONSTRAINT FK\_SEX CHECK (SEX IN ('男','女')),  CONSTRAINT FK\_PNO FOREIGN KEY (PNO) REFERENCES POST(PNO));  CREATE TABLE ATHLETE  (ANO VARCHAR(4) PRIMARY KEY,  ANAME VARCHAR(10) NOT NULL,  ID VARCHAR(10),  SEX VARCHAR(10),  CL\_NO VARCHAR(20) NOT NULL,  CONSTRAINT FK\_SEX2 CHECK (SEX IN ('男','女')),  CONSTRAINT FK\_CL\_NO FOREIGN KEY (CL\_NO) REFERENCES COLLEGE(CL\_NO));  CREATE TABLE COMPETITION  (CNO VARCHAR(4) PRIMARY KEY,  CNAME VARCHAR(20) NOT NULL,  SEX VARCHAR(10),  COMPETITION\_TYPE VARCHAR(10) NOT NULL,  FNO VARCHAR(4),  ENO VARCHAR(4),  TNO VARCHAR(4),  CONSTRAINT FK\_SEX1 CHECK (SEX IN ('男','女')),  CONSTRAINT FK\_FNO FOREIGN KEY (FNO) REFERENCES FIELD(FNO),  CONSTRAINT FK\_ENO FOREIGN KEY (ENO) REFERENCES EVENT\_CATEGORY(ENO),  CONSTRAINT FK\_TNO FOREIGN KEY (TNO) REFERENCES CTIME(TNO));  CREATE TABLE RULE  (RU\_NO VARCHAR(4) PRIMARY KEY,  RU\_NAME VARCHAR(30),  CNO VARCHAR(4),  CONSTRAINT FK\_CNO1 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE REFEREE\_COMPETITION  (RE\_NO VARCHAR(4),  CNO VARCHAR(4) ,  PRIMARY KEY(RE\_NO,CNO),  CONSTRAINT FK\_RE\_NO FOREIGN KEY (RE\_NO) REFERENCES REFERREE(RE\_NO),  CONSTRAINT FK\_CNO2 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE COLLEGE\_TEAM  (CT\_NO VARCHAR(4) PRIMARY KEY,  CT\_NAME VARCHAR(30) NOT NULL,  CL\_NO VARCHAR(4),  CNO VARCHAR(4),  CONSTRAINT FK\_CL\_NO2 FOREIGN KEY (CL\_NO) REFERENCES COLLEGE(CL\_NO),  CONSTRAINT FK\_CNO22 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE TEAM\_COMPETITION  (CNO VARCHAR(4) PRIMARY KEY,  TEAM\_NUMBER INT NOT NULL,  CONSTRAINT FK\_CNO3 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE TEAM\_PARTICIPATION  (CT\_NO VARCHAR(4),  CNO VARCHAR(4) ,  GRADE VARCHAR(20),  RANK VARCHAR(2) NOT NULL,  COMPETE\_ORDER VARCHAR(2) NOT NULL,  PRIMARY KEY(CT\_NO,CNO),  CONSTRAINT FK\_CT\_NO FOREIGN KEY (CT\_NO) REFERENCES COLLEGE\_TEAM(CT\_NO),  CONSTRAINT FK\_CNO4 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE TEAM\_ATHLETE  (CT\_NO VARCHAR(20),  ANO VARCHAR(4) ,  PRIMARY KEY(CT\_NO,ANO),  CONSTRAINT FK\_CT\_NO2 FOREIGN KEY (CT\_NO) REFERENCES COLLEGE\_TEAM(CT\_NO),  CONSTRAINT FK\_ANO FOREIGN KEY (ANO) REFERENCES ATHLETE(ANO));  CREATE TABLE INDIVIDUAL\_COMPETITION  (CNO VARCHAR(4) PRIMARY KEY,  LIMITED\_NUMBER VARCHAR(2) NOT NULL,  CONSTRAINT FK\_CNO5 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO));  CREATE TABLE INDIVIDUAL\_PARTICIPATION  (ANO VARCHAR(4) ,  CNO VARCHAR(4) ,  GRADE VARCHAR(20),  RANK VARCHAR(2) NOT NULL,  COMPETE\_ORDER VARCHAR(2) NOT NULL,  PRIMARY KEY(ANO,CNO),  CONSTRAINT FK\_ANO2 FOREIGN KEY (ANO) REFERENCES ATHLETE(ANO),  CONSTRAINT FK\_CNO6 FOREIGN KEY (CNO) REFERENCES COMPETITION(CNO)); |
| **（7）视图设计方案** |
| 1. **余额超过200元用户**   CREATE VIEW BALANCEUP200  AS SELECT ID,ANAME,BALANCE  FROM ACCOUNT  WHERE BALANCE >= 200  ORDER BY ID;   1. **余额低于200元用户**   CREATE VIEW BALANCEUP200  AS SELECT ID,ANAME,BALANCE  FROM ACCOUNT  WHERE BALANCE < 200  ORDER BY ID;   1. **某月至某月的堂食用户**   CREATE VIEW EATIN(ID,CNO,CNAME,CTEL,D\_TYPE)  AS SELECT ID,CNO,CNAME,CTEL,D\_TYPE  FROM USER,ACCOUNT,ORDER,ORDER\_DETAIL,DINING  WHERE D\_TYPE = EATIN AND  ORDER.BUYTIME BETWEEN XX AND XX AND  DINING DNO = ORDER\_DETAIL.DNO AND  ORDER\_DETAIL.ONO = ORDER.ONO AND  ORDER.ID= ACCOUNT.ID AND  ACCOUT.CNO = USER.CNO  ORDER BY ID;   1. **所有提供某种原料的供应商**   CREATE VIEW XX\_SUPPLIER(RMNO,RNAME,SNAME)  AS SELECT RMNO,RMNAME,SNO,SNAME  FROM RAWMATERAIL,SUPPLY,SUPPLIER  WHERE RNAME = 'XX' AND  RAWMATERAIL.RMNO = SUPPLY.RMNO AND  SUPPLY.SNO = SUPPLIER.SNO  ORDER BY RMNO;   1. **使用某种原料的商家**   CREATE VIEW USE\_XX(MNO,MNAME,RNAME)  AS SELECT MNO,MNAME,RMNAME  FROM MERCHANT,PURCHASE,RAWMATERIAL  WHERE RNAME = 'XX' AND  RAWMATERIAL.RMNO = PURCHASE.RMNO AND  PURCHASE.MNO = MERCHANT.MNO  ORDER BY MNO;   1. **使用某种支付方式**   CREATE VIEW PAY\_TIMES  AS SELECT PNO,PAYMENT\_TYPE,SUM(ONO) AS TIMES  FROM PAYMENT,ORDER\_DETAIL  WHERE PAYMENT.PNO = ORDER\_DETAIL.PNO  GROUP BY PNO;   1. **热销菜品**   CREATE VIEW HOTDINING  AS SELECT DISHNO,DNAME,SUM（ONO) \* NUMBER AS TOTAL  FROM ORDER\_ITEM,DISH  WHERE ORDER\_ITEM.DISHNO = DISH.DISHNO  GROUP BY ONO; |
| **（8）数据库安全方案的设计** |
| USER01 – 可以对订单表与订单明细表进行操作，并可将操作权限赋予其他使用者  USER02 – 由USER01赋权，仅能查看订单表与订单明细表 |
| **2.实现部分** |
| **（1）表格创建（包括约束条件的创建）** |
| **1create**  **1**  **CRE1CRE2CRE3**  **DESC1**  **DESC2DESC3** |
| **（2）数据录入（请一定记得COMMIT提交数据）** |
| 2  3  4  5  6  7  8  9  10  12  13  14  161  162  17  181  182 |
|  |
| **（3）常见查询的例子演示** |
| ①信息管理与工程学院的参赛运动员信息  20②男子跳远的比赛结果  NO.2  ③所有计时裁判员的姓名和编号  22  ④参赛学院的个数  23  ⑤获得女子4×100米项目第一名的学院编号和学院名称  24  ⑥参加男子1500米比赛的运动员编号和参赛顺序  NO.6  ⑦所有获得个人项目的第一名的运动员信息NO.7  ⑧赛事类别名称为“田赛(FIELD\_EVENT)”的项目名称和场地名称  NO.8  ⑨男子跳远项目的规则编号  27  ⑩参加男子4\*100接力的运动员学号和姓名28  ⑪获得女子800米前三的运动员姓名和学号29  ⑫男子跳高比赛的学院限制人数NO.12  ⑬赛事类别名称为“径赛(TRACK\_EVENT）”的项目名称、规则编号和场地编号31  ⑭男子4×100米接力前三名的学院编号和学院名称NO.14  （15）获得女子3000米第二名的运动员姓名和成绩  33 |
| 1. **视图创建** |
| * 1. **余额超过200元用户**   CREATE VIEW BALANCEUP200  AS SELECT ID,ANAME,BALANCE  FROM ACCOUNT  WHERE BALANCE >= 200  ORDER BY ID;     * 1. **余额低于200元用户**   CREATE VIEW BALANCEDOWN200  AS SELECT ID,ANAME,BALANCE  FROM ACCOUNT  WHERE BALANCE < 200  ORDER BY ID;     * 1. **2020年选择堂食用户**   CREATE VIEW EATIN(ID,CNO,CNAME,CTEL,ORDER\_DATE,D\_TYPE)  AS SELECT ACCOUNT.ID,ACCOUNT.CNO,CNAME,CTEL,ORDER\_DATE,D\_TYPE  FROM CTRIPUSER,ACCOUNT,CTRIPORDER,ORDER\_DETAIL,DINING  WHERE D\_TYPE = ‘EATIN’ AND  DINING.DNO = ORDER\_DETAIL.DNO AND  ORDER\_DETAIL.ONO = CTRIPORDER.ONO AND  CTRIPORDER.ID= ACCOUNT.ID AND  ACCOUT.CNO = CTRIPUSER.CNO  ORDER BY ID;       * 1. **所有提供某种原料的供应商（以RW001-Corn为例）**   CREATE VIEW CORN\_SUPPLIER  AS SELECT RAWMATERIAL.RMNO,RNAME,SUPPLIER.SNO,SNAME  FROM RAWMATERIAL,SUPPLY,SUPPLIER  WHERE RNAME = 'Corn' AND  RAWMATERIAL.RMNO = SUPPLY.RMNO AND  SUPPLY.SNO = SUPPLIER.SNO;     * 1. **使用某种原料的商家(Y以RW002-Cabbage为例)**   CREATE VIEW USE\_CABBAGE(MNO,MNAME,RNAME)  AS SELECT MERCHANT.MNO,MNAME,RNAME  FROM MERCHANT,PURCHASE,RAWMATERIAL  WHERE RNAME = 'Cabbage' AND  RAWMATERIAL.RMNO = PURCHASE.RMNO AND  PURCHASE.MNO = MERCHANT.MNO  ORDER BY MNO;     * 1. **使用某种支付方式的频数**   CREATE VIEW PAY\_TIMES  AS SELECT PNO, COUNT(ONO) AS TIMES  FROM ORDER\_DETAIL  WHERE PNO = PNO  GROUP BY PNO;       * 1. **热销菜品**   CREATE VIEW HOTDINING  AS SELECT DISHNO,SUM（QUANTITY) AS TOTAL  FROM ORDER\_ITEM  GROUP BY DISHNO; |
| **（5）数据库安全（可选）** |
| 1.用户创建    2.scott授权                3.删除用户 |
| **3.总结** |
| **（1）组员分工** |
|  |
| **（2）难点与不足分析** |
|  |
| **（3）收获或体会** |
|  |