

# 《Database System Concept》实验指导5.0

## 实验一 实验环境的搭建

### [实验目的]

1. 掌握建立本地 Net服务名（TNS）的方法；
2. 熟悉和掌握使用 SQL Plus工具操作数据库；
3. 熟悉 TOAD工具；

### [实验内容]

1. 使用 Oracle 的 TNS配置程序建立本地服务名；
  - a.运行 Oracle的 TNS配置程序；（程序里面的 oracle目录下的 Net Configuration Assistant）
  - b.输入远程数据库服务名——helowin；
  - c.输入远程数据库 IP地址（172.22.114.116）和端口（1521）；
  - d. 使用自己学号前加s作为用户名，学号作为密码测试；
  - e.输入一个自己容易记住的名字命名TNS，如 orcl之类的。
  - f.完成配置。
2. 使用 SQL Plus 工具登陆远程数据库；
  - a. 运行 Oracle的 SQL Plus工具；
  - b. 使用自己学号前加s作为用户名，学号作为密码，主机字符串为刚才自己配置的 TNS登陆；
  - c. 记录程序信息；
  - d. 使用“alter user 你的用户名 identified by 你的新密码；”来修改初始密码；
  - e. 键入“create table taba(cola int,colb int);”，记录 SQL语句运行结果；
  - f. 键入“insert into taba values(1,1);”，记录 SQL语句运行结果；
  - g. 键入“commit;”，提交插入操作；
  - h. 退出应用程序。
3. 在命令行下使用 SQL Plus登陆远程数据库；
  - a.点击运行菜单（或者按 Win+R键）；
  - b.输入 cmd；
  - c.在命令行窗口下输入 sqlplus 用户名/密码@TNS,如 sqlplus scott/tiger@orcl;
  - d.记录程序信息；
  - e.键入“select \* from taba;”，记录 SQL语句运行结果；
  - f.退出程序。
4. 使用 TOAD工具登陆远程数据库。
  - a.运行 TOAD程序（双击桌面的青蛙图标）；
  - b.选择刚才设定的 TNS，使用自己的姓名作为帐户和学号作为密码登陆；
  - c.打开 SQL Editor，键入“select \* from taba;”，按 F9，记录 SQL运行结果；
  - d.退出程序。

### [实验报告]

1. 记录所有的操作和操作结果；
2. 比较三种不同方式使用数据库的不同。

## 实验二 实验数据库的建立

### [实验目的]

1. 熟悉和掌握 SQL的 Create Table, Drop Table, Alter Table语句语法;
2. 熟悉和掌握 SQL Plus或 TOAD等开发工具;
3. 熟悉主键, 外键, check等完整性约束;
4. 掌握完整性约束的建立, 删除, 有效化和无效化; 熟悉相关的DDL命令。
5. 掌握脚本运行的方法。

### [预备知识]

- 1.掌握登陆远程数据库方法;
- 2.掌握至少一种 Oracle PL/SQL开发工具的使用;
3. PL/SQL中 Create Table, Drop Table, Alter Table语句语法。

### [实验内容]

#### 1. 建表

下载见表DDL脚本

[https://www.db-book.com/university-lab-dir/sample\\_tables-dir/DDL.sql](https://www.db-book.com/university-lab-dir/sample_tables-dir/DDL.sql)

在sqlplus中或者Toad中使用@命令执行建库脚本

#### 2. 插入数据库数据

下载数据脚本

[https://www.db-book.com/university-lab-dir/sample\\_tables-dir/largeRelations/largeRelationsInsertFile.sql](https://www.db-book.com/university-lab-dir/sample_tables-dir/largeRelations/largeRelationsInsertFile.sql)

在sqlplus中或者Toad中使用@命令执行建库脚本

#### 3. 主键约束

- a. 插入一条数据, 使得其违反instructor的主键约束
- b. 使得instructor主键无效化, 插入上面那条数据
- c. 尝试使instructor主键有效, 记录出现的结果
- d. 删除之前插入的数据, 使得instructor主键有效。

#### 4. 外键约束

- a. instructor表里插入一条数据, 使得其违反dept\_name列上的外键约束。
- b. department中加入新的院系"Cybersecurity", 在instructor表中加入一位老师, 院系为"Cybersecurity"。
- c. 在department中删除"Cybersecurity", 查看和记录instructor的变化, 分析原因。
- d. 删除instructor表的dept\_name上的外键约束
- e. 建立instructor表的外键约束, 删除规则为No action。

- f. 删除"Cybersecurity"院系，查看和记录结果，分析原因。
- 5. unique约束
  - a. 在takes表上建立一个unique约束，保证每人每学期每课程只能修一个课程段。
  - b. 创建一个样例，当插入这条记录时，会违反上面建立的约束，记录结果。
  - c. 使得刚才建立的约束无效化，并再次运行上一条SQL语句，记录结果。
  - d. 删除掉这个约束。
- 6. check约束
  - a. 在instructor表上建立一个check约束，保证工资不超过500000。
  - b. 在instructor表中插入一条记录，使其违反上续约束。

## [实验报告]

- 1. 保留所有建表的脚本
- 2. 记录所有的脚本，运行结果，并分析原因。
- 3. 理解主键，外键的含义及区别。
- 4. unique 约束与主键的关系和区别。
- 5. check 约束使用的限制。

## 实验三 插入、更新与删除

### [实验目的]

1. 熟练掌握 insert, update, delete语句的使用;
2. 理解提交与回滚的用法
3. 熟悉case子句的使用

### [预备知识]

1. SQL中的insert, update, delete的语法
2. rollback与commit的含义

### [实验内容]

下列操作1-9都必须只使用一条SQL语句完成。

1. 更新“99052”老师工资为40000。
2. 给“Comp. Sci.”院的老师加薪10%。
3. 给“Taylor”楼里的所有院系的所有老师涨薪1000。
4. 没有授课的教师中, 如果工资超过100000, 减薪10%; 否则减薪8%。
5. 将“History”学院的办公地点转到“English”学院所在楼。
6. 插入一个新老师, 院系为空值。
7. 删除院系为空值的所有老师。
8. 删除所有没有学生修读的课程段。
9. 成绩为“C-”视为没有通过这门课程, 请统计所有学生的已修学分, 并更新学生的tot\_cred列。
10. 建立一个history\_student表, 表的结构与student表相同
11. 将已修学分超过100分的学生, 插入history\_student表。
12. 开启另外一个会话, 连接到服务器。
13. 在原会话上修改或删除表里的一条记录, 在新会话上查看这个表的结果。
14. 在原会话上使用rollback回滚。在原会话上查看这个表的结果。
15. 在原会话上修改或删除表里的一条记录, 使用commit提交, 然后在两个会话中分别查看对应表的查询结果。

### [实验报告]

1. 保留所有脚本并记录实验结果。
2. 记录所有任务的 SQL 语句, 及查询的结果。
3. 回滚和提交的作用是什么?

## 实验四 查询与视图

### [实验目的]

1. 掌握 select 的基本使用。
2. 集合运算的语法。
3. 掌握建立视图的语法。
4. 掌握和理解外部连接的使用。
5. 掌握 group by 子句和基本聚类函数的使用。

### [预备知识]

1. select 语句的基本语法
2. create view 的语法
3. 外部链接的含义及语法

### [实验内容]

1. 查询所有教师的姓名。
2. 查询所有姓名中以 A 开头的教师。
3. 查询所有姓名中无视大小写，含字母 a 的教师。
4. 查询所有姓名中第 3 个字母是字母 a 的教师。
5. 查询“Comp. Sci.”院中已修学分低于 10 分的同学。
6. 查询课程与其先修课程的课程名（不需要查询没有先修课程的课）。
7. 查询课程与其先修课程的课程名（如果先修课程，那么先修课程的课程名为空值）。
8. 查询课程和其后继课程（如果没有后继课程，那么后继课程的课程名为空值）。
9. 查询“Shishkin”同学修课的记录，包括年份，学期，课程号，课程名称，课程学分，成绩；并按年份，学期，课程名称排序。
10. 查询所有院系的教师人数，并按教师人数降序排列。
11. 查询所有院系的平均工资，最低工资，最高工资。
12. 查询每年每学期开设课程总数，课程段总数，修课人总数，修课人次总数。
13. 工资超过 50000 的教师中，按院系分类，查询他们的平均工资，并按教师数逆序，院系名正序排列。
14. 查询平均工资超过 50000 的所有院系的院系名称，院系人数。
15. 建立一个视图，表示教师授课情况，包括教师 ID，姓名，院系，授课年份，授课学期，授课课程号，课程段号，课程名称。
16. 建立一个视图，表示学校所有人员的信息，包括人员编号（老师用教师号，学生用学号），姓名，所属院系，人员类型（教师为“instructor”，学生为“student”）。
17. 建立一个表 grade\_point (grade,point)，表示成绩的绩点，并加入数据

grade	point
A+	4.0
A	3.7
A-	3.2
B+	2.8
B	2.4
B-	2.0
C+	1.5
C	1.0
C-	0.0

GPA 的计算规则为

$$GPA = \frac{\sum(\text{课程绩点} \cdot \text{课程学分})}{\sum \text{课程学分}}$$

建立一个视图，查询所有学生的 ID，姓名，院系和其 GPA。如果学生没有修课，那么 GPA 为空值。

### [实验报告]

1. SQL 都是用一条 SQL 完成。
2. 记录所有任务的 SQL 语句，及查询的结果。

## 实验五 复杂查询

### [实验目的]

1. 掌握子查询的使用
2. 掌握 with 子句的使用
3. 掌握 Top-K 查询的写法
4. 掌握交叉表查询的写法
5. 了解开窗函数的使用
6. 了解 rollup 和 cube 的使用

### [预备知识]

1. 常用聚类函数的使用语法。
2. in 子查询与 exists 子查询。
3. with 子句的语法。
4. 标量子查询的语法

### [实验内容]

1. 查询修读人数最多的课程号，课程名称。
2. 查询'Comp. Sci.', 'Biology', 'Physics'院的老师 ID 和姓名
3. 成绩为"C-"视为不及格，求通过率最低的课程号，课程名称，学分和通过率。
4. 查询工资比所在院平均工资高的老师的职工号，姓名，院系，工资。
5. 查询修读了'63039'同学所修所有课程的学生号和姓名。
6. 使用 rank 函数查询已修学分最多前 10 的同学的学号，姓名，院系，已修学分和排名
7. 使用标量子查询实现上一个的查询
8. 使用 rank 函数查询各院，已修学分最多的前 3 同学的学号，姓名，院系，已修学分和排名
9. 使用标量子查询实现上一个的查询
10. 使用 dense\_rank 函数查询已修学分最多前 10（密集排名）的同学的学号，姓名，院系，已修学分和密集排名
11. 使用标量子查询实现上一个的查询
12. 使用 dense\_rank 函数查询各院，已修学分最多的前 3（密集排名）同学的学号，姓名，院系，已修学分和密集排名
13. 使用标量子查询实现上一个的查询
14. 查询各院在 2008 年到 2010 年三年的修课总人次，列依次为院系名，2008，2009，2010，并按各院字典序排列显示。
15. 查询各课程 A 类(A+,A,A-),B 类(B+,B,B-),C 类(C+,C,C-)成绩的人数，列依次为课程号，课程名称，A，B，C。
16. 查询各院姓名首字母是 A,B,C 的老师的工资总额，列依次为院名，A，B，C。
17. 查询每年的修课人次和近三年的均值（前年，去年，今年）。
18. 查询每年各院的修课总人次，每年的修课总人次，全校修课总人次。
19. 查询每年各院的修课总人次，每年的修课总人次，各院的修课总人次，全校修课总人次。

### [实验报告]

1. SQL 都是用一条 SQL 完成。
2. 记录所有任务的 SQL 语句，及查询的结果。
3. 比较 in 子句与 exists 子句的区别。

## 实验六 关系型数据库设计

### [实验目的]

1. 掌握 E-R 的画法
2. 掌握数据库设计的方法和要点
3. 理解和掌握范式理论对于数据库设计的要求

### [预备知识]

1. ER 模型及图元
2. ER 模型到关系型数据库转换规则
3. 范式理论中的检测与分解算法

### [实验内容]

1. 有下列数据需求，请画出这个数据库的 E-R 设计图。某大学的程序设计实践系统数据需求如下：
  - a) 用户包括用户号，姓名，年级，专业，班级，密码，最近登录时间，最近使用语言，做过的题目数，做对的题目数，所属角色。
  - b) 题目包括题号，名称，时间限制，内存限制，题目内容。题目含若干组测试数据，一组测试数据包括输入数据和输出数据。题目支持出题方提供的验证器（可执行文件）进行数据验证判题。
  - c) 用户可以对题目进行提交，每个提交包括提交号，所用语言，源代码。系统会对提交进行判题，会返回判题结果和运行时长，所用内存大小。运行时长和所用内存大小在某些判题结果中为空值。
  - d) 比赛包括比赛号，比赛名称，比赛开始时间，比赛时长，比赛的描述，比赛公告，比赛的类型分为公开、私有。用户可以创建比赛，并且可以对自己创建的比赛进行管理。每个比赛包含若干道题目，一场比赛中赛题应该是唯一的。用户可以参加比赛，并在比赛中对赛题进行提交，可以查看自己提交的判题结果。系统并根据过题数和罚时，对参加比赛的用户进行排名，用户可以看到排名情况。
2. 根据你的 E-R 图，将其转换成数据库模型，并指出其符合第几范式的要求。
3. 你的数据库模型满足 BCNF 范式的要求吗？是否可以修改其设计使得其达到 BCNF 范式要求？如果可以请修改其设计，如果不可以请修改其设计达到 3NF 的要求，以及怎么做，才能保证数据的一致性。
4. 加入下列需求变更
  - a) 题目的每组数据有一个对应的分数，题目需要对所有测试数据进行测试，题目的得分为所有获得通过的数据分值的累加和。用户可以看到每组数据的测试结果和最后的分数。比赛中新增一个 OI 排名，其按照所有题目获得分值的累加逆序排名。  
修改你的 E-R 设计和数据库模型设计，使得其满足新的需求。
  - b) 增加一个角色的管理功能。管理员可以建立和删除角色，查询角色的列表和属于这个角色的用户，可以更改角色的名称和权限。一个用户只能属于某一个角色，一个角色可以有多个用户。  
修改你的 E-R 设计和数据库模型设计，使得其满足新的需求。
  - c) 将用户和角色的权限改为多对多，即一个用户可以是多种角色，一个角色有多个用户。用户拥有的权限是所有角色权限的并集。修改你的 E-R 设计和数据库模型设计，使得其满足新的需求。

### [实验报告]

1. E-R图，数据库设计文档（数据字典，数据库模型图）
2. 实验的问题的答案及解决方法。



## 实验七 范式理论

### [实验目的]

1. 掌握范式的概念
2. 掌握属性闭包的计算及应用
3. 掌握函数依赖保持的概念及算法
4. 掌握BCNF判定算法，分解后模式的BCNF判定算法
5. 掌握Fc的计算算法及3NF分解算法
6. 掌握BCNF分解算法

### [预备知识]

1. 范式理论中的基础概念
2. 范式理论中的检测与分解算法
3. 程序设计能力

### [实验内容]

1. 实现属性闭包计算算法，并应用此算法，实现判定超码，判定候选码，计算函数依赖集闭包，判定函数依赖是否属于函数依赖集闭包等功能。
2. 实现函数依赖保持判定算法。
3. 实现BCNF的判定算法，用于判断模式是否满足BCNF。
4. 实现分解后模式的BCNF判定算法，用于判断分解后的模式是否属于BCNF。
5. 实现函数依赖集正则覆盖的构造算法。
6. 实现3NF分解算法。
7. 实现BCNF分解算法。
8. 将以上功能组合形成一个完整的工具软件。

### [实验报告]

1. 范式工具软件的设计与实现文档
2. 源代码
3. 使用说明书