数据庫習題：

8．设有关系模式R（职工编号，日期，日营业额，部门名，部门经理），该模式统计商店里每个职工的日营业额，以及职工所在的部门和经理信息。如果规定：每个职工每天只有一个营业额；每个职工只在一个部门工作；每个部门只有一个经理。试回答下列问题：

(1) 根据上述规定，写出模式R的基本FD和候选键。

(2) 说明R不是2NF的理由，并把R分解成2NF模式集。

(3) 进而分解成3NF模式集。

解：

1. 基本的FD(函数依賴)有三个：

（职工编号，日期）→ 日营业额

职工编号 → 部门名

部门名 → 部门经理

R的关键码为：（职工编号，日期）。

⑵ R中有两个这样的FD：

（职工编号，日期）→（部门名，部门经理）

职工编号 →（部门名，部门经理）

可见前一个FD是局部依赖，所以R不是2NF模式。

R应分解Rl（职工编号，部门名，部门经理）

R2（职工编号，日期，日营业额）

此处，Rl和R2都是2NF模式。

⑶ R2已是3NF模式。

在R1中，存在两个FD：职工编号 → 部门名

部门名 → 部门经理

因此，“职工编号 → 部门经理”是一个传递依赖，Rl不是3NF模式。

R1应分解成R11（职工编号，部门名）

R12（部门名，部门经理）

这样，ρ={R11，Rl2，R2}是一个3NF模式集。

9．设有关系模式R（运动员编号，比赛项目，成绩，比赛类别，比赛主管），如果规定：每个运动员每参加一个比赛项目，只有一个成绩；每个比赛项目只属于一个比赛类别；每个比赛类别只有一个比赛主管。试回答下列问题：

(1) 根据上述规定，写出模式R的基本FD和候选键。

(2) 说明R不是2NF的理由，并把R分解成2NF模式集。

(3) 进而分解成3NF模式集。

解：⑴ 基本的FD有3个：

（运动员编号，比赛项目）→ 成绩

比赛项目 → 比赛类别

比赛类别 → 比赛主管

R的关键码为（运动员编号，比赛项目）。

⑵ R有两个这样的FD：

（运动员编号，比赛项目）→ （比赛类别，比赛主管）

比赛项目 → （比赛类别，比赛主管）

可见，前一个FD是部分（局部）函数依赖，所以R不是2NF模式。

如果把R分解成R1（比赛项目，比赛类别，比赛主管）

R2（运动员编号，比赛项目，成绩）

这里，R1和R2都是2NF模式。

⑶ R2已是3NF模式。

在R1中，存在两个FD：比赛项目 → 比赛类别

比赛类别 → 比赛主管

因此，“比赛项目 → 比赛主管”是一个传递依赖，R1不是3NF模式。

R1应分解为R11（比赛项目，比赛类别）

R12（比赛类别，比赛主管）

这样，ρ={R11，R12，R2}是一个3NF模式集。

11．设某商业集团数据库中有三个实体集。一是“商店”实体集，属性有商店编号、商店名、地址等；二是“商品”实体集，属性有商品号、商品名、规格、单价等；三是“职工”实体集，属性有职工编号、姓名、性别、业绩等。

商店与商品间存在“销售”联系，每个商店可销售多种商品，每种商品也可放在多个商店销售，每个商店销售每一种商品，有月销售量；商店与职工间存在着“聘用”联系，每个商店有许多职工，每个职工只能在一个商店工作，商店聘用职工有聘期和月薪。

试画出ER图，并在图上注明属性、联系的类型。再转换成关系模式集，并指出每个关系模式的主键和外键。

解：ER图如下图所示。

M

销售

聘用

商店编号

商店

商店名

地址

聘期

月薪

职工编号

业绩

职工

性别

姓名

月销售量

N

1

N

商品号

商品名

单价

商品

规格

所转换成的关系：

商品（商品号，商品名，规格，单价）

商店（商店编号，商店名，地址）

销售（商店编号，商品号，月销售量）

职工（职工编号，姓名，性别，业绩，聘期，月薪，商店编号）

12．设某商业集团数据库中有三个实体集。一是“公司”实体集，属性有公司编号、公司名、地址等；二是“仓库”实体集，属性有仓库编号、仓库名、地址等；三是“职工”实体集，属性有职工编号、姓名、性别等。

公司与仓库间存在“隶属”联系，每个公司管辖若干仓库，每个仓库只能属于一个公司管辖；

仓库与职工间存在“聘用”联系，每个仓库可聘用多个职工，每个职工只能在一个仓库工作，仓库聘用职工有聘期和工资。

试画出E-R图，并在图上注明属性、联系的类型。再转换成关系模式集，并指出每个关系模式的主键和外键。

解：ER图及属性、联系图如下：

N

1

隶属

聘用

仓库编号

仓库

仓库名

地址

聘期

工资

N

1

公司编号

公司名

地址

公司

职工编号

职工

性别

姓名

将ER图转换成为关系模式集为：

公司（公司编号，公司名，地址）

仓库（仓库编号，仓库名，地址，公司编号）

职工（职工编号，姓名，性别，仓库编号，聘期，工资）

13．设某商业集团数据库中有三个实体集。一是“商品”实体集，属性有商品号、商品名、规格、单价等；二是“商店”实体集，属性有商店号、商店名、地址等；三是“供应商”实体集，属性有供应商编号、供应商名、地址等。

供应商与商品间存在“供应”联系，每个供应商可供应多种商品，每种商品可向多个供应商订购，供应商供应每种商品有月供应量；商店与商品间存在“销售”联系，每个商店可销售多种商品，每种商品可在多个商店销售，商店销售商品有月计划数。

试画出E-R图，并在图上注明属性、联系的类型。再转换成关系模式集，并指出每个关系模式的主键和外键。

解：ER图及属性、联系图为：

M

N

M

供应

销售

月计划数

商品号

单价

商品

商品名

规格

供应商名

供应商编号

供应商

地址

商店

商店号

地址

商店名

月供应量

N

这个ER图转换的关系模式如下：

商品（商品号，商品名，规格，单价）

供应商（供应商编号，供应商名，地址）

商店（商店号，商店名，地址）

供应（商品号，供应商编号，月供应量）

销售（商品号，商店号，月计划数）

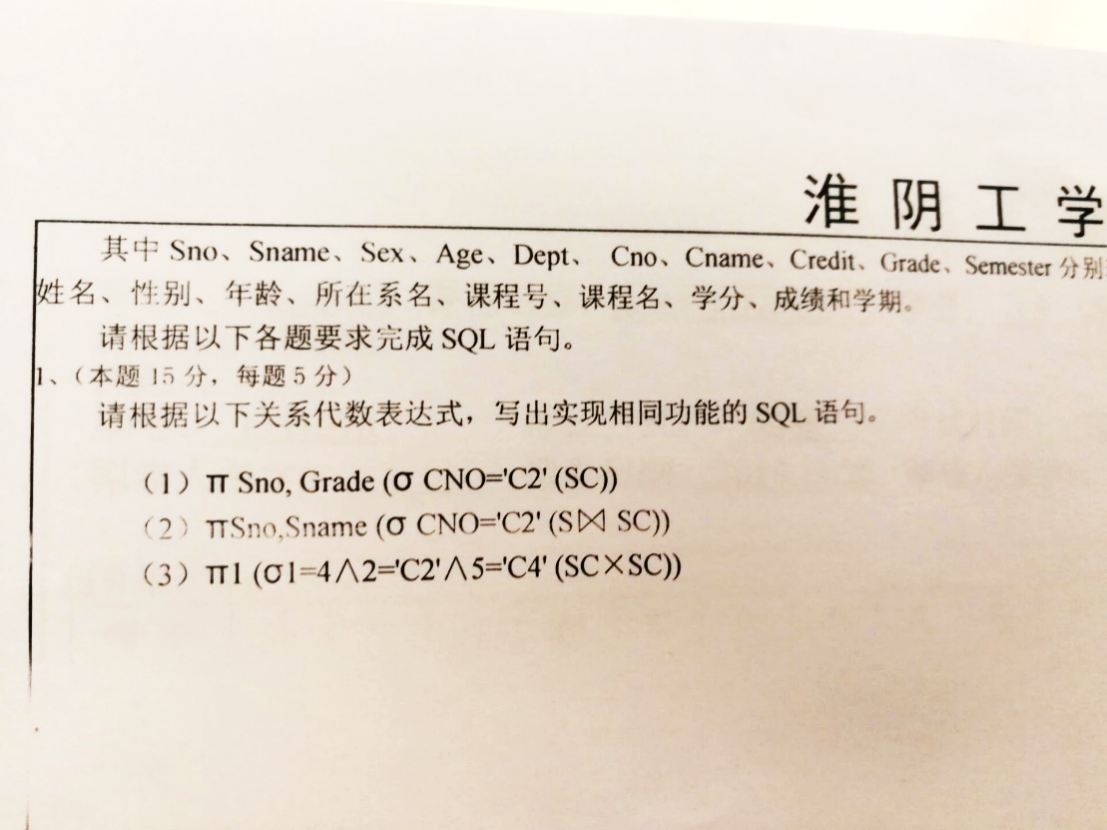
3. 对于JXGL数据库的三个基本表

S(SNO,SNAME, SEX, AGE,SDEPT)

SC(SNO,CNO,GRADE)

C(CNO,CNAME,CDEPT,TNAME)

表数据參考P40, P41



1. 检索学习课程号为C2的学生姓名和成绩。

Select sno, grade from sc where cno=’c2’;

1. 检索学习课程号为C2的学生学号和成绩(关连S和SC表)

Select sc.sno, s.sname from sc inner join s on sc.sno=s.sno where sc.cno='c2';

1. 检索至少选修课程号为C2和C4的学生学号。

Select sno from sc where cno in ('c2', 'c4');

2．对于JXGL数据库的三个基本表

S(SNO,SNAME, SEX, AGE,SDEPT)

SC(SNO,CNO,GRADE)

C(CNO,CNAME,CDEPT,TNAME)

试用T-SQL查询语句表达下列查询：

(1) 查询“王志强”所授课程的课程号和课程名。

(2) 查询年龄大于20岁的男学生的学号和姓名。

(3) 查询学号为S6的学生所学课程的课程名和任课教师名。

(4) 查询至少选修“王志强”老师所授课程中一门课程的女学生姓名。

(5) 查询“李小刚”同学不学的课程的课程号。

(6) 查询至少选修两门课程的学生学号。

解：

(1) USE JXGL

GO

SELECT CNO,CNAME

FROM C

WHERE TNAME='王志强'

GO

(2) USE JXGL

GO

SELECT SNO,SNAME

FROM S

WHERE SEX='M' AND AGE>20

GO

(3) USE JXGL

GO

SELECT CNAME,TNAME

FROM S JOIN SC ON S.SNO=SC.SNO AND S.SNO='S6'

JOIN C ON SC.CNO=C.CNO

GO

(4) USE JXGL

GO

SELECT SNAME

FROM S

WHERE SNO IN

(

SELECT SNO

FROM SC

WHERE CNO IN

(SELECT CNO

FROM C

WHERE TNAME='王志强'

)

)

GO

(5) USE JXGL

GO

SELECT CNO

FROM C

EXCEPT

SELECT CNO

FROM S JOIN SC ON S.SNO=SC.SNO AND S.SNAME='李小刚'

GO

(6) USE JXGL

GO

select SNO,count(CNO) as 选修门数

from sc

group by SNO having count(CNO)>1

GO

3. 试用T-SQL查询语句表达下列对习题2数据库中三个基本表S、SC、C的查询：

(1) 统计有学生选修的课程门数。

(2) 求选修C4号课程的学生的平均年龄。

(3) 求“王志强”老师所授课程的每门课程的学生平均成绩。

(4) 统计每门课程的学生选修人数（超过10人的课程才统计）。要求输出课程号和选修人数，查询结果按人数降序排列，若人数相同，按课程号升序排列。

(5) 查询姓“王”的所有学生的姓名和年龄。

(6) 在SC中查询成绩为空值的学生学号和课程号。

(7) 查询年龄大于女同学平均年龄的男学生姓名和年龄。

解：

(1) USE JXGL

GO

SELECT COUNT(DISTINCT CNO)

FROM SC

GO

(2) USE JXGL

GO

SELECT AVG(AGE)

FROM S JOIN SC ON S.SNO=SC.SNO AND CNO='C4'

GO

(3) USE JXGL

GO

SELECT SC.CNO,AVG(GRADE)

FROM SC JOIN C ON SC.CNO=C.CNO AND TNAME='王志强'

GROUP BY SC.CNO

GO

(4) USE JXGL

GO

SELECT CNO,COUNT(SNO)

FROM SC

GROUP BY CNO HAVING COUNT(\*)>10

ORDER BY 2 DESC,1

GO

(5) USE JXGL

GO

SELECT SNAME,AGE

FROM S

WHERE SNAME LIKE '王%'

GO

(6) USE JXGL

GO

SELECT SNO,CNO

FROM SC

WHERE GRADE IS NULL

GO

(7) USE JXGL

GO

SELECT SNAME,AGE

FROM S

WHERE SEX='M' AND AGE>(SELECT AVG(AGE)

FROM S

WHERE SEX='F')

GO

JXGL数据表

Create database JXGL;

Use JXGL;

use JXGL;

create table sc (

sno VARCHAR(9) NOT NULL,

cno VARCHAR(4) NOT NULL,

grade int NULL check (grade is NULL or( grade between 0 and 100)),

PRIMARY KEY(sno, cno))ENGINE=InnoDB charset=utf8;

create table c (

cno VARCHAR(4) NOT NULL,

cname varchar(20),

cdept varchar(10),

tname varchar(20),

PRIMARY KEY(cno))ENGINE=InnoDB charset=utf8;

create table s (

sno VARCHAR(9) NOT NULL,

sname varchar(8),

sex varchar(2),

age int(4),

sdept varchar(10),

PRIMARY KEY(sno))ENGINE=InnoDB charset=utf8;

INSERT INTO `jxgl`.`c` (`cno`, `cname`, `cdept`, `tname`) VALUES ('c1', '高等数學', 'IS', '王紅衛');

INSERT INTO `jxgl`.`c` (`cno`, `cname`, `cdept`, `tname`) VALUES ('c2', '数据庫原理', 'CS', '李紹麗');

INSERT INTO `jxgl`.`c` (`cno`, `cname`, `cdept`, `tname`) VALUES ('c3', '数据結構', 'CS', '刘良');

INSERT INTO `jxgl`.`s` (`sno`, `sname`, `sex`, `age`, `sdept`) VALUES ('s1', '王麗萍', 'F', '19', 'CS');

INSERT INTO `jxgl`.`s` (`sno`, `sname`, `sex`, `age`, `sdept`) VALUES ('s2', '姜芸', 'F', '20', 'IS');

INSERT INTO `jxgl`.`s` (`sno`, `sname`, `sex`, `age`, `sdept`) VALUES ('s3', '馬長友', 'M', '20', 'CS');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s1', 'c1', '87');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s1', 'c2', '78');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s1', 'c3', '90');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s2', 'c1', '67');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s2', 'c2', '79');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s2', 'c3', '56');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s3', 'c1', '80');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s3', 'c2', '76');

INSERT INTO `jxgl`.`sc` (`sno`, `cno`, `grade`) VALUES ('s3', 'c3', '92');

创建student表

在性别上，设置了只能选 男或者是女

mysql> create table student (

-> sno VARCHAR(7) NOT NULL,

-> sname VARCHAR(16),

-> ssex CHAR(2) DEFAULT '男' CHECK (ssex='男' or ssex='女'),

-> sage int,

-> sdept VARCHAR(2),

-> PRIMARY KEY(sno)) ENGINE = InnoDb;

Query OK, 0 rows affected (0.20 sec)

创建course表

要注意：课程表上，cpno其实是course表本身的cno的外键。

mysql> create table course (

-> cno VARCHAR(2) NOT NULL,

-> cname VARCHAR(20),

-> cpno VARCHAR(2) NULL,

-> credit int,

-> PRIMARY KEY(cno),

-> FOREIGN key(cpno) references course(cno)) ENGINE = InnoDB;

Query OK, 0 rows affected (0.17 sec)

mysql>

创建sc表

mysql> create table sc (

-> sno VARCHAR(7) NOT NULL,

-> cno VARCHAR(7) NOT NULL,

-> grade int NULL check (grade is NULL or( grade between 0 and 100)),

-> PRIMARY KEY(sno, cno),

-> Foreign Key(sno) references student(sno),

-> Foreign Key(cno) references course(cno))ENGINE=InnoDB;

Query OK, 0 rows affected (0.18 sec)

插入是student表

mysql> insert into student value('2005001', '钱横', '男', 18, 'Cs');

Query OK, 1 row affected (0.04 sec)

mysql> insert into student value('2005002', '王林', '女', 19, 'Cs');

Query OK, 1 row affected (0.06 sec)

mysql> insert into student value('2005003', '李民', '男', 20, 'Is');

Query OK, 1 row affected (0.06 sec)

mysql> insert into student value('2005004', '赵欣然', '女', 16, 'Ma');

Query OK, 1 row affected (0.17 sec)

插入课程表course

mysql> insert into course value('6', '数据处理基础', NULL, 4);

Query OK, 1 row affected (0.09 sec)

mysql> insert into course value('7', 'C语言', 6, 3);

Query OK, 1 row affected (0.08 sec)

mysql> insert into course value('4', '操作系统原理', 6, 3);

Query OK, 1 row affected (0.06 sec)

mysql> insert into course value('5', '数据结构', 7, 4);

Query OK, 1 row affected (0.04 sec)

mysql> insert into course value(1, '数据库基础', 5, 4);

Query OK, 1 row affected (0.09 sec)

mysql> insert into course value(3, '信息系统导论', 1, 3);

Query OK, 1 row affected (0.10 sec)

mysql> insert into course value(2, '数学分析', NULL, 2);

Query OK, 1 row affected (0.10 sec)

mysql>

由于之前设置了cpno到cno的一个外键，所以，需要按特定的顺序完成。

mysql> insert into sc value('2005001', 1, 87);

Query OK, 1 row affected (0.02 sec)

mysql> insert into sc value('2005001', 2, 67);

Query OK, 1 row affected (0.02 sec)

mysql> insert into sc value('2005001', 3, 90);

Query OK, 1 row affected (0.07 sec)

mysql> insert into sc value('2005002', 2, 95);

Query OK, 1 row affected (0.04 sec)

mysql> insert into sc value('2005003', 3, 88);

Query OK, 1 row affected (0.06 sec)

在第五章里面插入的一些数据

mysql> insert into student values ('2005007', '李涛', '男', 19, 'Is');

Query OK, 1 row affected (0.18 sec)

mysql> insert into student values ('2005008', '陈高', '女', 21, 'At'),

-> ('2005009', '张杰', '男', 17, 'At');

Query OK, 2 rows affected (0.03 sec)

Records: 2 Duplicates: 0 Warnings: 0

再插入一些数据，下面是添加一些人

mysql> insert into student values ('2005010', '赵青江', '男', 18, 'Cs');

Query OK, 1 row affected (0.04 sec)

mysql> insert into student values ('2005011', '张丽萍', '女', 19, 'Ch');

Query OK, 1 row affected (0.01 sec)

mysql> insert into student values ('2005012', '陈景欢', '男', 20, 'Is');

Query OK, 1 row affected (0.04 sec)

mysql> insert into student values ('2005013', '陈婷婷', '女', 16, 'Ph');

Query OK, 1 row affected (0.09 sec)

mysql> insert into student values ('2005014', '李军', '女', 16, 'Ph');

Query OK, 1 row affected (0.05 sec)

再添加一些选课的记录

mysql> insert into sc values ('2005010', 1, 87);

Query OK, 1 row affected (0.06 sec)

mysql> insert into sc values ('2005010', 2, NULL);

Query OK, 1 row affected (0.07 sec)

mysql> insert into sc values ('2005010', 3, 80);

Query OK, 1 row affected (0.04 sec)

mysql> insert into sc values ('2005010', 4, 87);

Query OK, 1 row affected (0.01 sec)

mysql> insert into sc values ('2005010', 6, 85);

Query OK, 1 row affected (0.02 sec)

mysql> insert into sc values ('2005011', 1, 52);

Query OK, 1 row affected (0.10 sec)

mysql> insert into sc values ('2005011', 2, 47);

Query OK, 1 row affected (0.01 sec)

mysql> insert into sc values ('2005011', 3, 53);

Query OK, 1 row affected (0.04 sec)

mysql> insert into sc values ('2005011', 5, 45);

Query OK, 1 row affected (0.06 sec)

mysql> insert into sc values ('2005012', 1, 84);

Query OK, 1 row affected (0.07 sec)

mysql> insert into sc values ('2005012', 3, NULL);

Query OK, 1 row affected (0.05 sec)

mysql> insert into sc values ('2005012', 4, 67);

Query OK, 1 row affected (0.02 sec)

mysql> insert into sc values ('2005012', 5, 81);

Query OK, 1 row affected (0.02 sec)