



南京航空航天大学

NANJING UNIVERSITY OF AERONAUTICS AND ASTRONAUTICS

计算机科学与技术学院  
/人工智能学院

## 数据库原理

——课程作业

课程：数据库原理

班级：1621402

姓名：黄钰轩

学号：162140222

教师：郑吉平

学期：2023-2024 春季

## 目录

1 第一章作业	1
2 第二章作业	4
3 第三章作业	10
4 第四章作业	17
5 第五章作业	19

## 1 第一章作业

**题目 1.** 试述数据、数据库、数据库管理系统、数据库系统的概念.

**解答.**

(1) 数据: 描述事物的符号记录.

- 在现实生活, 数据是可识别的抽象符号, 即描述事物的符号记录.
- 在计算机中, 数据是所有能被计算机处理的符号的总称.
- 在数据库中, 数据是数据库中存储的基本对象.

(2) 数据库:

- 一组相互有关联的数据集合.
- 长期储存在计算机中的有组织的、可管理和可共享的数据集合.

(3) 数据库管理系统:

- 位于用户 (应用程序) 与操作系统之间的数据库管理软件.
- 一个管理数据的大型复杂基础软件系统.

(4) 数据库系统:

- 由数据库管理系统和相关工具组成的软件系统, 用于管理和操作大量数据.

**题目 2.** 试述数据模型的概念、作用及其包含的三个要素.

**解答.**

(1) 数据库模型的概念:

- 数据模型是数据库结构的基础, 它是描述数据 (数据结构)、数据之间的联系, 数据语义即数据操作, 以及一致性 (完整性) 约束的概念和工具的集合.

(2) 数据库模型的作用:

- 在数据库领域, 数据模型被广泛用于表示这些数据库设计的“描述”, 更好的刻画数据.

- 数据模型精确描述了系统的静态特性、动态特性和完整性约束条件.

(3) 数据库模型的三个要素:

- 数据结构: 数据结构描述数据库的组成对象以及对象之间的联系.
- 数据操纵: 是指对数据库中各种对象 (型) 的实例 (值) 允许进行的操作的集合, 包括操作及有关的操作规则, 是对系统动态特性的描述.
- 完整性约束: 是一组完整性规则的集合. 完整性规则是给定的数据模型中数据及其联系所具有的制约和依存规则, 用以限定符合数据模型的数据库状态以及状态的变化, 以保证数据的正确、有效、相容.

**题目 3.** 试述数据库系统的三级模式结构, 并说明这种结构的优点是什么.

**解答.**

(1) 数据库系统的三级模式结构由模式、外模式和内模式组成.

- 模式也称逻辑模式, 是数据库中全体数据的逻辑结构和特征的描述, 是所有用户的公共数据视图. 模式的一个具体值称为模式的一个实例.
- 外模式, 亦称子模式或用户模式, 是数据库用户 (包括应用程序员和最终用户) 能够看见和使用的局部数据的逻辑结构和特征的描述, 是数据库用户的数据视图, 是与某一应用有关的数据的逻辑表示.
- 内模式也称物理模式或存储模式, 它是对数据物理结构和存方式的描述, 是数据在数据库内部的组织方式.

(2) 三级模式结构的优点:

- 数据库系统的三级模式是对数据的三个抽象级别, 它把数据的具体组织留给数据库管理系统, 使用户能逻辑抽象地处理数据, 而不必关心数据在计算机中的具体表示方式和存储方式.
- 数据库系统在这三级模式之间提供了两层映像: 外模式/模式映像和模式/内模式映像. 正是这两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性.

**题目 4.** 试述数据与程序的物理独立性和逻辑独立性. 为什么数据库系统具有较强的数据与程序的独立性?

解答.

- (1)
  - 数据与程序的物理独立性: 当数据库的存储结构改变时 (如选用了另一种存储结构), 由数据库管理员对模式/内模式映像做相应改变, 可以使模式保持不变, 从而应用程序也不必改变, 保证了数据与程序的物理独立性, 简称数据的物理独立性.
  - 数据与程序的逻辑独立性: 当模式改变时 (例如增加新的关系、新的属性、改变属性的数据类型等), 由数据库管理员对各个外模式/模式的映像做相应改变, 可以使外模式保持不变. 应用程序是依据数据的外模式编写的, 从而应用程序不必修改, 保证了数据与程序的逻辑独立性, 简称数据的逻辑独立性.
- (2) 数据库管理系统在三级模式之间提供的两层映像保证了数据库系统中的数据能够具有较高的逻辑独立性和物理独立性.

## 2 第二章作业

题目 1. 试述关系模型的三个组成部分.

解答. 关系模型由关系数据结构、关系操作集合和关系完整性约束三部分组成.

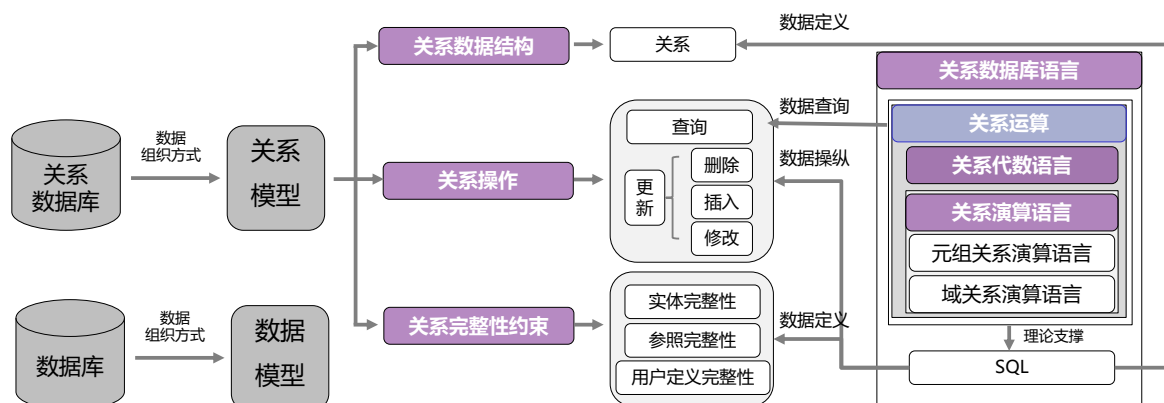


图 2.1: 关系模型框架

题目 2. 设有一个 **SPJ** 数据库，包括四个关系模式 **S**、**P**、**J** 和 **SPJ**.

$S(SNO, SNAME, STATUS, CITY);$

$P(PNO, PNAME, COLOR, WEIGHT);$

$J(JNO, JNAME, CITY);$

$SPJ(SNO, PNO, JNO, QTY).$

供应商表 **S** 由供应商代码 (SNO)、供应商姓名 (SNAME)、供应商状态 (STATUS)、供应商所在城市 (CITY) 组成.

零件表 **P** 由零件代码 (PNO)、零件名 (PNAME)、颜色 (COLOR)、重量 (WEIGHT) 组成.

工程项目表 **J** 由工程项目代码 (JNO)、工程项目名 (JNAME)、工程项目所在城市 (CITY) 组成.

供应情况表 **SPJ** 由供应商代码 (SNO)、零件代码 (PNO)、工程项目代码 (JNO)、供应数量 (QTY) 组成，表示某供应商供应某种零件给某工程项目的数量为 QTY.

今有若干数据如图 2.2 所示.

试用关系代数、元组关系演算语言 ALPHA 完成如下查询:

(1) 求供应工程 J1 零件的供应商代码 SNO.

- (2) 求供应工程 J1 零件 P1 的供应商代码 SNO.
- (3) 求供应工程 J1 零件为红色的供应商代码 SNO.
- (4) 求没有使用天津供应商生产的红色零件的工程号 JNO.
- (5) 求至少使用了与供应商 S1 所供应的全部零件相同零件号的工程号 JNO.

S

SNO	SNAME	STATUS	CITY
S1	精益	20	天津
S2	盛锡	10	北京
S3	东方红	30	北京
S4	丰泰盛	20	天津
S5	为名	30	上海

P

PNO	PNAME	COLOR	WEIGHT
P1	螺母	红	12
P2	螺栓	绿	17
P3	螺丝刀	蓝	14
P4	螺丝刀	红	14
P5	凸轮	蓝	40
P6	齿轮	红	30

J

JNO	JNAME	CITY
J1	三建	北京
J2	一汽	长春
J3	弹簧厂	天津
J4	造船厂	天津
J5	机车厂	唐山
J6	无线电厂	常州
J7	半导体厂	南京

SPJ

SNO	PNO	JNO	QTY
S1	P1	J1	200
S1	P1	J3	100
S1	P1	J4	700
S1	P2	J2	100
S2	P3	J1	400
S2	P3	J2	200
S2	P3	J4	500
S2	P3	J5	400
S2	P5	J1	400
S2	P5	J2	100
S3	P1	J1	200
S3	P3	J1	200
S4	P5	J1	100
S4	P6	J3	300
S4	P6	J4	200
S5	P2	J4	100
S5	P3	J1	200
S5	P6	J2	200
S5	P6	J4	500

图 2.2: SPJ 数据库数据

解答.

- (1) • 关系代数:

$$\Pi_{SNO}(\sigma_{JNO='J1'}(\mathbf{SPJ}))$$

- ALPHA 语言:

```
1 GET W(SPJ.SNO) :
2   SPJ.JNO = 'J1'
```

- (2) • 关系代数:

$$\Pi_{SNO}(\sigma_{JNO='J1' \wedge PNO='P1'}(\mathbf{SPJ}))$$

- ALPHA 语言:

```
1 GET W(SPJ.SNO) :
2   SPJ.JNO = 'J1'
3   ^ SPJ.PNO = 'P1'
```

- (3) • 关系代数:

$$\Pi_{SNO}(\Pi_{SNO,PNO}(\sigma_{JNO='J1'}(\mathbf{SPJ})) \bowtie \Pi_{PNO}(\sigma_{COLOR='红'}(\mathbf{P})))$$

- ALPHA 语言:

```
1 RANGE P PX
2 GET W(SPJ.SNO) :
3   SPJ.JNO = 'J1'
4   ^ ∃PX(PX.COLOR = '红'
5     ^ PX.PNO = SPJ.PNO)
```

- (4) • 关系代数:

$$\Pi_{JNO}(\mathbf{SPJ}) - \Pi_{JNO}(\sigma_{CITY='天津' \wedge COLOR='红'}(\mathbf{S} \bowtie \mathbf{SPJ} \bowtie \mathbf{P}))$$

- ALPHA 语言:

```
1 RANGE S SX
2   P PX
3   SPJ SPJX
4 GET W(J.JNO) :
```



```

5      ¬∃SPJX(SPJX.JNO = J.JNO
6          ∧ ∃SX(SX.SNO = SPJX.SNO
7              ∧ SX.CITY = '天津'
8              ∧ ∃PX(PX.PNO=SPJX.PNO
9                  ∧ PX.COLOR = '红'))

```

(5) • 关系代数:

$$\Pi_{JNO,PNO}(\mathbf{SPJ}) \div \Pi_{PNO}(\sigma_{SNO='S1'}(\mathbf{SPJ}))$$

• ALPHA 语言:

```

1 RANGE P PX
2     SPJ SPJX
3     SPJ SPJY
4 GET W(J.JNO) :
5     ∀PX(∃SPJX(SPJX.PNO = PX.PNO
6             ∧ SPJX.SNO = 'S1')
7         ⇒ ∃SPJY(SPJY.JNO = J.JNO
8             ∧ SPJY.PNO = PX.PNO))

```

**题目 3.** 试述等值连接与自然连接的区别和联系.

**解答.**

- 自然连接是一种特殊的等值连接.
- 自然连接将连接条件指定为  $R$  和  $S$  中属性名相同的列做等值连接, 因此  $p$  可省略.

题目 3 的注记. 以 **Student**、**SC** 表为例,

$$\mathbf{Student} \bowtie_{\text{Student.Sno}=\text{SC.Sno}} \mathbf{SC}$$

结果如下:

Student.Sno	Student.Sname	Student.Sgender	Student.Sage	Student.Sdept	SC.Sno	SC.Cno	SC.Grade
2021310721	李博	男	17	CS	2021310721	5	98
2021310722	赵宇	男	19	CS	2021310722	1	87
2021310723	张敏	女	18	CS	2021310723	1	92
2021310723	张敏	女	18	CS	2021310723	5	76
2021310724	王勇	男	18	MA	2021310724	7	84
2021310725	刘佳	女	17	MA	2021310725	4	95

图 2.3: 等值连接

而

$$\mathbf{Student} \bowtie \mathbf{SC}$$

结果如下:

Sno	Sname	Sgender	Sage	Sdept	Cno	Grade
2021310721	李博	男	17	CS	5	98
2021310722	赵宇	男	19	CS	1	87
2021310723	张敏	女	18	CS	1	92
2021310723	张敏	女	18	CS	5	76
2021310724	王勇	男	18	MA	7	84
2021310725	刘佳	女	17	MA	4	95

图 2.4: 自然连接

题目 4. 关系代数的基本运算有哪些? 如何用这些基本运算来表示其他运算?

解答.

- 基本运算:

– 选择 ( $\sigma$ ):

$$\sigma_p(R) = \{t | t \in R \wedge p(t) = \text{True}\}$$

– 投影 ( $\Pi$ ):

$$\Pi_{A_1, A_2, \dots, A_k}(R) = \{t[A_1, A_2, \dots, A_k] | t \in R\}$$

– 并 ( $\cup$ ):

$$R \cup S = \{t | t \in R \vee t \in S\}$$

– 差 ( $-$ ):

$$R - S = \{t | t \in R \wedge t \notin S\}$$

– 笛卡尔积 ( $\times$ ):

$$R \times S = \{(t, q) | t \in R \wedge q \in S\}$$

• 附加运算:

– 交 ( $\cap$ ):

$$R \cap S = \{t | t \in R \wedge t \in S\}$$

满足:

$$R \cap S = R - (R - S)$$

– 连接 ( $\bowtie$ ):

$$R \bowtie_p S = \{(t, q) | t \in R \wedge q \in S \wedge p((t, q)) = \text{True}\}$$

满足:

$$R \bowtie_p S = \sigma_p(R \times S)$$

– 除 ( $\div$ ):

设  $R(A_1, A_2, \dots, A_m, B_1, B_2, \dots, B_n)$  和  $S(B_1, B_2, \dots, B_n, C_1, C_2, \dots, C_k)$  是两个关系, 则  $R \div S$  的属性为  $A_1, A_2, \dots, A_m$ , 且:

$$R \div S = \{t | t \in \Pi_{A_1, A_2, \dots, A_m}(R) \wedge (\forall q \in \Pi_{B_1, B_2, \dots, B_n}(S), (t, q) \in R)\}$$

满足:

$$R \div S = \Pi_{A_1, A_2, \dots, A_m}(R) - \Pi_{A_1, A_2, \dots, A_m}((\Pi_{A_1, A_2, \dots, A_m}(R) \times \Pi_{B_1, B_2, \dots, B_n}(S)) - R)$$

### 3 第三章作业

题目 1. 有两个关系  $\mathbf{S}(A, B, C, D)$  和  $\mathbf{T}(C, D, E, F)$ , 写出与下列查询等价的 SQL 表达式:

(1)  $\sigma_{A=10}(\mathbf{S})$ .

(2)  $\Pi_{A,B}(\mathbf{S})$ .

(3)  $\mathbf{S} \bowtie \mathbf{T}$ .

(4)  $\mathbf{S} \bowtie_{S.C=T.C} \mathbf{T}$ .

(5)  $\mathbf{S} \bowtie_{A<E} \mathbf{T}$ .

(6)  $\Pi_{C,D}(\mathbf{S}) \times \mathbf{T}$ .

解答.

(1)

```
1 SELECT *
2 FROM S
3 WHERE A = '10';
```

(2)

```
1 SELECT DISTINCT A, B
2 FROM S;
```

(3)

```
1 SELECT A, B, S.C, S.D, E, F
2 FROM S, T
3 WHERE S.C = T.C
4      AND S.D = T.D;
```

(4)

```
1 SELECT *
2 FROM S, T
3 WHERE S.C = T.C;
```

(5)

```
1 SELECT *
2 FROM S, T
3 WHERE S.A < T.E;
```

(6)

```
1 SELECT S.C, S.D, T.*
2 FROM S, T;
```

**题目 2.** 使用 SQL 语句建立 (题目 2-6) 中 SPJ 数据库的四个表 (**S**、**P**、**J** 和 **SPJ**), 并针对建立的四个表用 SQL 语句完成 (题目 2-6) 中的查询.

解答.

- 建表:

— 建 **S** 表:

```
1 CREATE TABLE S (
2     SNO CHAR(2),
3     SNAME VARCHAR(8),
4     STATUS VARCHAR(2),
5     CITY VARCHAR(6)
6 );
```

— 建 **P** 表:

```
1 CREATE TABLE P (
2     PNO CHAR(2),
3     PNAME VARCHAR(8),
4     COLOR CHAR(1),
5     WEIGHT SMALLINT
6 );
```

— 建 **J** 表:

```
1 CREATE TABLE J (
2     JNO CHAR(2),
3     JNAME VARCHAR(8),
4     CITY VARCHAR(6)
5 );
```

— 建 SPJ 表:

```
1 CREATE TABLE SPJ (  
2     SNO CHAR(2),  
3     PNO CHAR(2),  
4     JNO CHAR(2),  
5     QTY SMALLINT  
6 );
```

• 查询:

(1)

```
1 SELECT DISTINCT SNO  
2 FROM SPJ  
3 WHERE JNO = 'J1';
```

(2)

```
1 SELECT DISTINCT SNO  
2 FROM SPJ  
3 WHERE JNO = 'J1'  
4     AND PNO = 'P1';
```

(3)

```
1 SELECT DISTINCT SNO  
2 FROM SPJ, P  
3 WHERE JNO = 'J1'  
4     AND SPJ.PNO = P.PNO  
5     AND COLOR = '红';
```

(4)

```
1 SELECT J.JNO  
2 FROM J  
3 WHERE J.JNO NOT IN (  
4     SELECT SPJ.JNO  
5     FROM SPJ  
6     JOIN S  
7         ON SPJ.SNO = S.SNO  
8     JOIN P  
9         ON SPJ.PNO = P.PNO  
10    WHERE S.CITY = '天津'  
11        AND P.COLOR = '红'  
12 )
```

(5)

```
1 SELECT J.JNO
2 FROM J
3 WHERE NOT EXISTS (
4     SELECT PNO
5     FROM SPJ SPJX
6     WHERE SPJX.SNO = 'S1'
7         AND NOT EXISTS (
8             SELECT 1
9             FROM SPJ
10            WHERE SPJ.JNO = J.JNO
11                AND SPJ.PNO = SPJX.PNO
12        )
13 );
```

**题目 3.** 针对 (题目 2-6) 中的四个表 (**S**、**P**、**J** 和 **SPJ**)，试用 SQL 完成以下各项操作：

- (1) 找出所有供应商的姓名和所在城市.
- (2) 找出所有零件的名称、颜色、重量.
- (3) 找出使用供应商 S1 所供应零件的工程代码.
- (4) 找出工程项目 J2 使用的各种零件的名称及其数量.
- (5) 找出上海厂商供应的所有零件代码.
- (6) 找出使用上海产的零件的工程名称.
- (7) 找出没有使用天津产的零件的工程代码.
- (8) 把全部红色零件的颜色改成蓝色.
- (9) 把由 S5 供给 J2 的零件 P6 改为由 S3 供应.
- (10) 从供应商关系中删除 S2 的记录，并从供应情况关系中删除相应的记录.
- (11) 请将 (S2, J6, P4, 200) 插入供应情况关系.

解答.

(1)

```
1 SELECT SNAME, CITY
2 FROM S;
```

(2)

```
1 SELECT PNAME, COLOR, WEIGHT
2 FROM P;
```

(3)

```
1 SELECT DISTINCT JNO
2 FROM SPJ
3 WHERE SNO = 'S1';
```

(4)

```
1 SELECT P.PNAME, SPJ.QTY
2 FROM SPJ, P
3 WHERE P.PNO = SPJ.PNO
4        AND SPJ.JNO = 'J2';
```

(5)

```
1 SELECT SPJ.PNO
2 FROM SPJ, S
3 WHERE S.SNO = SPJ.SNO
4        AND S.CITY = '上海';
```

(6)

```
1 SELECT J.JNAME
2 FROM SPJ, S, J
3 WHERE S.SNO = SPJ.SNO
4        AND S.CITY = '上海'
5        AND J.JNO = SPJ.JNO;
```

(7)

```
1 SELECT DISTINCT JNO
2 FROM SPJ
3 WHERE JNO NOT IN (
4     SELECT DISTINCT SPJ.JNO
5     FROM SPJ, S
```



```
6      WHERE S.SNO = SPJ.SNO
7      AND S.CITY = '天津'
8 );
```

(8)

```
1 UPDATE P
2 SET COLOR = '蓝'
3 WHERE COLOR = '红';
```

(9)

```
1 UPDATE SPJ
2 SET SNO = 'S3'
3 WHERE SNO = 'S5'
4     AND JNO = 'J4'
5     AND PNO = 'P6';
```

(10)

```
1 DELETE FROM S
2 WHERE SNO = 'S2';
3
4 DELETE FROM SPJ
5 WHERE SNO = 'S2';
```

(11)

```
1 INSERT INTO SPJ
2 VALUES ('S2', 'J6', 'P4', 200);
```

**题目 4.** 针对 (题目 2-6) 中的四个表 (**S**、**P**、**J** 和 **SPJ**)，为三建工程项目建立一个供应情况的视图，包括供应商代码 (SNO)、零件代码 (PNO)、供应数量 (QTY)。针对该视图完成下列查询：

- (1) 找出三建工程项目使用的各种零件代码及其数量。
- (2) 找出供应商 S1 供应三建工程情况。

解答.

- 创建视图:

```
1 CREATE VIEW SPJV AS
2 SELECT SNO, SPJ.PNO, QTY
3 FROM SPJ, J
4 WHERE SPJ.JNO = J.JNO
5 AND J.JNAME = '三建';
```

- 查询

(1)

```
1 SELECT DISTINCT PNO, QTY
2 FROM SPJV;
```

(2)

```
1 SELECT DISTINCT *
2 FROM SPJV
3 WHERE SNO = 'S1';
```

## 4 第四章作业

题目 1. 今有以下两个关系模式:

职工 (职工号, 姓名, 年龄, 职务, 工资, 部门号);

部门 (部门号, 名称, 经理名, 地址, 电话号).

请用 SQL 的 GRANT 语句和 REVOKE 语句 (加上视图机制) 实现以下授权定义和存取控制的功能:

- (1) 用户王明对两个表有 SELECT 权限.
- (2) 用户李勇对两个表有 INSERT 和 DELETE 权限.
- (3) 每个职工只对自己的记录有 SELECT 权限.
- (4) 用户刘星对职工表有 SELECT 权限, 对“工资”字段具有更新权限.
- (5) 用户张新具有修改这两个表的结构权限.
- (6) 用户周平具有对两个表的所有权限, 并具有给其他用户授权的权限.
- (7) 用户杨兰具有从每个部门职工中 SELECT 最高工资、最低工资、平均工资的权限, 但不能查看每个人的工资.

解答.

(1)

```
1 GRANT SELECT
2 ON TABLE '职工', '部门'
3 TO '王明';
```

(2)

```
1 GRANT INSERT, DELETE
2 ON TABLE '职工', '部门'
3 TO '李勇';
```

(3)

```
1 CREATE VIEW '职工个人信息' AS
2 SELECT * FROM '职工'
```

```
3 WHERE '姓名' = 'USER';  
4  
5 GRANT SELECT  
6 ON TABLE '职工个人信息'  
7 TO PUBLIC;
```

(4)

```
1 GRANT SELECT, UPDATE ('工资')  
2 ON TABLE '职工'  
3 TO '刘星';
```

(5)

```
1 GRANT ALTER TABLE  
2 ON TABLE '职工', '部门'  
3 TO '张新';
```

(6)

```
1 GRANT ALL PRIVILIGES  
2 ON TABLE '职工', '部门'  
3 TO '周平'  
4 WITH GRANT OPTION;
```

(7)

```
1 CREATE VIEW '部门工资统计' AS  
2 SELECT '部门号',  
3        MAX('工资') AS '最高工资',  
4        MIN('工资') AS '最低工资',  
5        AVG('工资') AS '平均工资'  
6 FROM '职工'  
7 GROUP BY '部门号';  
8  
9 GRANT SELECT  
10 ON TABLE '部门工资统计'  
11 TO '杨兰';
```

## 5 第五章作业

题目 1. 假设有下面两个关系模式:

职工 (职工号, 姓名, 出生日期, 职务, 工资, 部门号), 其中职工号为主码;

部门 (部门号, 名称, 经理姓名, 电话), 其中部门号为主码.

用 SQL 定义这两个关系模式, 要求在模式中完成以下完整性约束的定义:

- (1) 定义每个模式的主码.
- (2) 定义参照完整性约束.
- (3) 定义职工年龄不超过 65 岁.

解答.

```
1  /* 职工表 */
2  CREATE TABLE EMP (
3      Empno NUMBER(4) PRIMARY KEY,
4      Ename VARCHAR(10),
5      Birthday DATE,
6      CONSTRAINT C1 CHECK (
7          FLOOR(DATEDIFF(CURDATE(), Birthday) / 365) <= 65
8      ),
9      Job VARCHAR(9),
10     Sal NUMBER(7, 2),
11     Deptno NUMBER(2),
12     CONSTRAINT FK_DEPTNO FOREIGN KEY (Deptno) REFERENCES DEPT (Deptno)
13 );
14
15 /* 部门表 */
16 CREATE TABLE DEPT (
17     Deptno NUMBER(2) PRIMARY KEY,
18     Deptname VARCHAR(10),
19     Manager VARCHAR(10),
20     PhoneNumber CHAR(12)
21 );
```