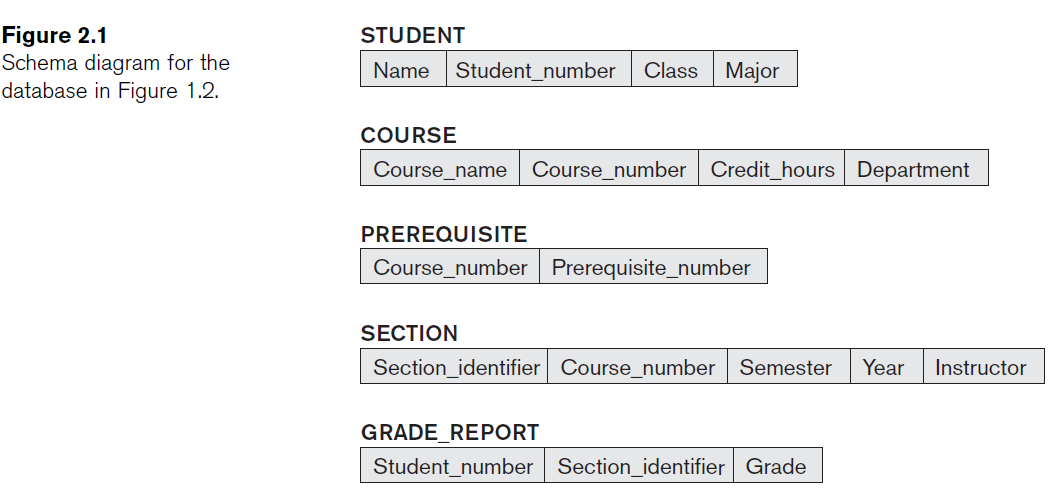
**Chapter 4: Basic SQL**



**4.5Consider the database shown in Figure 1.2, whose schema is shown in Figure 2.1. What are the referential integrity constraints that should hold on the schema? Write appropriate SQL DDL statements to define the database.**

**CREATE TABLE STUDENT** (Name **VARCHAR(30)** **NOT NULL**,

StudentNumber **INTEGER NOT NULL**,

Class **CHAR NOT NULL**,

Major **CHAR(4),**

**PRIMARY KEY** (StudentNumber));

**CREATE TABLE COURSE** ( CourseName **VARCHAR(30) NOT NULL**,

CourseNumber **CHAR(8) NOT NULL**,

CreditHours **INTEGER**,

Department **CHAR(4),**

**PRIMARY** **KEY** (CourseNumber),

**UNIQUE** (CourseName));

**CREATE TABLE PREREQUISITE** ( CourseNumber **CHAR(8)** **NOT NULL**,

PrerequisiteNumber **CHAR(8) NOT NULL**,

**PRIMARY KEY (**CourseNumber, PrerequisiteNumber),

**FOREIGN KEY** (CourseNumber) **REFERENCES**

**COURSE** (CourseNumber),

**FOREIGN KEY** (PrerequisiteNumber) **REFERENCES**

**COURSE** (CourseNumber));

**CREATE TABLE SECTION** ( SectionIdentifier) **INTEGER NOT NULL,**

CourseNumber **CHAR(8) NOT NULL**,

Semester **VARCHAR(6) NOT NULL,**

Year **CHAR(4) NOT NULL,**

Instructor **VARCHAR(15),**

**PRIMARY KEY** (SectionIdentifier),

**FOREIGN KEY** (CourseNumber) **REFERENCES**

**COURSE** (CourseNumber) );

**CREATE TABLE GRADE\_REPORT** ( StudentNumber **INTEGER NOT NULL,**

SectionIdentifier **INTEGER NOT NULL**,

Grade CHAR,

**PRIMARY KEY** (StudentNumber, SectionIdentifier),

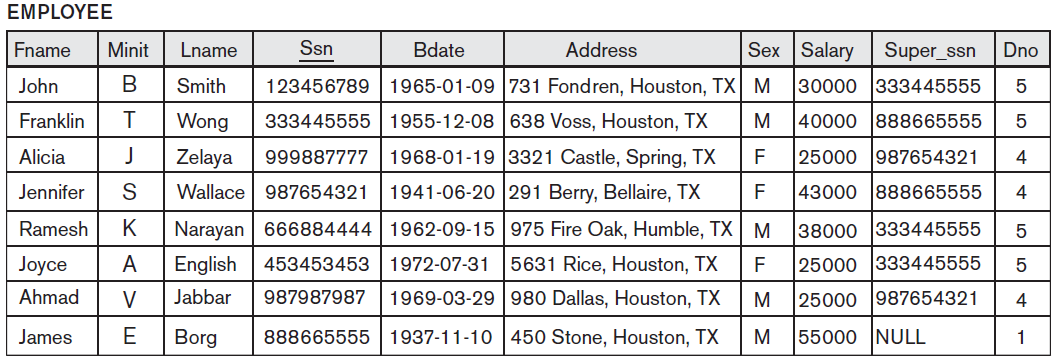
**FOREIGN KEY** (StudentNumber) **REFERENCES**

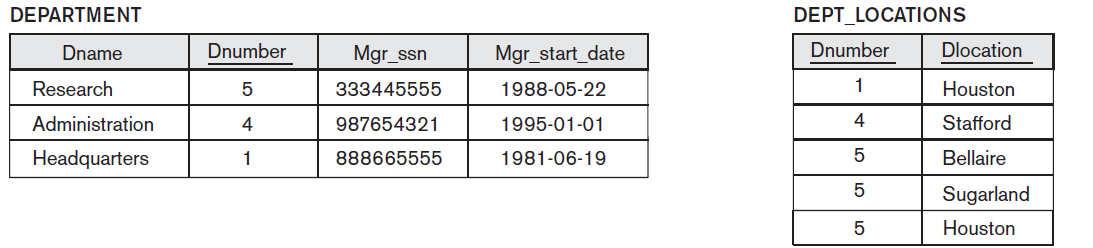
**STUDENT** (StudentNumber),

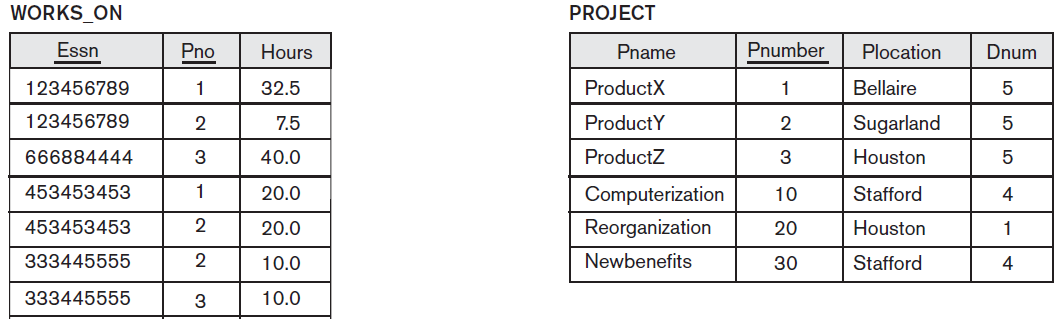
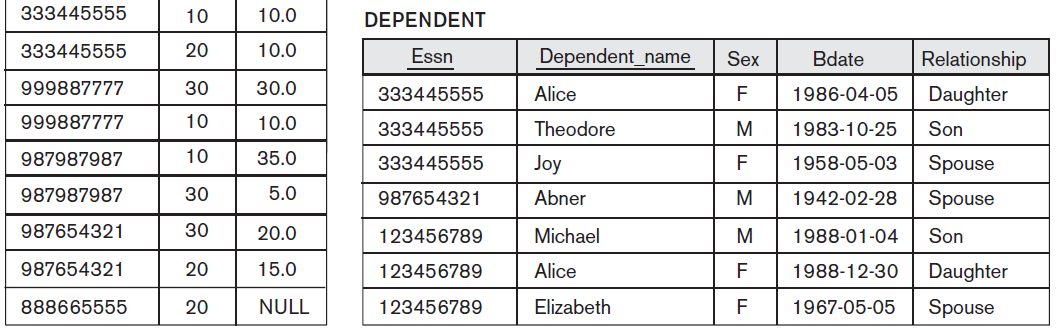
**FOREIGN KEY** (SectionIdentifier) **REFERENCES**

**SECTION** (SectionIdentifier));

1. **Figure:** One possible database state for the COMPANY relational database schema.





- 

**Specify the updates for the below using the SQL update commands.**

1. Insert <‘Robert’, ‘F’, ‘Scott’, ‘943775543’, ‘1972-06-21’, ‘2365 Newcastle Rd,

Bellaire, TX’, M, 58000, ‘888665555’, 1> into EMPLOYEE.

Answer: **INSERT INTO EMPLOYEE**

**VALUES** ('Robert', 'F', 'Scott', '943775543', '21-JUN-42', '2365 Newcastle Rd, Bellaire, TX',

M, 58000, '888665555', 1)

1. Insert <‘ProductA’, 4, ‘Bellaire’, 2> into PROJECT.

Answer: **INSERT INTO PROJECT**

**VALUES** ('ProductA', 4, 'Bellaire', 2)

1. Insert <‘Production’, 4, ‘943775543’, ‘2007-10-01’> into DEPARTMENT.

Answer: **INSERT INTO DEPARTMENT**

**VALUES** ('Production', 4, '943775543', '01-OCT-88')

1. Insert <‘677678989’, NULL, ‘40.0’> into WORKS\_ON.

Answer: **INSERT INTO WORKS\_ON**

**VALUES** ('677678989', NULL, '40.0')

1. Insert <‘453453453’, ‘John’, ‘M’, ‘1990-12-12’, ‘spouse’> into DEPENDENT.

Answer: **INSERT INTO DEPENDENT**

**VALUES** ('453453453', 'John', M, '12-DEC-60', 'SPOUSE')

1. Delete the WORKS\_ON tuples with Essn = ‘333445555’.

Answer**: DELETE FROM WORKS\_ON**

**WHERE ESSN**= '333445555'

1. Delete the EMPLOYEE tuple with Ssn = ‘987654321’.

Answer: **DELETE FROM EMPLOYEE**

**WHERE SSN**= '987654321'

1. Delete the PROJECT tuple with Pname = ‘ProductX’.

Answer: **DELETE FROM PROJECT**

**WHERE PNAME**= 'ProductX'

1. Modify the Mgr\_ssn and Mgr\_start\_date of the DEPARTMENT tuple with Dnumber = 5 to ‘123456789’ and ‘2007-10-01’, respectively.

Answer: **UPDATE DEPARTMENT**

**SET MGRSSN** = '123456789', MGRSTARTDATE = '01-OCT-88'

**WHERE DNUMBER**= 5

1. Modify the Super\_ssn attribute of the EMPLOYEE tuple with Ssn = ‘999887777’ to ‘943775543’.

Answer: **UPDATE EMPLOYEE**

**SET SUPERSSN** = '943775543'

**WHERE SSN**= '999887777'

1. Modify the Hours attribute of the WORKS\_ON tuple with Essn = ‘999887777’ and Pno = 10 to ‘5.0’.

Answer: **UPDATE WORKS\_ON**

**SET HOURS** = '5.0'

**WHERE ESSN**= '999887777' AND PNO= 10

**4.10: Specify the following queries in SQL on the COMPANY relational database schema shown in Figure 3.5. Show the result of each query if it is applied to the COMPANY database in Figure 3.6.**

**a. Retrieve the names of all employees in department 5 who work more than 10 hours per week on the ProductX project.**

**b. List the names of all employees who have a dependent with the same first name as themselves.**

**c. Find the names of all employees who are directly supervised by ‘Franklin Wong’.**

**Ans:**

a)Query:

SELECT FNAME,MINIT,LNAME

FROM EMPLOYEE E, PROJECT P, WORKS\_ON W

WHERE E.SSN=W.ESSN

AND       W.PNO=P.PNUMBER

AND       DNO=5

AND       HOURS >10

AND       PNAME='ProductX';

Result)

|  |  |
| --- | --- |
| FNAME | LNAME |
| John | Smith |
| Joyce | English |

b) Query:

SELECT FNAME,MINIT,LNAME

FROM EMPLOYEE E, DEPENDENT D

WHERE E.SSN=D.ESSN

AND E. FNAME=D.DEPENDENT\_NAME;

|  |  |
| --- | --- |
| FNAME | LNAME |
|  |  |

c) Query:

SELECT E.FNAME,E.MINIT,E.LNAME

FROM EMPLOYEE E, EMPLOYEE S

WHERE E.SUPERSSN=S.SSN

AND       S.FNAME='Franklin'

AND       S.LNAME='Wong';

Result:

|  |  |
| --- | --- |
| FNAME | LNAME |
| John | Smith |
| Ramesh | Narayan |
| Joyce | English |

**4.12: Specify the following queries in SQL on the database schema of Figure 1.2.**

**a. Retrieve the names of all senior students majoring in ‘CS’ (computer science).**

**b. Retrieve the names of all courses taught by Professor King in 2007 and 2008.**

**c. For each section taught by Professor King, retrieve the course number, semester, year, and number of students who took the section.**

**d. Retrieve the name and transcript of each senior student (Class = 4) majoring in CS. A transcript includes course name, course number, credit hours, semester, year, and grade for each course completed by the student.**

**Ans:**

a)

SELECT Name

FROM STUDENT

WHERE Major='CS'

b)

SELECT Course\_Name

FROM COURSE, SECTION

WHERE COURSE.Course\_Number=SECTION.Course\_Number AND Instructor='King'

AND (Year='07' OR Year=’08’)

c)

SELECT Course\_Number, Semester, Year, COUNT(\*)

FROM SECTION, GRADE\_REPORT

WHERE Instructor='King' AND SECTION.SectionIdentifier=GRADE\_REPORT.SectionIdentifier

GROUP BY Course\_Number, Semester, Year

d)

SELECT Name, Course\_Name, C.Course\_Number, Credit\_Hours, Semester, Year, Grade

FROM STUDENT ST, COURSE C, SECTION S, GRADE\_REPORT G

WHERE Class=4 AND Major='CS' AND ST.Student\_Number=G.Student\_Number AND

G.Section\_Identifier=S.Section\_Identifier AND S.Course\_Number=C.Course\_Number

**4.13: Write SQL update statements to do the following on the database schema shown in Figure 1.2.**

**a. Insert a new student, <‘Johnson’, 25, 1,‘Math’>, in the database.**

**b. Change the class of student ‘Smith’ to 2.**

**c. Insert a new course, <‘Knowledge Engineering’, ‘CS4390’, 3, ‘CS’>.**

**d. Delete the record for the student whose name is ‘Smith’ and whose student number is 17.**

**Ans:**

a)

INSERT INTO STUDENT

VALUES ('Johnson', 25, 1, 'MATH')

b)

UPDATE STUDENT

SET CLASS = 2

WHERE Name='Smith'

c)

INSERT INTO COURSE

VALUES ('Knowledge Engineering','CS4390', 3,'CS')

d)

DELETE FROM STUDENT

WHERE Name='Smith' AND Student\_Number=17

**4.15: Consider the EMPLOYEE table’s constraint EMPSUPERFK as specified in Figure 4.2 is changed to read as follows:**

**CONSTRAINT EMPSUPERFK**

**FOREIGN KEY (Super\_ssn) REFERENCES EMPLOYEE(Ssn)**

**ON DELETE CASCADE ON UPDATE CASCADE)**

**Answer the following questions:**

**a. What happens when the following command is run on the database state shown in Figure 3.6? DELETE EMPLOYEE WHERE Lname = ‘Borg’**

**b. Is it better to CASCADE or SET NULL in case of EMPSUPERFK constraint ON DELETE?**

**Ans:**

a) The database is affected in the way that 8 rows are deleted and the table is empty. The James E. Borg entry is deleted from the table, and each employee related to him will also be deleted from all the tables.

b) It is better to SET NULL, since an employee is not fired (DELETED) when their supervisor is deleted. Instead, their SUPERSSN should be set as SET NULL so that a new supervisor can be assigned to them.