Relational schema for SQL queries
[from Elmasri, R & Navathe, S (2000), Fundamentals of Database Systems (4th ed), Addison Wesley]

EMPLOYEE (FNAME, MINIT, LNAME, SSN, BDATE, ADDRESS, SEX, SALARY, #SUPERSSN, #DNO)

DEPARTMENT (DNAME, DNUMBER, #MGRSSN, MGRSTARTDATE)

DEPT_LOCATIONS (#DNUMBER, DLOCATION)

PROJECT (PNAME, PNUMBER, PLOCATION, #DNUM)

WORKS_ON (#ESSN, #PNO, HOURS)

DEPENDENT (#ESSN, DEPENDENT_NAME, SEX, BDATE, RELATIONSHIP)

SQL queries

Query 0

Retrieve the birthdate and address of the employee(s) whose name is ‘John B Smith’

```
SELECT      BDATE, ADDRESS
FROM        EMPLOYEE
WHERE       FNAME = ‘John’ AND MINIT = ‘B’ AND LNAME = ‘Smith’;
```

Query 1

Retrieve the name and address of all employees who work for the ‘Research’ department

```
SELECT      FNAME, LNAME, ADDRESS
FROM        EMPLOYEE, DEPARTMENT
WHERE       DNAME = ‘Research’ AND DNUMBER = DNO;
```

Query 1A

Ambiguous attribute names

```
SELECT      FNAME, EMPLOYEE.NAME, ADDRESS
FROM        EMPLOYEE, DEPARTMENT
WHERE       DEPARTMENT.NAME = ‘Research’ AND
            DEPARTMENT.DNUMBER = EMPLOYEE.DNUMBER;
```

Query 1B

Aliasing

```
SELECT      E.FNAME, E.NAME, E.ADDRESS
FROM        EMPLOYEE E, DEPARTMENT D
WHERE       D.NAME = ‘Research’ AND D.DNUMBER = E.DNUMBER;
```

Query 1C

Retrieve all the attribute values of EMPLOYEE tuples who work in department number 5

```
SELECT      *
FROM        EMPLOYEE
WHERE       DNO = 5;
```

Query 1D

Retrieve all the attributes of an EMPLOYEE and the attributes of the DEPARTMENT he or she works in for every employee of the ‘Research’ department

```
SELECT      *
FROM        EMPLOYEE, DEPARTMENT
WHERE       DNAME = ‘Research’ AND DNO = DNUMBER;
```

Query 2

For every project located in ‘Stafford’, list the project number, the controlling department number and the department manager’s last name, address and birthdate

```
SELECT      PNUMBER, DNUM, LNAME, ADDRESS, BDATE
FROM        PROJECT, DEPARTMENT, EMPLOYEE
WHERE       DNUM = DNUMBER AND MGRSSN = SSN AND PLOCATION = ‘Stafford’;
```

Query 3

Retrieve the name of each employee who works on *all* the projects controlled by department number 5

```
SELECT          FNAME, LNAME
FROM            EMPLOYEE
WHERE           (       (SELECT         PNO
                        FROM            WORKS_ON
                        WHERE           SSN = ESSN)

                        CONTAINS

                        (SELECT         PNUMBER
                        FROM            PROJECT
                        WHERE           DNUM = 5)       );
```

Query 3A

Reformulation of query 3.

```
SELECT          FNAME, LNAME
FROM            EMPLOYEE
WHERE           NOT EXISTS
                (       (SELECT         PNUMBER
                        FROM            PROJECT
                        WHERE           DNUM = 5)

                        EXCEPT

                        (SELECT         PNO
                        FROM            WORKS_ON
                        WHERE           SSN = ESSN)     );
```

Query 3B

Reformulation of query 3 using two-level nesting

```
SELECT          LNAME, FNAME
FROM            EMPLOYEE
WHERE           NOT EXISTS
                (SELECT         *
                FROM            WORKS_ON B
                WHERE           B.PNO IN        (SELECT         PNUMBER
                                                FROM            PROJECT
                                                WHERE           DNUM = 5)       )

                                AND

                                NOT EXISTS      (SELECT         *
                                                FROM            WORKS_ON C
                                                WHERE           C.ESSN = SSN
                                                                AND
                                                                C.PNO = B.PNO));
```

Query 4

Make a list of project numbers for projects that involve an employee whose last name is ‘Smith’, either as a worker or as a manager of the department that controls the project

```
(SELECT     DISTINCT PNUMBER
FROM        PROJECT, DEPARTMENT, EMPLOYEE
WHERE       DNUM = DNUMBER AND MGRSSN = SSN AND LNAME = ‘Smith’)
UNION
(SELECT     DISTINCT PNUMBER
FROM        PROJECT, WORKS_ON, EMPLOYEE
WHERE       PNUMBER = PNO AND ESSN = SSN AND LNAME = ‘Smith’);
```

Query 4A

Reformulation of query 4 using nested queries

```
SELECT      DISTINCT PNUMBER
FROM        PROJECT
WHERE       PNUMBER IN  (SELECT     PNUMBER
                        FROM        PROJECT, DEPARTMENT, EMPLOYEE
                        WHERE       DNUM = DNUMBER AND MGRSSN = SSN AND
                                    LNAME = ‘Smith’)
            OR
            PNUMBER IN  (SELECT     PNO
                        FROM        WORKS_ON, EMPLOYEE
                        WHERE       ESSN = SSN AND LNAME = ‘Smith’);
```

Query 5

Retrieve the names of all employees who have two or more dependents

```
SELECT      LNAME, FNAME
FROM        EMPLOYEE
WHERE       (SELECT     COUNT (*)
            FROM        DEPENDENT
            WHERE       SSN = ESSN) >= 2;
```

Query 6

Retrieve the names of employees who have no dependents

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE
WHERE       NOT EXISTS        (SELECT     *
                              FROM        DEPENDENT
                              WHERE       SSN = ESSN);
```

Query 7

List the names of managers who have at least one dependent

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE
WHERE       EXISTS      (SELECT     *
                         FROM       DEPENDENT
                         WHERE      SSN = ESSN)

            AND

            EXISTS      (SELECT     *
                         FROM       DEPARTMENT
                         WHERE      SSN = MGRSSN);
```

Query 8

For each employee, retrieve the employee’s first and last name, and the first and last name of his or her immediate supervisor.

```
SELECT      E.FNAME, E.LNAME, S.FNAME, S.LNAME
FROM        EMPLOYEE AS E, EMPLOYEE AS S
WHERE       E.SUPERSSN = S.SSN;
```

Query 8A

Reformulation of query 8 to retrieve the last name of each employee and his or her supervisor, while renaming the resulting attribute names as EMPLOYEE_NAME and SUPERVISOR_NAME

```
SELECT      E.LNAME AS EMPLOYEE_NAME, S.LNAME AS SUPERVISOR_NAME
FROM        EMPLOYEE AS E, EMPLOYEE AS S
WHERE       E.SUPERSSN = S.SSN;
```

Query 9

Select all EMPLOYEE SSNs in the database

```
SELECT      SSN
FROM        EMPLOYEE;
```

Query 10

Select all combinations of EMPLOYEE SSN and DEPARTMENT DNAME in the database

```
SELECT      SSN, DNAME
FROM        EMPLOYEE, DEPARTMENT;
```

Query 10A

Select the CROSS PRODUCT of the EMPLOYEE and DEPARTMENT relations

```
SELECT      *
FROM        EMPLOYEE, DEPARTMENT;
```

Query 11

Retrieve the salary of every employee

```
SELECT ALL  SALARY
FROM        EMPLOYEE;
```

Query 11A

Retrieve all distinct salary values

```
SELECT DISTINCT    SALARY
FROM               EMPLOYEE;
```

Query 12

Retrieve all employees whose address is in Houston, Texas

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE
WHERE       ADDRESS LIKE ‘%Houston,TX%’;
```

Query 12A

Find all employees who were born during the 1950s

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE
WHERE       BDATE LIKE ‘_ _ _ _ _ _ _ 195 _’;
```

Query 13

Show the resulting salaries if every employee working on the ‘ProductX’ project is given a 10 percent raise

```
SELECT      FNAME, LNAME, 1.1*SALARY
FROM        EMPLOYEE, WORKS_ON, PROJECT
WHERE       SSN = ESSN AND PNO = PNUMBER AND PNAME = ‘ProductX’;
```

Query 14

Retrieve all employees in department 5 whose salary is between £30,000 and £40,000

```
SELECT      *
FROM        EMPLOYEE
WHERE       (SALARY BETWEEN 30000 AND 40000) AND DNO = 5;
```

Query 15

Retrieve a list of employees and the projects they are working on, ordered by department and, within each department, ordered alphabetically by last name, first name

```
SELECT      DNAME, LNAME, FNAME, PNAME
FROM        DEPARTMENT, EMPLOYEE,WORKS_ON, PROJECT
WHERE       DNUMBER = DNO AND SSN = ESSN AND PNO = PNUMBER
ORDER BY    DNAME DESC, LNAME ASC, FNAME ASC;
```

Query 16

Retrieve the name of each employee who has a dependent with the same first name and same sex as the employee

```
SELECT      E.FNAME, E.LNAME
FROM        EMPLOYEE AS E
WHERE       E.SSN IN    (SELECT     ESSN
                         FROM       DEPENDENT
                         WHERE      E.FNAME = DEPENDENT_NAME AND E.SEX = SEX);
```

Query 16A

Reformulation of query 16 as single block query

```
SELECT      E.FNAME, E.LNAME
FROM        EMPLOYEE AS E, DEPENDENT AS D
WHERE       E.SSN = D.ESSN AND E.SEX = D.SEX AND E.FNAME = D.DEPENDENT_NAME;
```

Query 16B

Reformulation of query 16 using EXISTS

```
SELECT      E.FNAME, E.LNAME
FROM        EMPLOYEE AS E
WHERE       EXISTS      (SELECT     *
                        FROM        DEPENDENT
                        WHERE       E.SSN = ESSN AND E.SEX = SEX AND
                                    E.FNAME = DEPENDENT_NAME);
```

Query 17

Retrieve the social security numbers of all employees who work on project number 1, 2 or 3

```
SELECT      DISTINCT ESSN
FROM        WORKS_ON
WHERE       PNO IN (1,2,3);
```

Query 18

Retrieve the names of all employees who do not have supervisors

```
SELECT      FNAME, LNAME
FROM        EMPLOYEE
WHERE       SUPERSSN IS NULL;
```

Query 19

Find the sum of the salaries of all employees, the maximum salary, the minimum salary, and the average salary

```
SELECT      SUM (SALARY), MAX (SALARY), MIN (SALARY), AVG (SALARY)
FROM        EMPLOYEE;
```

Query 20

Find the sum of the salaries of all employees of the ‘Research’ department, as well as the maximum salary, the minimum salary, and the average salary in this department

```
SELECT      SUM (SALARY), MAX (SALARY), MIN (SALARY), AVG (SALARY)
FROM        EMPLOYEE, DEPARTMENT
WHERE       DNO = DNUMBER AND DNAME = ‘Research’;
```

Query 21

Retrieve the total number of employees in the company

```
SELECT      COUNT (*)
FROM        EMPLOYEE;
```

Query 22

Retrieve the number of employees in the ‘Research’ department

```
SELECT      COUNT (*)
FROM        EMPLOYEE, DEPARTMENT
WHERE       DNO = DNUMBER AND DNAME = ‘Research’;
```

Query 23

Count the number of distinct salary values in the database

```
SELECT      COUNT (DISTINCT SALARY)
FROM        EMPLOYEE;
```

Query 24

For each department, retrieve the department number, the number of employees in the department, and their average salary

```
SELECT      DNO, COUNT (*), AVG (SALARY)
FROM        EMPLOYEE
GROUP BY    DNO;
```

Query 25

For each project, retrieve the project number, the project name, and the number of employees who work on that project

```
SELECT      PNUMBER, PNAME, COUNT (*)
FROM        PROJECT, WORKS_ON
WHERE       PNUMBER = PNO
GROUP BY    PNUMBER, PNAME;
```

Query 26

For each project on which more than two employees work, retrieve the project number, the project name, and the number of employees who work on the project

```
SELECT      PNUMBER, PNAME, COUNT (*)
FROM        PROJECT, WORKS_ON
WHERE       PNUMBER = PNO
GROUP BY    PNUMBER
HAVING      COUNT (*) > 2;
```

Query 27

For each project, retrieve the project number, the project name, and the number of employees from department 5 who work on the project

```
SELECT      PNUMBER, PNAME, COUNT (*)
FROM        PROJECT, WORKS_ON, EMPLOYEE
WHERE       PNUMBER = PNO AND SSN = ESSN AND DNO = 5
GROUP BY    PNUMBER, PNAME;
```

Query 28

For each department that has more than five employees, retrieve the department number and the number of its employees who are making more than £40,000

```
SELECT      DNUMBER, COUNT (*)
FROM        DEPARTMENT, EMPLOYEE
WHERE       DNUMBER = DNO AND SALARY > 40000 AND
            DNO IN      (SELECT     DNO
                         FROM       EMPLOYEE
                         GROUP BY   DNO
                         HAVING     COUNT (*) > 5)
GROUP BY    DNUMBER;
```

Data definition

```
CREATE TABLE EMPLOYEE
        (FNAME              VARCHAR(15)         NOT NULL,
        MINIT               CHAR                DEFAULT NULL,
        LNAME               VARCHAR(15)         NOT NULL,
        SSN                 CHAR(9)             NOT NULL,
        BDATE               DATE                DEFAULT NULL,
        ADDRESS             VARCHAR(30)         DEFAULT NULL,
        SEX                 CHAR                DEFAULT NULL,
        SALARY              DECIMAL(10,2)       DEFAULT NULL,
        SUPERSSN            CHAR(9)             DEFAULT NULL,
        DNO                 INT                 NOT NULL            DEFAULT 1,
        PRIMARY KEY (SSN),
        FOREIGN KEY (SUPERSSN) REFERENCES EMPLOYEE (SSN),
        FOREIGN KEY (DNO) REFERENCES DEPARTMENT (DNUMBER) );

CREATE TABLE DEPARTMENT
        (DNAME              VARCHAR(15)         UNIQUE,
        DNUMBER             INT                 PRIMARY KEY,
        MGRSSN              CHAR(9)             NOT NULL,
        MGRSTARTDATE        DATE                DEFAULT NULL,
        FOREIGH KEY (MGRSSN) REFERENCES EMPLOYEE (SSN) );

CREATE TABLE DEPT_LOCATIONS
        (DNUMBER            INT                 NOT NULL,
        DLOCATION           VARCHAR(15)         NOT NULL,
        PRIMARY KEY (DNUMBER, DLOCATION),
        FOREIGN KEY (DNUMBER) REFERENCES DEPARTMENT (DNUMBER) );

CREATE TABLE PROJECT
        (PNAME              VARCHAR(15)         NOT NULL,
        PNUMBER             INT                 NOT NULL,
        PLOCATION           VARCHAR(15)         DEFAULT NULL,
        DNUM                INT                 NOT NULL,
        PRIMARY KEY (PNUMBER),
        UNIQUE (PNAME),
        FOREIGN KEY (DNUM) REFERENCES DEPARTMENT (DNUMBER) );

CREATE TABLE WORKS_ON
        (ESSN               CHAR(9)             NOT NULL,
        PNO                 INT                 NOT NULL,
        HOURS               DECIMAL(3,1)        NOT NULL,
        PRIMARY KEY (ESSN, PNO),
        FOREIGN KEY (ESSN) REFERENCES EMPLOYEE (SSN),
        FOREIGN KEY (PNO) REFERENCES PROJECT (PNUMBER) );

CREATE TABLE DEPENDENT
        (ESSN               CHAR(9)             NOT NULL,
        DEPENDENT_NAME      VARCHAR(15)         NOT NULL,
        SEX                 CHAR                DEFAULT NULL,
        BDATE               DATE                DEFAULT NULL,
        RELATIONSHIP        VARCHAR(8)          DEFAULT NULL,
        PRIMARY KEY (ESSN, DEPENDENT_NAME),
        FOREIGN KEY (ESSN) REFERENCES EMPLOYEE (SSN) );
```

Update statements

Update 1

Add a new tuple to the EMPLOYEE relation

```
INSERT INTO  EMPLOYEE
VALUES       (‘Richard’, ‘K’, ‘Marini’, ‘653298653’, ‘1962-12-30’, ’98 Oak Forest, Katy, TX’, ‘M’, 37000,
             ‘987654321’, 4);
```

Update 1A

Enter a tuple for a new employee for whom only FNAME, LNAME, DNO AND SSN attributes are known

```
INSERT INTO  EMPLOYEE (FNAME, LNAME, DNO, SSN)
VALUES       (‘Richard’, ‘Marini’, 4, ‘653298653’);
```

Update 2

Rejected if referential integrity constraint enforced by DBMS and no department tuple with DNUMBER 2 exists

```
INSERT INTO  EMPLOYEE (FNAME, LNAME, SSN, DNO)
VALUES       (‘Robert’, ‘Hatcher’, ‘980760540’, 2);
```

Update 2A

Rejected if NOT NULL checking enforced by DBMS (SSN not present)

```
INSERT INTO  EMPLOYEE (FNAME, LNAME, DNO)
VALUES       (‘Robert’, ‘Hatcher’, 5);
```

Update 3A / B

Create a temporary table that has the name, number of employees and total salaries for each department

```
CREATE TABLE        DEPTS_INFO
(DEPT_NAME          VARCHAR(15),
NO_OF_EMPS          INTEGER,
TOTAL_SAL           INTEGER);

INSERT INTO         DEPTS_INFO (DEPT_NAME, NO_OF_EMPS, TOTAL_SAL)
SELECT              DNAME, COUNT (*), SUM (SALARY)
FROM                (DEPARTMENT JOIN EMPLOYEE ON DNUMBER = DNO)
GROUP BY            DNAME;
```

Update 4A

Deletion of tuples

```
DELETE FROM         EMPLOYEE
WHERE               LNAME = ‘Brown’;
```

Update 4B

Deletion of tuples

```
DELETE FROM         EMPLOYEE
WHERE               SSN = ‘123456789’;
```

Update 4C

Deletion of tuples

```
DELETE FROM     EMPLOYEE
WHERE           DNO IN      (SELECT     DNUMBER
                             FROM       DEPARTMENT
                             WHERE      DNAME = ‘Research’);
```

Update 4D

Deletion of tuples

```
DELETE FROM     EMPLOYEE
```

Update 5

Change the location and controlling department number of project number 10

```
UPDATE      PROJECT
SET         PLOCATION = ‘Bellaire’, DNUM = 5
WHERE       PNUMBER = 10;
```

Update 6

Give all employees in the Research department a 10% raise in salary

```
UPDATE      EMPLOYEE
SET         SALARY = SALARY * 1.1
WHERE       DNO IN      (SELECT     DNUMBER
                         FROM       DEPARTMENT
                         WHERE      DNAME = ‘Research’);
```

View statements

View 1

```
CREATE VIEWWORKS_ON1
AS      SELECT      FNAME, LNAME, PNAME, HOURS
        FROM        EMPLOYEE, PROJECT, WORKS_ON
        WHERE       SSN = ESSN AND PNO = PNUMBER;
```

View 1A

```
DROP VIEW           WORKS_ON1;
```

View 2

```
CREATE VIEWDEPT_INFO (DEPT_NAME, NO_OF_EMPS, TOTAL_SAL)
AS      SELECT      DNAME, COUNT (*), SUM (SALARY)
        FROM        DEPARTMENT, EMPLOYEE
        WHERE       DNUMBER = DNO
        GROUP BY    DNAME;
```

Query view 1

Retrieve first name and last name of all employees who work on ‘Project X’

```
SELECT      FNAME, LNAME
FROM        WORKS_ON1
WHERE       PNAME = ‘ProjectX’;
```

Update view 1

Update the PNAME attribute of ‘John Smith’ from ‘ProductX’ to ‘ProductY’

```
UPDATE      WORKS_ON1
SET         PNAME = ‘ProductY’
WHERE       LNAME = ‘Smith’ AND FNAME = ‘John’ AND PNAME = ‘ProductX’;
```

A

```
UPDATE      WORKS_ON
SET         PNO =       (SELECT     PNUMBER
                         FROM       PROJECT
                         WHERE      PNAME = ‘ProductY’)
WHERE       ESSN IN     (SELECT     SSN
                         FROM       EMPLOYEE
                         WHERE      LNAME = ‘Smith’ AND FNAME = ‘John’)

            AND

            PNO IN      (SELECT     PNUMBER
                         FROM       PROJECT
                         WHERE      PNAME = ‘ProductX’);
```

B

```
UPDATE      PROJECT
SET         PNAME = ‘ProductY’
WHERE       PNAME = ‘ProductX’;
```

Update view 2

Does not make sense because TOTAL_SAL is the sum of individual employees’ salaries

```
UPDATE      DEPT_INFO
SET         TOTAL_SAL = 100000
WHERE       DNAME = ‘Research’;
```