Chapter 7: More SQL: Complex Queries, Triggers, Views, and Schema Modification

Database Systems CS203

By: Anam Qureshi

Outline

- Variations of Insert Operation
- Importance of Foreign Key Constraint
- Three-value logic
- Nested Queries
- Joined Tables
- Aggregate Functions
- Grouping
- Case Statements
- Assertions
- Triggers
- Views
- Modification

Insert Operation

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

INSERT INTO     EMPLOYEE (Fname, Lname, Dno, Ssn)
VALUES          ('Richard', 'Marini', 4, '653298653');

INSERT INTO     EMPLOYEE (Fname, Lname, Ssn, Dno)
VALUES          ('Robert', 'Hatcher', '980760540', 2);

INSERT INTO     EMPLOYEE (Fname, Lname, Dno)
VALUES          ('Robert', 'Hatcher', 5);
```

Insert Operation

```
CREATE TABLE    WORKS_ON_INFO
( Emp_name      VARCHAR(15),
  Proj_name     VARCHAR(15),
  Hours_per_week  DECIMAL(3,1) );

INSERT INTO     WORKS_ON_INFO ( Emp_name, Proj_name,
                Hours_per_week )
SELECT          E.Lname, P.Pname, W.Hours
FROM            PROJECT P, WORKS_ON W, EMPLOYEE E
WHERE           P.Pnumber = W.Pno AND W.Essn = E.Ssn;
```

```
CREATE TABLE    D5EMPS LIKE EMPLOYEE
(SELECT         E.*
FROM            EMPLOYEE AS E
WHERE           E.Dno = 5) WITH DATA;
```

Reading and Practice Assignment

- 6.4.2 &6.4.3
- Solve Review Questions
- Solve Exercise Questions

Importance of a Foreign Key

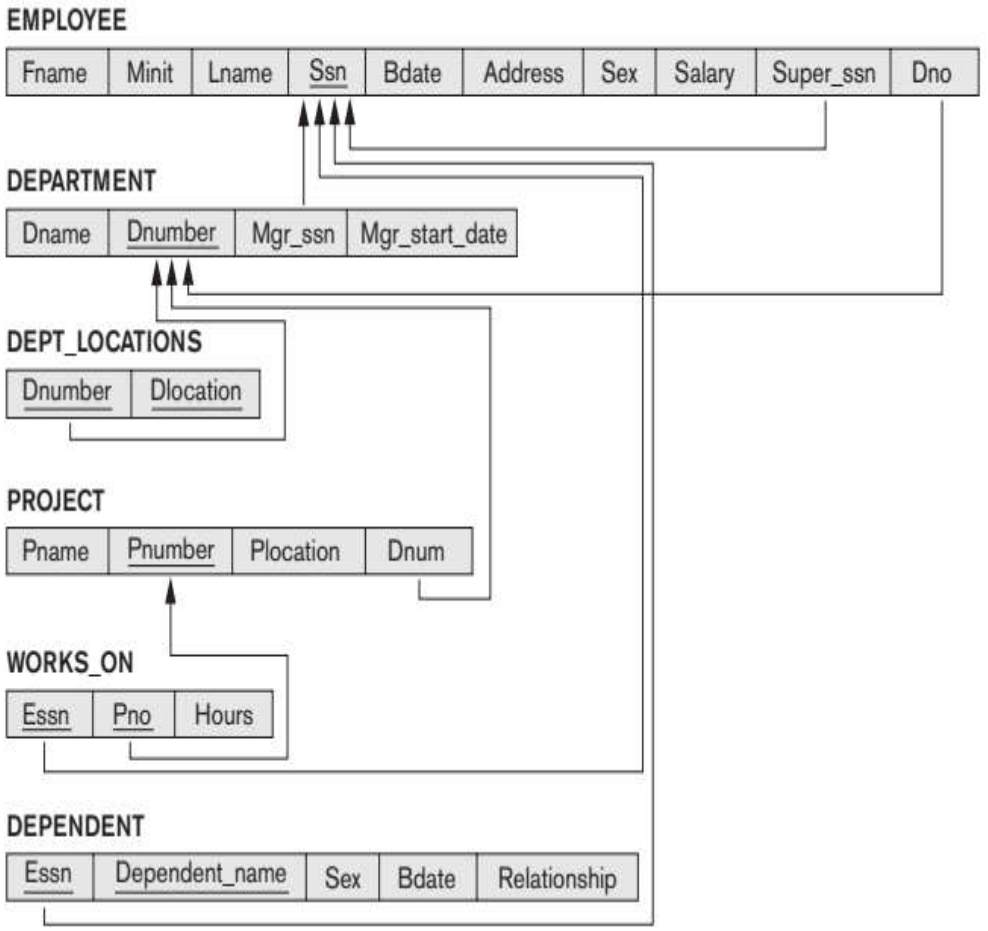
- Within Same Relation/Table

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

Importance of a Foreign Key

- In different Relations/Tables
  - Foreign key maintains referential integrity
  - One relation may contain single or multiple foreign keys
  - Used to create relationship among tables so hat join can be applied
  - To avoid referential integrity constraints violation, Use foreign key constraints as modification part (i.e., Alter table..)



NULL Constraint

1. **Unknown value.** A person's date of birth is not known, so it is represented by NULL in the database. An example of the other case of unknown would be NULL for a person's home phone because it is not known whether or not the person has a home phone.
2. **Unavailable or withheld value.** A person has a home phone but does not want it to be listed, so it is withheld and represented as NULL in the database.
3. **Not applicable attribute.** An attribute LastCollegeDegree would be NULL for a person who has no college degrees because it does not apply to that person.

Three-Value Boolean Logic

- SQL uses comparison operators IS or IS NOT rather than = or <>.
- Because SQL considers each NULL value distinct, hence equality comparison is not appropriate.
- In case of join, tuples with NULL values for the join attributes are not included in the result (unless it is an OUTER JOIN)

Table 7.1 Logical Connectives in Three-Valued Logic

(a)

AND	TRUE	FALSE	UNKNOWN
TRUE	TRUE	FALSE	UNKNOWN
FALSE	FALSE	FALSE	FALSE
UNKNOWN	UNKNOWN	FALSE	UNKNOWN

(b)

OR	TRUE	FALSE	UNKNOWN
TRUE	TRUE	TRUE	TRUE
FALSE	TRUE	FALSE	UNKNOWN
UNKNOWN	TRUE	UNKNOWN	UNKNOWN

(c)

NOT	
TRUE	FALSE
FALSE	TRUE
UNKNOWN	UNKNOWN

Retrieve the names of all employees who do not have supervisors.

```
SELECT   Fname, Lname
FROM     EMPLOYEE
WHERE    Super_ssn IS NULL;
```

Nested Queries OR Sub Queries

Nested Queries

- **Non-Correlated Sub queries**
  - •Execution process: First Inner queries are executed then their values are matched using comparison operators (IN,NOT IN, SOME, ALL, and others) with outer queries.

    Select ColumnNames From Table Names Where Column Names (=,IN,NOT IN, ALL >, SOME , etc) (Select From Where Clause)

  - •The nested/inner query may result an scalar, a row, or multiple rows.
  - • The equivalent operator '=' is used only when inner query results as scalar.
  - •Attributes of sub queries can not be used in Select Statement of outer queries

Nested Queries

- **Correlated Sub queries**
  - In correlated sub query the nested query is executed once for each tuple of an outer query.
- Sub queries can be nested inside select, where, update, insert and delete statements

Example

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

Query: Select the department names to which no employee is assigned yet.

Example

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-01-09	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
Alicia	J	Zelaya	999887777	1968-01-19	3321 Castle, Spring, TX	F	25000	987654321	4
Jennifer	S	Wallace	987654321	1941-06-20	291 Berry, Bellaire, TX	F	43000	888665555	4
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5
Joyce	A	English	453453453	1972-07-31	5631 Rice, Houston, TX	F	25000	333445555	5
Ahmad	V	Jabbar	987987987	1969-03-29	980 Dallas, Houston, TX	M	25000	987654321	4
James	E	Borg	888665555	1937-11-10	450 Stone, Houston, TX	M	55000	NULL	1

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date
Research	5	333445555	1988-05-22
Administration	4	987654321	1995-01-01
Headquarters	1	888665555	1981-06-19

Query: Retrieve the number of total employees working in each department.

Example

Query 4. Make a list of all 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
WHERE     Pnumber IN
          ( SELECT    Pnumber
            FROM      PROJECT, DEPARTMENT, EMPLOYEE
            WHERE     Dnum = Dnumber AND
                      Mgr_ssn = Ssn AND Lname = 'Smith' )
          OR
          Pnumber IN
          ( SELECT    Pno
            FROM      WORKS_ON, EMPLOYEE
            WHERE     Essn = Ssn AND Lname = 'Smith' );
```

Example

Query: Select the Essns of all employees who work the same (project,hours) combination on some project that employee ‘John Smith’(whose Ssn=‘123456789’) works on.

```
SELECT    DISTINCT Essn
FROM      WORKS_ON
WHERE     (Pno, Hours) IN    ( SELECT    Pno, Hours
                              FROM      WORKS_ON
                              WHERE     Essn = '123456789' );
```

Example

Query: List the names of employees whose salary is greater than the salary of all the employees in department 5.

```
SELECT    Lname, Fname
FROM      EMPLOYEE
WHERE     Salary > ALL    ( SELECT    Salary
                            FROM      EMPLOYEE
                            WHERE     Dno = 5 );
```

The EXISTS and UNIQUE Functions in SQL for correlating queries

`EXISTS` function

- Check whether the result of a correlated nested query is empty or not. They are Boolean functions that return a TRUE or FALSE result.

`EXISTS` and `NOT EXISTS`

- Typically used in conjunction with a correlated nested query

SQL function `UNIQUE(Q)`

- Returns `TRUE` if there are no duplicate tuples in the result of query Q