CHAPTER 6

Basic Structured Query Language (SQL)

Chapter 6 Outline

- SQL Data Definition and Data Types
- Specifying Constraints in SQL
- Basic Retrieval Queries in SQL
- INSERT, DELETE, and UPDATE Statements in SQL
- Additional Features of SQL

Structured Query Language (SQL)

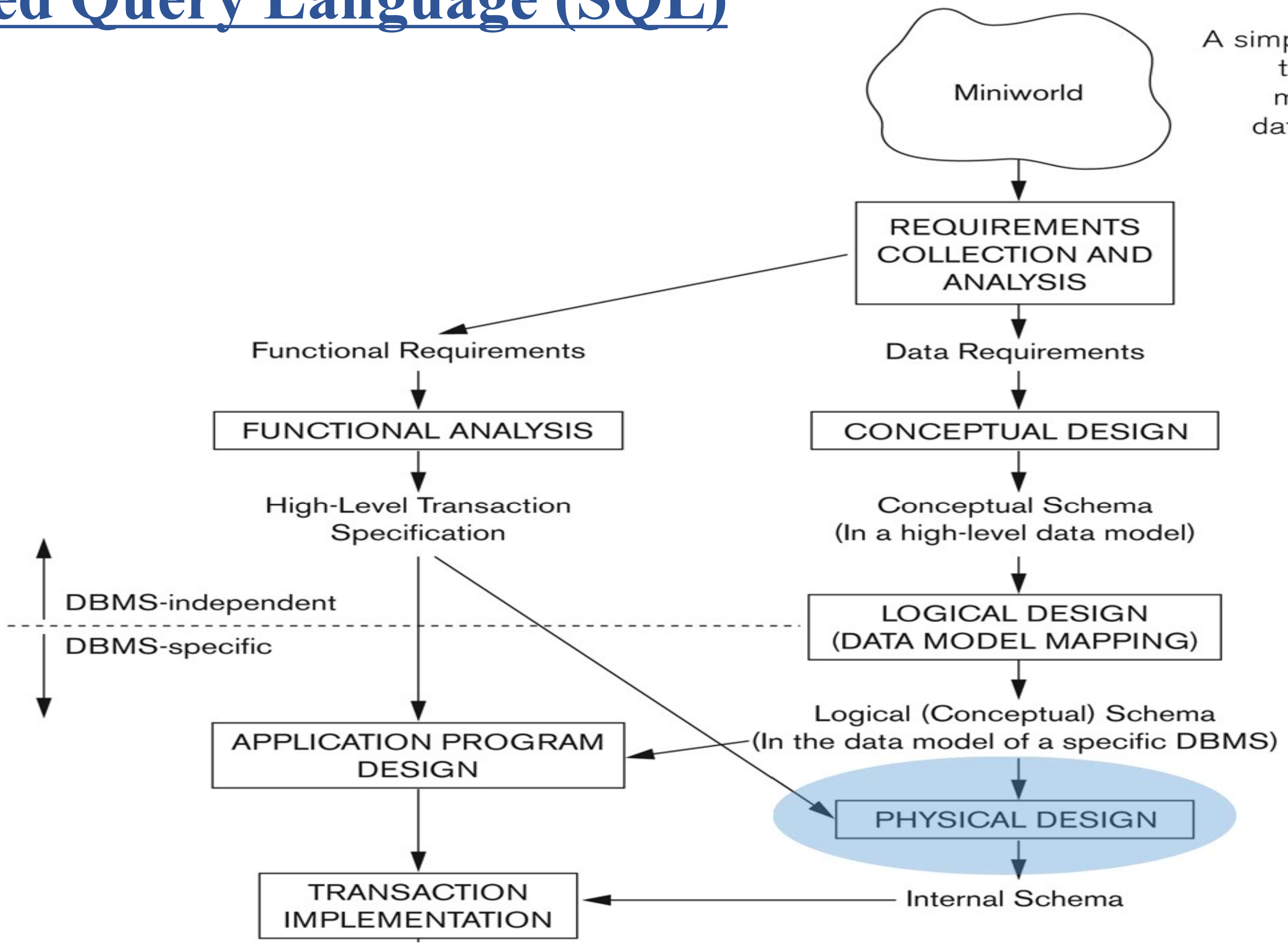
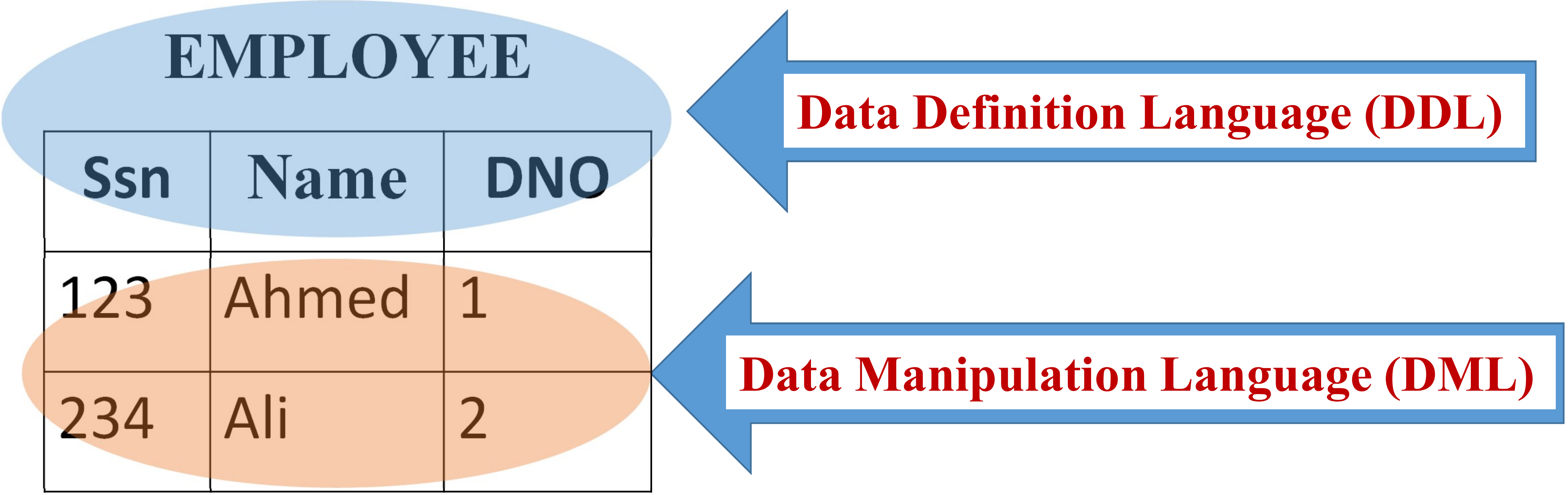

Miniworld
REQUIREMENTS COLLECTION AND ANALYSIS
Functional Requirements
Data Requirements
FUNCTIONAL ANALYSIS
CONCEPTUAL DESIGN
High-Level Transaction Specification
Conceptual Schema (In a high-level data model)
DBMS-independent
DBMS-specific
LOGICAL DESIGN (DATA MODEL MAPPING)
Logical (Conceptual) Schema (In the data model of a specific DBMS)
APPLICATION PROGRAM DESIGN
PHYSICAL DESIGN
TRANSACTION IMPLEMENTATION
Internal Schema

Figure 3.1
A simplified diagram to illustrate the main phases of database design.

data manipulation language (DML)

data definition language (DDL)



EMPLOYEE

Ssn	Name	DNO
123	Ahmed	1
234	Ali	2

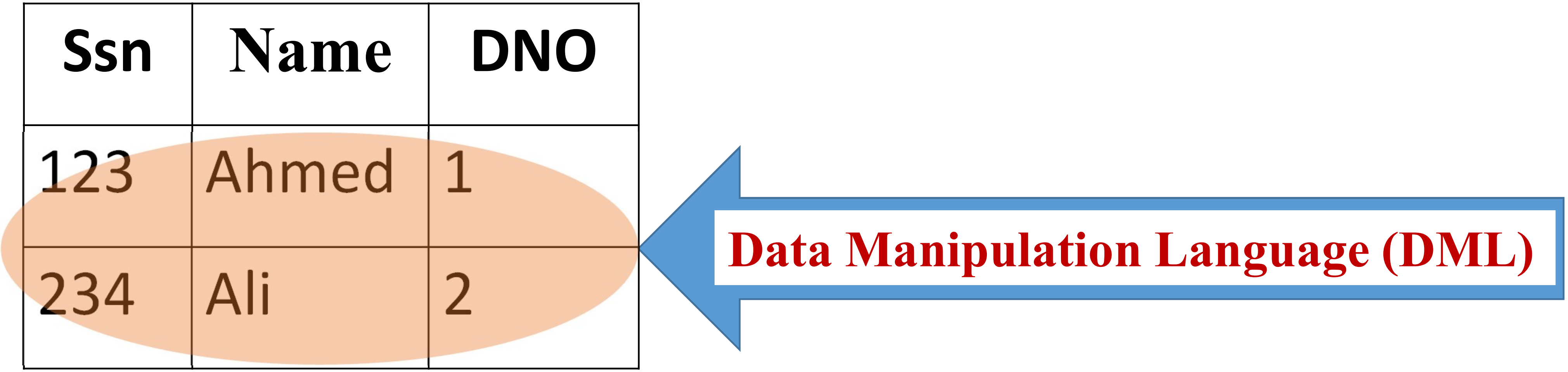
Data Definition Language (DDL)

Data Manipulation Language (DML)

Data Manipulation Language (DML)

RETRIEVE,
INSERT,
DELETE, and
UPDATE Statements in SQL

EMPLOYEE



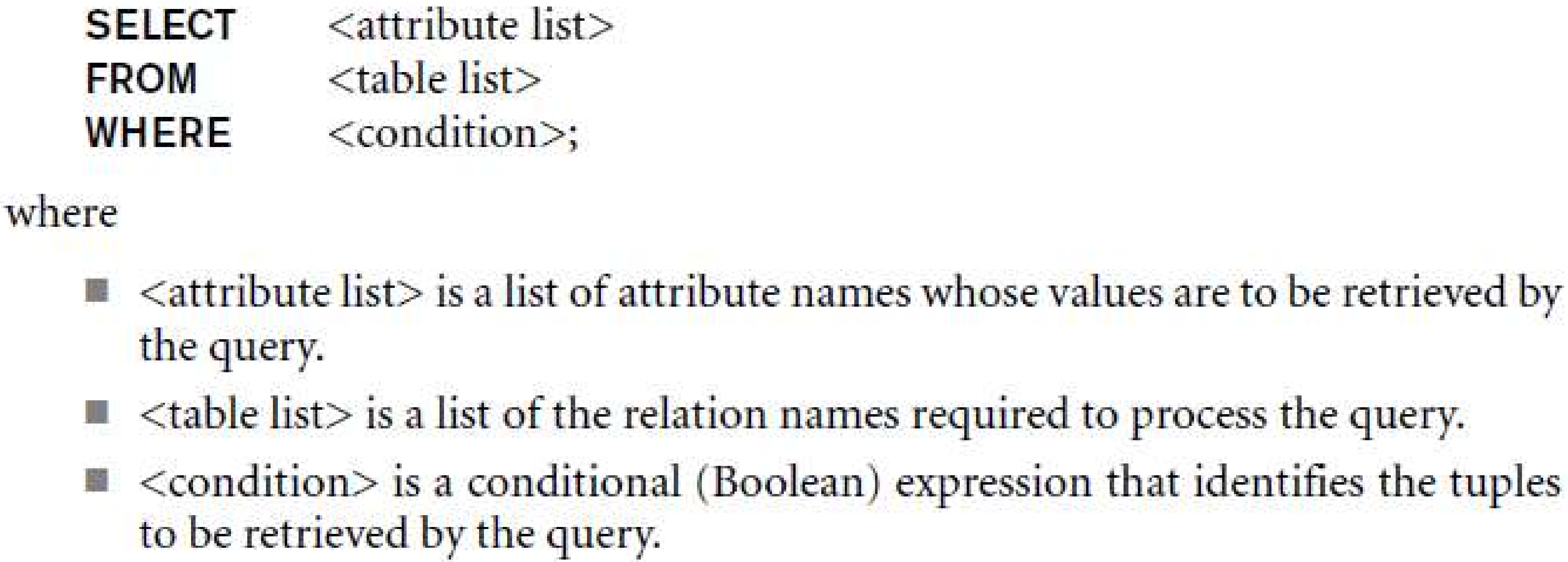
Ssn	Name	DNO
123	Ahmed	1
234	Ali	2

Data Manipulation Language (DML)

RETRIEVE, INSERT, DELETE, and UPDATE Statements in SQL

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:



```
SELECT    <attribute list>
FROM      <table list>
WHERE     <condition>;
```

where

- <attribute list> is a list of attribute names whose values are to be retrieved by the query.
- <table list> is a list of the relation names required to process the query.
- <condition> is a conditional (Boolean) expression that identifies the tuples to be retrieved by the query.

Basic Retrieval Queries in SQL

- SELECT statement
  - One basic statement for retrieving information from a database
- SQL allows a table to have two or more tuples that are identical in all their attribute values
  - Unlike relational model (relational model is strictly set-theory based)
  - Multiset or bag behavior

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
CREATE  TABLE  EMP (
        Ssn         INT,
        Name     VARCHAR(30) ,
        Dno         INT
);

CREATE  TABLE  DEPT (
        Dname         VARCHAR (25),
        Dnumber     INT
);
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM EMP ;
```

```
SSN     NAME
11      Ahmed Mohamed
12      Ali Hasan Adel
13      Mohsen Mahmod
14      Alaa Ali Mohamed
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Dname
FROM DEPT ;
```

DNAME
Research
Administration
Software

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Specify an asterisk (*)
  - Retrieve all the attribute values of the selected tuples

```
SELECT  *
FROM EMP ;
```

```
SSN     NAME                DNO
11      Ahmed Mohamed       1
12      Ali Hasan Adel 2
13      Mohsen Mahmod       1
14      Alaa Ali Mohamed    3
15      Yaser Hesein Ali    2
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE Name ='Ali Hasan Adel';
```

```
SSN    NAME
12       Ali Hasan Adel
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  DNO=1;
```

- **Selection condition**
  - Boolean condition that must be true for any retrieved tuple

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Logical comparison operators

Operator	Description
=	Equal
>	Greater than
<	Less than
>=	Greater than or equal
<=	Less than or equal
<>	Not equal. **Note:** In some versions of SQL this operator may be written as !=
BETWEEN	Between a certain range
LIKE	Search for a pattern
IN	To specify multiple possible values for a column
AND, OR, NOT	

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  DNO <> 1;
```

```
SSN    NAME
12     Ali Hasan Adel
14     Alaa Ali Mohamed
15     Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  DNO < 3 AND Ssn >11;
```

```
SSN     NAME
12      Ali Hasan Adel
13      Mohsen Mahmod
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  DNO < 3 OR Ssn >11;
```

```
SSN    NAME
11     Ahmed Mohamed
12     Ali Hasan Adel
13     Mohsen Mahmod
14     Alaa Ali Mohamed
15     Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- Basic form of the SELECT statement:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  DNO < 3 AND  NOT (Ssn >11);
```

```
SSN    NAME
11     Ahmed Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Logical comparison operators

=, <, <=, >, >=, <>, AND, OR, NOT

DNO < 3 AND Ssn >11

DNO < 3 OR Ssn >=11

DNO <= 3 AND NOT (Ssn >11)

Logical comparison operators

The SQL BETWEEN Operator

- The BETWEEN operator selects values within a given range. The values can be numbers, text, or dates.

- The BETWEEN operator is inclusive: **begin** and **end** values are included.

Ssn BETWEEN 2 AND 7

From 2 to 7

Basic Retrieval Queries in SQL

- The SQL BETWEEN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn BETWEEN 12 AND 14;
```

begin and end values are included

```
SSN    NAME
12     Ali Hasan Adel
13     Mohsen Mahmod
14     Alaa Ali Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Logical comparison operators

The SQL IN Operator

- The IN operator allows you to specify multiple values in a WHERE clause.

- The IN operator is a shorthand for multiple OR conditions.

```
Ssn  IN (12, 13,15 )

Ssn= 12    OR    Ssn=13    OR   Ssn=15

Name IN ( 'Ali Hasan Adel', 'Yaser Hesein Ali')
```

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn IN (12, 15 );
```

```
SSN     NAME
12      Ali Hasan Adel
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn IN ('12', '15' );
```

```
SSN     NAME
12      Ali Hasan Adel
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name IN ( 'Ali Hasan Adel',
                  'Yaser Hesein Ali');
```

```
SSN     NAME
12      Ali Hasan Adel
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name IN ( 'Ali  Hasan Adel',
                  'Yaser Hesein Ali');
```

```
SSN    NAME
15     Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name IN ( 'Ali Hasan Adel  ',
                    'Yaser Hesein Ali');
```

```
SSN    NAME
15     Yaser Hesein Ali
```

If the **NAME** is inserted as **'Ali Hasan Adel'** not as **'Ali Hasan Adel '**

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   DNO IN (
                SELECT  Dnumber
                FROM  DEPT
                WHERE   Dname='Research'
                );
```

```
SSN     NAME
11      Ahmed Mohamed
13      Mohsen Mahmod
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL IN Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   (Ssn, DNO) IN (
               SELECT  MgrSsn, Dnumber
               FROM  DEPT
               WHERE   Dname='Research'
               );
```

```
SSN    NAME
11     Ahmed Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber	MgeSsn
Research	1	11
Administration	2	15
Software	3	14

Logical comparison operators

The SQL IN, ANY, SOME, ALL Operators

Ssn IN (12, 13,15)

Ssn = ANY (12, 13,15)

Ssn = SOME (12, 13,15)

Logical comparison operators

The SQL IN, ANY, SOME, ALL Operators

Other operators that can be combined with ANY (or SOME): >, >=, <, <=, and <>

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

Ssn >= ANY (12, 13,15) 12, 13, 14, 15

Ssn <= SOME (12, 13,15) 11, 12, 13, 14, 15

Ssn <> SOME (12, 13,15) 11, 12, 13, 14, 15

Ssn <= ALL (12, 13,15) 11, 12

Basic Retrieval Queries in SQL

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn >= ANY (12, 15 );
```

```
SSN     NAME
12      Ali Hasan Adel
13      Mohsen Mahmod
14      Alaa Ali Mohamed
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn <> SOME (12, 15 );
```

Ssn	Name
11	Ahmed Mohamed
12	Ali Hasan Adel
13	Mohsen Mahmod
14	Alaa Ali Mohamed
15	Yaser Hesein Ali

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn <> ANY (12, 15 );
```

Ssn	Name
11	Ahmed Mohamed
12	Ali Hasan Adel
13	Mohsen Mahmod
14	Alaa Ali Mohamed
15	Yaser Hesein Ali

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn <> ALL (12, 15 );
```

```
SSN     NAME
11      Ahmed Mohamed
13      Mohsen Mahmod
14      Alaa Ali Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn > ALL (12, 13 );
```

```
SSN     NAME
14      Alaa Ali Mohamed
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Basic Retrieval Queries in SQL

- The SQL ANY Operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   DNO <= ANY (
                SELECT  Dnumber
                FROM  DEPT
                WHERE   Dname='Research'
                );
```

```
SSN     NAME
11      Ahmed Mohamed
13      Mohsen Mahmod
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Substring Pattern Matching

LIKE comparison operator

- Used **for string pattern matching**
- **%** replaces an arbitrary number of **zero** or **more** characters
- underscore **(_)** replaces a **single** charater

Substring Pattern Matching

LIKE comparison operator

- Used for string **pattern matching**
- % replaces an arbitrary number of zero or more characters
- underscore (_) replaces a single character

LIKE Operator	**Description**
Name LIKE 'a%'	Finds any values that start with "a"
Name LIKE '%a'	Finds any values that end with "a"
Name LIKE '%or%'	Finds any values that have "or" in any position
Name LIKE '_r%'	Finds any values that have "r" in the second position
Name LIKE 'a_%'	Finds any values that start with "a" and are at least 2 characters in length
Name LIKE 'a__%'	Finds any values that start with "a" and are at least 3 characters in length
Name LIKE 'a%o'	Finds any values that start with "a" and ends with "o"

Substring Pattern Matching

- **LIKE** comparison operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name LIKE  'A%' ;
```

```
SSN     NAME
11      Ahmed Mohamed
12      Ali Hasan Adel
14      Alaa Ali Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Substring Pattern Matching

- **LIKE** comparison operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name LIKE  'A %' ;
```

```
SSN    NAME
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Substring Pattern Matching

- **LIKE** comparison operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name LIKE  'a%' ;
```

SSN	NAME

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Substring Pattern Matching

- **LIKE** comparison operator

```
SELECT  Ssn , Name
FROM  EMP
WHERE   Name LIKE  'A__ %' ;
```

```
SSN    NAME
12     Ali Hasan Adel
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Substring Pattern Matching

LIKE comparison operator

- Used for string **pattern matching**
- **%** replaces an arbitrary number of zero or more characters
- underscore (_) replaces a single character

LIKE Operator
Name LIKE 'a%'
Name LIKE '%a'
Name LIKE '%or%'
Name LIKE '_r%'
Name LIKE 'a_%'
Name LIKE 'a__%'
Name LIKE 'a%o'

Arithmetic Operators

- Standard arithmetic operators:
  - Addition (+), subtraction (–), multiplication (*), and division (/)

Ssn - DNO = 10

DNO + 2 = 5

DNO / 2 >= 5

Ssn * DNO > 14

Arithmetic Operators

- Standard arithmetic operators:

```
SELECT  Ssn , Name
FROM  EMP
WHERE   DNO + 1 = 3;
```

```
SSN     NAME
12      Ali Hasan Adel
15      Yaser Hesein Ali
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Arithmetic Operators

- Standard arithmetic operators:

```
SELECT  Ssn , Name
FROM  EMP
WHERE   DNO + Ssn = 14;
```

```
SSN     NAME
12      Ali Hasan Adel
13      Mohsen Mahmod
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Arithmetic Operators

- Standard arithmetic operators:

```
SELECT  Ssn , Name
FROM  EMP
WHERE  Ssn  / DNO = 6;
```

```
SSN    NAME
12     Ali Hasan Adel
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Query from Two or More Relations

- Retrieve the name of all employees who works for the ‘research’ department

```
SELECT  Name
FROM  EMP, DEPT
WHERE  EMP.DNO=DEPT.Dnumber AND
            Dname='Research' ;
```

EMP

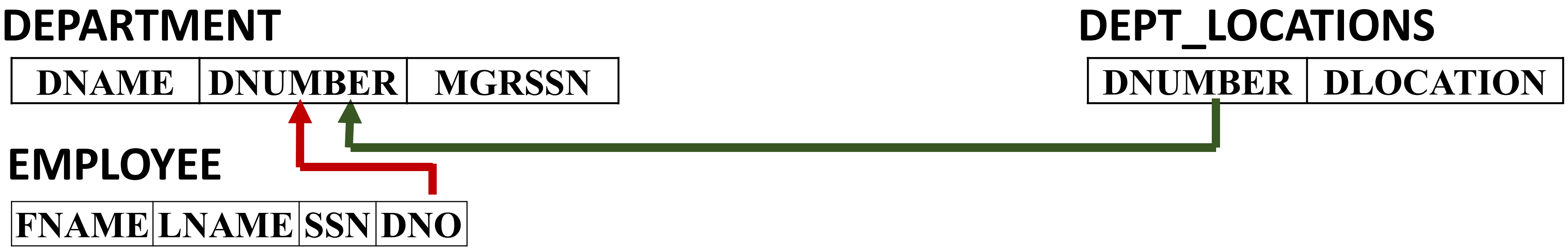
Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

NAME
Ahmed Mohamed
Mohsen Mahmod

Query from Two or More Relations



DEPARTMENT

DNAME	DNUMBER	MGRSSN

DEPT_LOCATIONS

DNUMBER	DLOCATION

EMPLOYEE

FNAME	LNAME	SSN	DNO

- Retrieve the firs name of all employees who works for all departments that have ‘Stafford’ location

```
SELECT  FNAME
FROM    EMPLOYEE,    DEPARTMENT,    DEPT_LOCATIONS
WHERE   EMPLOYEE.DNO  =   DEPARTMENT. DNUMBER AND
        DEPARTMENT. DNUMBER  =  DEPT_LOCATIONS . DNUMBER
        AND  DLOCATION ='Stafford' ;
```

FNAME
Ahmad
Alicia
Jennifer

The SQL SELECT DISTINCT Statement

- The SELECT DISTINCT statement is used to return only distinct (different) values.

- Inside a table, a column often contains many duplicate values; and sometimes you only want to list the different (distinct) values.

The SQL SELECT DISTINCT Statement

EMPLOYEE

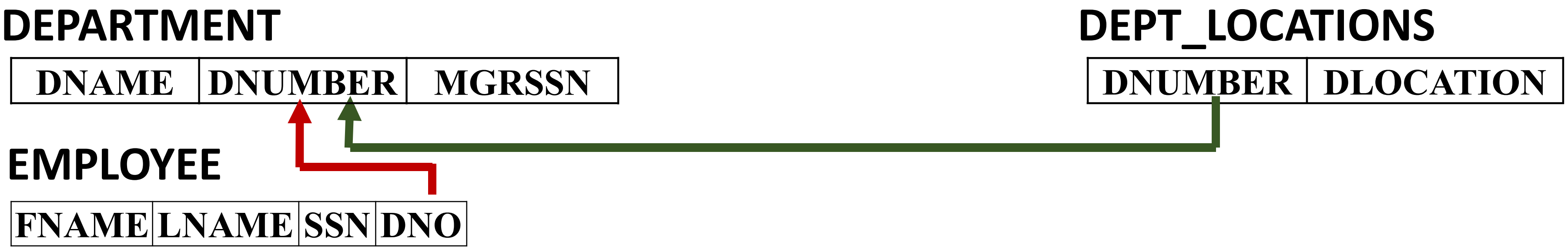
FNAME	LNAME	SSN	DNO

- Retrieve the Dno of all employees

DNO
1
6
4
5
8
7

```
SELECT DISTINCT  DNO
FROM     EMPLOYEE;
```

The SQL SELECT DISTINCT Statement



DEPARTMENT

DNAME	DNUMBER	MGRSSN

DEPT_LOCATIONS

DNUMBER	DLOCATION

EMPLOYEE

FNAME	LNAME	SSN	DNO

- Retrieve the firs name and SSN of all employees who works for all departments that have location with at least 3 characters in length

```
SELECT  FNAME, SSN
FROM    EMPLOYEE,   DEPARTMENT,   DEPT_LOCATIONS
WHERE   EMPLOYEE.DNO  =  DEPARTMENT. DNUMBER AND
        DEPARTMENT. DNUMBER  =  DEPT_LOCATIONS . DNUMBER
        AND  DLOCATION LIKE ‘___%' ;
```

EMPLOYEE

FNAME	SSN	DNO
Kim	333333300	6
Bob	666666600	8

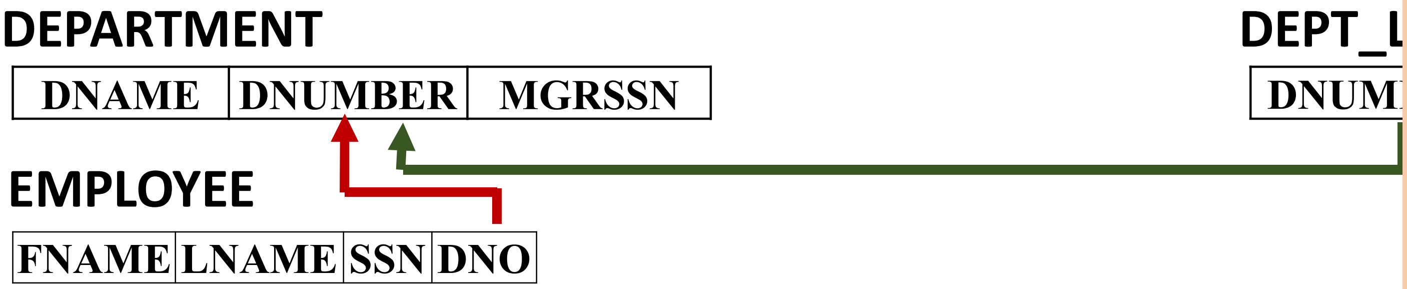
DEPARTMENT

DNAME	DNUMBER	MGRSSN
Software	6	111111100
Sales	8	555555500

DEPT_LOCATIONS

DNUMBER	DLOCATION
6	Atlanta
6	Sacramento
8	Chicago
8	Dallas
8	Miami
8	Philadephia
8	Seattle

The SQL SELECT DISTINCT Statement

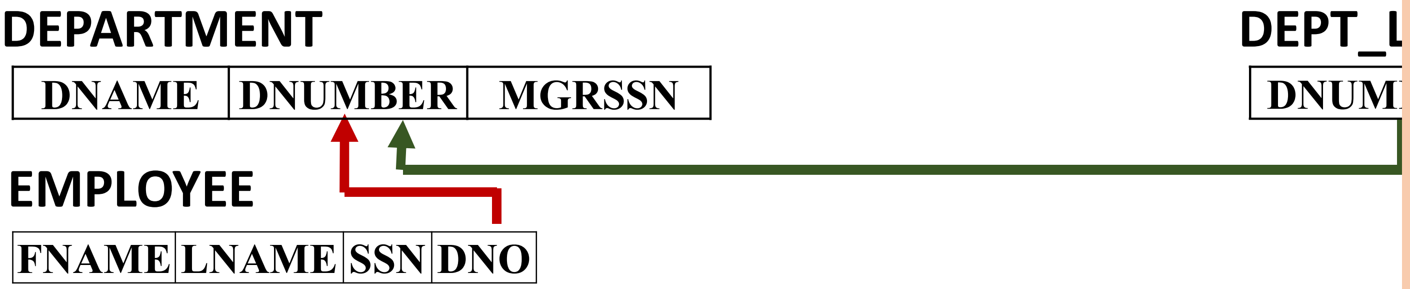


- Retrieve the firs name and SSN of all employees who w departments that have location with at least 3 characters

```
SELECT  FNAME, SSN
FROM    EMPLOYEE,   DEPARTMENT,   DEPT_LOC
WHERE   EMPLOYEE.DNO  =  DEPARTMENT. DNUM
        DEPARTMENT. DNUMBER  =  DEPT_LOCAT                ER
        AND  DLOCATION LIKE '___%' ;
```

FNAME	SSN
Bob	666666600
Bob	666666600
Bob	666666600
Bob	666666600
Bob	666666600
Kim	333333300
Kim	333333300
Jared	111111100
Jared	111111100
John	555555500
John	555555500
Brad	111111103
Brad	111111103
Justin	111111102
Justin	111111102
Jon	111111101
Jon	111111101
Jeff	333333301
Jeff	333333301

The SQL SELECT DISTINCT Statement



- Retrieve the firs name and SSN of all employees who w departments that have location with at least 3 characters

```
SELECT DISTINCT  FNAME, SSN
FROM    EMPLOYEE,   DEPARTMENT,   DEPT_LOC
WHERE   EMPLOYEE.DNO  =   DEPARTMENT. DNUM
        DEPARTMENT. DNUMBER  =  DEPT_LOCAT                ER
        AND  DLOCATION LIKE '___%' ;
```

FNAME	SSN
Kim	333333300
Justin	111111102
Helga	666666609
Carl	666666611
Ray	666666606
Brad	111111103
Jon	111111101
Naveen	666666610
Bob	666666600
John	555555500
Jill	666666601
Jon	666666605
Gerald	666666607
Arnold	666666608
Kate	666666602
Lyle	666666603
Sammy	666666612
Red	666666613
Jeff	333333301
Billie	666666604

Aliasing, Renaming, and Tuple Variables

- **Aliases** or **tuple variables**
  - Declare alternative relation names E and S

```
SELECT  Name
FROM    EMP, DEPT
WHERE   EMP.DNO=DEPT.Dnumber AND
         Name ='Research' ;
```

Error:
ORA-00918: column ambiguously defined

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Name	Dnumber
Research	1
Administration	2
Software	3

Aliasing, Renaming, and Tuple Variables

- **Aliases** or **tuple variables**
  - Declare alternative relation names E and S

```
SELECT  EMP.Name AS Ename
FROM    EMP, DEPT
WHERE   EMP.DNO=DEPT.Dnumber AND
        DEPT.Name='Research' ;
```

ENAME
Ahmed Mohamed
Mohsen Mahmod

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Name	Dnumber
Research	1
Administration	2
Software	3

Aliasing, Renaming, and Tuple Variables

- **Aliases** or **tuple variables**
  - Declare alternative relation names E and S

```
SELECT  E.Name AS Ename
FROM    EMP  E, DEPT  D
WHERE   E.DNO=D.Dnumber AND
        D.Name='Research' ;
```

EMP

Ssn	**Name**	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Name	**Dnumber**
Research	1
Administration	2
Software	3

```
ENAME
Ahmed Mohamed
Mohsen Mahmod
```

Tables as Sets in SQL (cont'd.)

- Set operations
    - **UNION**, **`EXCEPT or MINUS`** (difference), **INTERSECT**
    - Corresponding multiset operations: **UNION ALL**, **EXCEPT ALL**, **INTERSECT ALL**)

Tables as Sets in SQL (cont'd.)

The SQL UNION Operator

- The UNION operator is used to combine the result-set of two or more SELECT statements.
- Every SELECT statement within UNION must have the same number of columns
- The columns must also have similar data types
- The columns in every SELECT statement must also be in the same order

Tables as Sets in SQL (cont'd.)

The SQL **UNION** Operator

UNION Syntax

```
SELECT column_name(s) FROM table1
UNION
SELECT column_name(s) FROM table2;
```

UNION ALL Syntax

- The UNION operator selects only distinct values by default. To allow duplicate values, use UNION ALL:

```
SELECT column_name(s) FROM table1
UNION ALL
SELECT column_name(s) FROM table2;
```

Tables as Sets in SQL (cont'd.)

```
SELECT  Name
FROM    EMP
WHERE   DNO = 2
UNION
SELECT Name
FROM    EMP
WHERE   Name LIKE  'A%';
```

Ahmed Mohamed

Alaa Ali Mohamed

Ali Hasan Adel

Yaser Hesein Ali

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

Ali Hasan Adel

Yaser Hesein Ali

The UNION operator selects only distinct values by default.

Ahmed Mohamed

Ali Hasan Adel

Alaa Ali Mohamed

Tables as Sets in SQL (cont’d.)

```
SELECT  Name
FROM    EMP
WHERE   DNO = 2
UNION ALL
SELECT Name
FROM    EMP
WHERE   Name LIKE  'A%';
```

NAME

Ali Hasan Adel

Yaser Hesein Ali

Ahmed Mohamed

Ali Hasan Adel

Alaa Ali Mohamed

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

Ali Hasan Adel

Yaser Hesein Ali

Ahmed Mohamed

Ali Hasan Adel

Alaa Ali Mohamed

Tables as Sets in SQL (cont’d.)

```
SELECT  Name, DNO
FROM    EMP
WHERE   Name LIKE  'A%'
 UNION
SELECT  Dname, Dnumber
FROM    DEPT
WHERE   Dname LIKE  'A%';
```

```
NAME                 DNO
Ahmed Mohamed        1
Ali Hasan Adel       2
Alaa Ali Mohamed     3
Administration       2
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

DEPT

Dname	Dnumber
Research	1
Administration	2
Software	3

Tables as Sets in SQL (cont’d.)

The SQL INTERSECT Operator

```
SELECT  Name
FROM    EMP
WHERE   DNO = 2
INTERSECT
SELECT Name
FROM    EMP
WHERE   Name LIKE  'A%';
```

NAME
Ali Hasan Adel

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

Ali Hasan Adel

Yaser Hesein Ali

Ahmed Mohamed

Ali Hasan Adel

Alaa Ali Mohamed

Tables as Sets in SQL (cont'd.)

The SQL EXCEPT or MINUS (difference)

```
SELECT  Name
FROM    EMP
WHERE   DNO = 2
MINUS
SELECT Name
FROM    EMP
WHERE   Name LIKE  'A%';
```

NAME

Yaser Hesein Ali

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

Ali Hasan Adel

Yaser Hesein Ali

Ahmed Mohamed

Ali Hasan Adel

Alaa Ali Mohamed

Ordering of Query Results

Use **ORDER BY** clause

- Keyword **DESC** to see result in a descending order of values
- Keyword **ASC** to specify ascending order explicitly

ORDER BY D.Dname **DESC**, E.Lname **ASC**, E.Fname **ASC**

Ordering of Query Results

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

```
SELECT  Name, DNO
FROM    EMP
WHERE   Ssn <= 15
ORDER BY DNO;
```

```
NAME                 DNO
Ahmed Mohamed        1
Mohsen Mahmod        1
Yaser Hesein Ali     2
Ali Hasan Adel       2
Alaa Ali Mohamed     3
```

Ascending Order

Ordering of Query Results

```
SELECT  Name, DNO
FROM     EMP
WHERE  Ssn <= 15
ORDER BY DNO ASC;
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

```
NAME                DNO
Ahmed Mohamed       1
Mohsen Mahmod       1
Yaser Hesein Ali    2
Ali Hasan Adel      2
Alaa Ali Mohamed    3
```

Ascending Order

Ordering of Query Results

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

```
SELECT  Name, DNO
FROM    EMP
WHERE   Ssn <= 15
ORDER BY DNO DESC;
```

```
NAME               DNO
Alaa Ali Mohamed   3
Ali Hasan Adel     2
Yaser Hesein Ali   2
Ahmed Mohamed      1
Mohsen Mahmod      1
```

Descending Order

Ordering of Query Results

```
SELECT  Name, DNO
FROM     EMP
WHERE  Ssn <= 15
ORDER BY DNO ASC, Name DESC;;
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

```
NAME                DNO
Mohsen Mahmod       1
Ahmed Mohamed       1
Yaser Hesein Ali    2
Ali Hasan Adel      2
Alaa Ali Mohamed    3
```

INSERT, DELETE, and UPDATE Statements in SQL

- Three commands used to modify the database:
  - **INSERT**, **DELETE**, and **UPDATE**

The INSERT Command

- Specify the relation name and a list of values for the tuple

```
INSERT INTO EMP
VALUES   (16, 'Mohamed', 3 );
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	**3**

The INSERT Command

- Specify the relation name and a list of values for the tuple

```
INSERT INTO EMP
VALUES   ('16', 'Mohamed', '3' );
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	**3**

```
CREATE  TABLE  EMP (
      Ssn         INT,
      Name     VARCHAR(30) ,
      Dno        INT
);
```

The INSERT Command

- Specify the relation name and a list of values for the tuple

```
INSERT INTO EMP
VALUES   ('16', Mohamed, '3' );
```

ERROR

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

```
CREATE  TABLE  EMP (
      Ssn        INT,
      Name    VARCHAR(30) ,
      Dno       INT
);
```

SQL NULL Values

- A field with a NULL value is a field with no value.
- If a field in a table is optional, it is possible to insert a new record or update a record without adding a value to this field. Then, the field will be saved with a NULL value.
- Note: A NULL value is different from a zero value or a field that contains spaces. A field with a NULL value is one that has been left blank during record creation!

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	

The INSERT Command

- Specify the relation name and a list of values for the tuple

```
INSERT INTO EMP(Ssn, Name)
VALUES   (16, 'Mohamed' );
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	

The INSERT Command

- Specify the relation name and a list of values for the tuple

```
INSERT INTO EMP(Ssn, DNO)
VALUES   (16, 3);
```

EMP

Ssn	**Name**	**DNO**
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16		**3**

How to Test for NULL Values?

- It is not possible to test for NULL values with comparison operators, such as =, <, or <>.
- We will have to use the **IS NULL** and **IS NOT NULL** operators instead.

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	

How to Test for NULL Values?

- It is not possible to test for NULL values with comparison operators, such as =, <, or <>.
- We will have to use the **IS NULL** and **IS NOT NULL** operators instead.

```
SELECT  Name
FROM    EMP
WHERE   DNO IS NULL;
```

```
NAME
Mohamed
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	

How to Test for NULL Values?

- It is not possible to test for NULL values with comparison operators, such as =, <, or <>.
- We will have to use the **IS NULL** and **IS NOT NULL** operators instead.

```
SELECT  Name
FROM    EMP
WHERE   DNO IS NOT NULL;
```

EMP

Ssn	**Name**	**DNO**
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	

Name
Ahmed Mohamed
Ali Hasan Adel
Mohsen Mahmod
Alaa Ali Mohamed
Yaser Hesein Ali

The **DELETE** Command

- Removes tuples from a relation
  - Includes a WHERE clause to select the tuples to be deleted

```
DELETE FROM EMP
WHERE  DNO IS NULL;
```

EMP

Ssn	**Name**	**DNO**
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2
16	**Mohamed**	

The **DELETE** Command

- Removes tuples from a relation
  - Includes a WHERE clause to select the tuples to be deleted

```
DELETE FROM EMP
WHERE  DNO =3;
```

EMP

Ssn	**Name**	**DNO**
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2

The DELETE Command

- Removes tuples from a relation
  - Includes a WHERE clause to select the tuples to be deleted

```
DELETE FROM EMP;
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
15	Yaser Hesein Ali	2

The **UPDATE** Command

- Modify attribute values of one or more selected tuples
- Additional **SET** clause in the **UPDATE** command
  - Specifies attributes to be modified and new values

The UPDATE Command

```
UPDATE EMP
SET Ssn=18 , Name='Mohamed'
WHERE  DNO =3;
```

EMP

Ssn	Name	DNO
11	Ahmed Mohamed	1
12	Ali Hasan Adel	2
13	Mohsen Mahmod	1
14	Alaa Ali Mohamed	3
15	Yaser Hesein Ali	2