Lecture 27

THE NORTHCAP UNIVERSITY

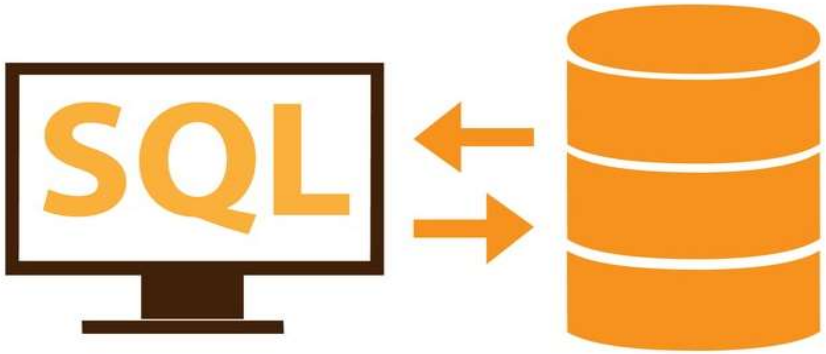
Basic SQL

30 YEARS OF EXCELLENCE

NAAC Accredited

Basic SQL

- **Structured Query Language**
- Statements for data definitions, queries, and updates (both DDL and DML)
- Each statement in SQL ends with a semicolon



What can be done using SQL

- execute queries on a database
- retrieve data from a database
- insert records in a database
- update records in a database
- delete records from a database
- create new databases
- create new tables in a database
- create stored procedures in a database
- create views in a database
- set permissions on tables, procedures, and views

Relational Algebra vs SQL

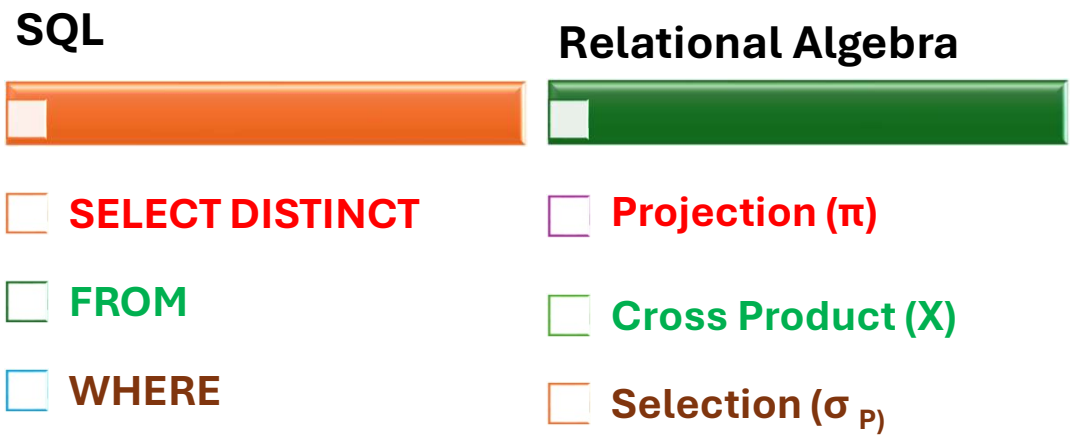
SELECT DISTINCT A1, A2, ... An

FROM R1, R2, ... Rn

WHERE P

==========

$$\pi_{A1, A2, \ldots\ldots An} (\sigma_{P} (R1 \times R2 \times \ldots\ldots\ldots \times Rn)$$



COMPANY Database

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

DEPT_LOCATIONS

Dnumber	Dlocation

PROJECT

Pname	Pnumber	Plocation	Dnum

WORKS_ON

Essn	Pno	Hours

DEPENDENT

Essn	Dependent_name	Sex	Bdate	Relationship

CREATE TABLE Command

- To create a new table in a database.
- Specify each attributes and their data types (eg. INTEGER, FLOAT, DECIMAL(i,j), CHAR(n), VARCHAR(n))
- **Syntax:**

CREATE TABLE ***table_name*** **(**

column1 datatype**,**

column2 datatype**,**

column3 datatype**,**

....);

CREATE TABLE Command

- A constraint NOT NULL may be specified on an attribute
- Can specify the primary key attributes, secondary keys and referential integrity constraints (foreign keys).

```
CREATE TABLE DEPT
  (  DNAME  VARCHAR(10) NOT NULL,
     DNUMBER          INTEGER NOT NULL,
     MGRSSN CHAR(9),
     MGRSTARTDATE CHAR(9),
     PRIMARY KEY (DNUMBER),
     UNIQUE (DNAME),
     FOREIGN KEY (MGRSSN) REFERENCES EMP );
```

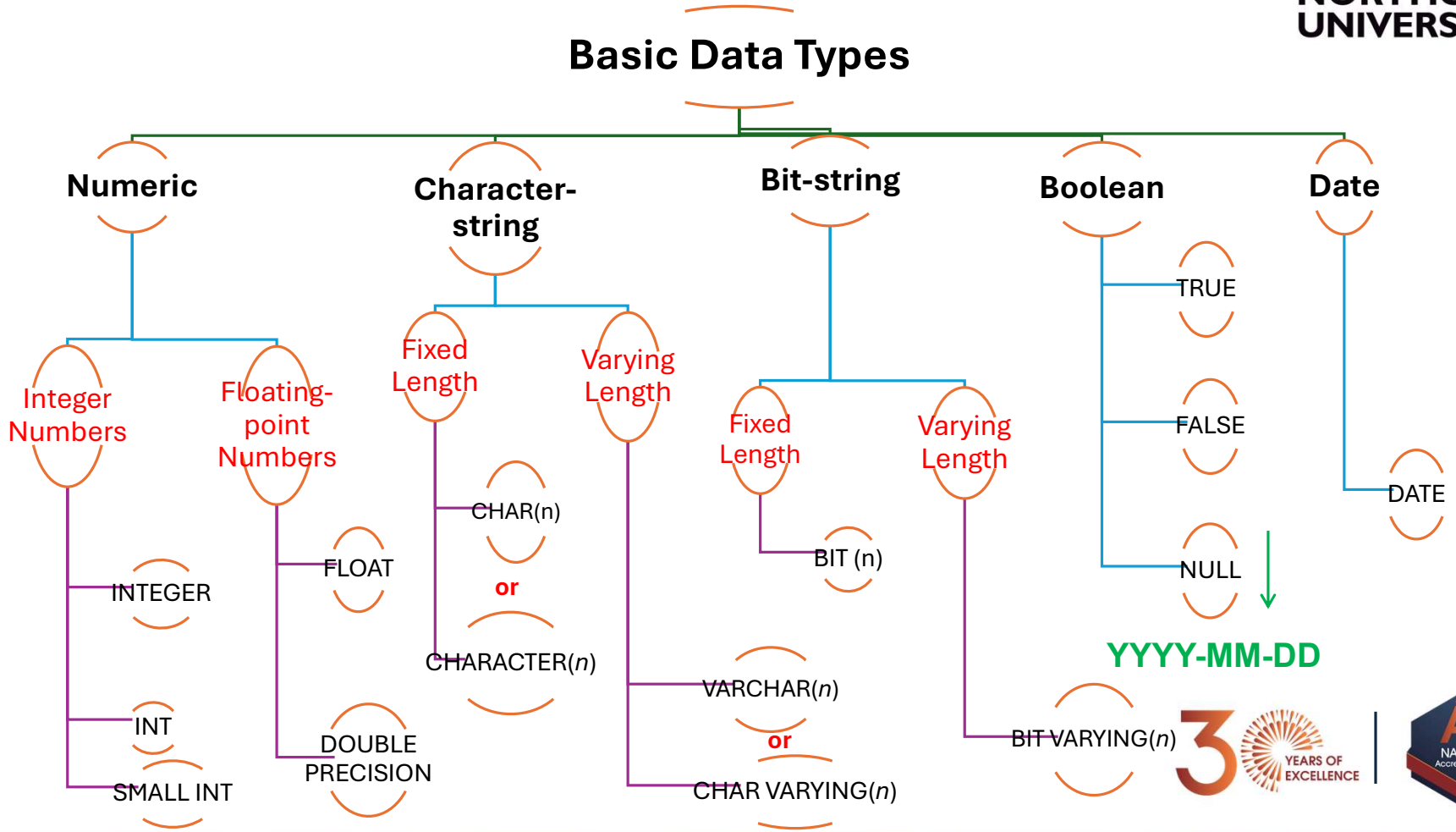
CREATE TABLE Command Example

```
CREATE TABLE EMPLOYEE
        ( Fname             VARCHAR(15)         NOT NULL,
          Minit             CHAR,
          Lname             VARCHAR(15)         NOT NULL,
          Ssn               CHAR(9)             NOT NULL,
          Bdate             DATE,
          Address           VARCHAR(30),
          Sex               CHAR,
          Salary            DECIMAL(10,2),
          Super_ssn         CHAR(9),
          Dno               INT                 NOT NULL,
        PRIMARY KEY (Ssn),
        FOREIGN KEY (Super_ssn) REFERENCES EMPLOYEE(Ssn),
        FOREIGN KEY (Dno) REFERENCES DEPARTMENT(Dnumber) );
CREATE TABLE DEPARTMENT
        ( Dname             VARCHAR(15)         NOT NULL,
          Dnumber           INT                 NOT NULL,
          Mgr_ssn           CHAR(9)             NOT NULL,
          Mgr_start_date    DATE,
        PRIMARY KEY (Dnumber),
        UNIQUE (Dname),
        FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn) );
```

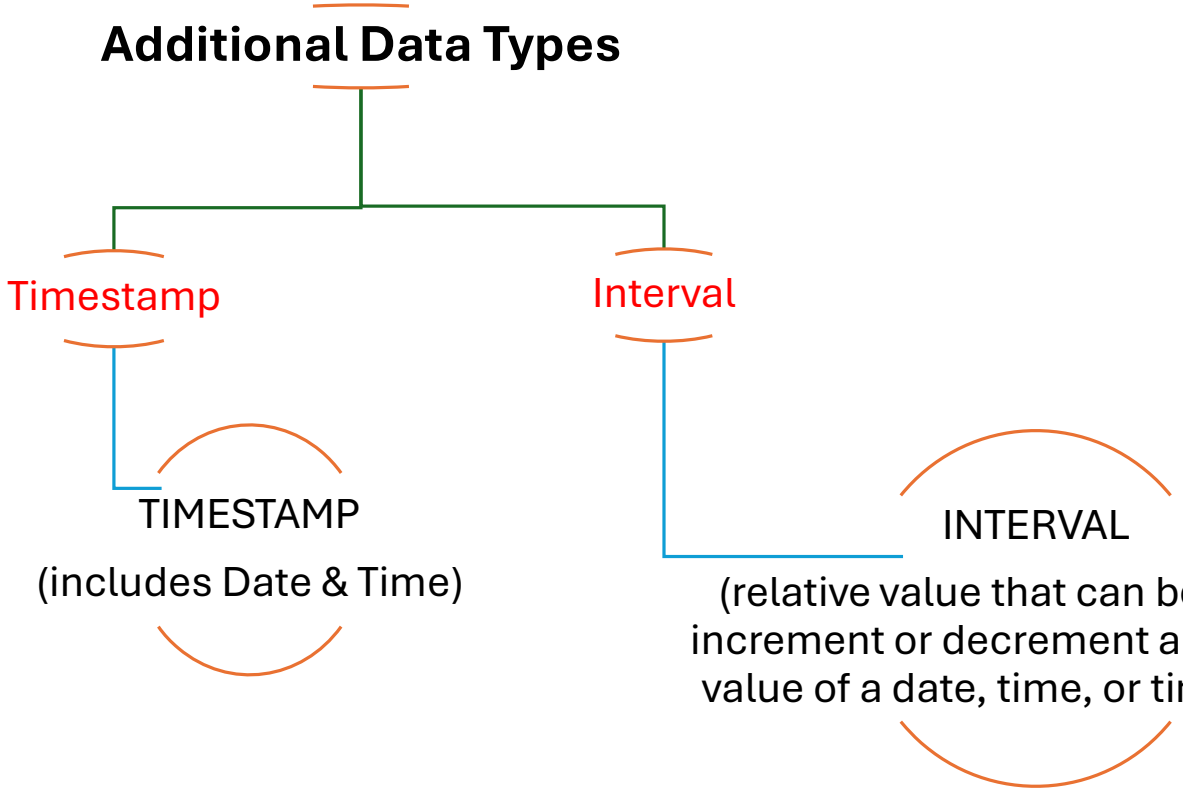
CREATE TABLE Command Example

```
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),
          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,
          Bdate             DATE,
          Relationship      VARCHAR(8),
        PRIMARY KEY (Essn, Dependent_name),
        FOREIGN KEY (Essn) REFERENCES EMPLOYEE(Ssn) );
```

Attribute Data Types and Domains



Attribute Data Types and Domains



Attribute Data Types and Domains

THE NORTHCAP UNIVERSITY

- **Domain**
  - Name used with the attribute specification
  - Makes it easier to change the data type for a domain that is used by numerous attributes
  - Improves schema readability
  - **Example:**

CREATE DOMAIN SSN_TYPE AS CHAR(9);

use SSN_TYPE in place of CHAR(9)

30 YEARS OF EXCELLENCE | NAAC Accredited

Specifying Attribute Constraints & Attribute Defaults

```
CREATE TABLE DEPT
  (  DNAME VARCHAR(10) NOT NULL,
     MGRSSN CHAR(9) DEFAULT 'N.A.',
     MGRSTARTDATE CHAR(9),
     UNIQUE (DNAME),
     DNUMBER INT NOT NULL CHECK(DNUMBER > 0
          AND DNUMBER < 21);
     PRIMARY KEY (DNUMBER),
     FOREIGN KEY (MGRSSN) REFERENCES EMP );
```

NULL is not permitted for DNAME

Set Default value for MGRSSN

Limit the value range of DNUMBER

Specifying Constraints on Tuples Using CHECK

```
CREATE TABLE DEPT
  (  DNAME VARCHAR(10) NOT NULL,
     MGRSSN CHAR(9) DEFAULT 'N.A.',
     MGRSTARTDATE DATE,
     UNIQUE (DNAME),
     DNUMBER INT NOT NULL,
     D_CREATE_DATE DATE,
     PRIMARY KEY (DNUMBER),
     FOREIGN KEY (MGRSSN) REFERENCES EMP,
     CHECK (D_CREATE_DATE <= MGRSTARTDATE);
);
```

- *Specified at the end of CREATE TABLE command*
- *Apply to each tuple individually*

Specifying Key and Referential Integrity Constraints

```
CREATE TABLE DEPARTMENT
    ( Dname VARCHAR(15) NOT NULL,
    Dnumber INT NOT NULL,
    Mgr_ssn CHAR(9) NOT NULL,
    Mgr_start_date DATE,
    PRIMARY KEY (Dnumber),
    UNIQUE (Dname),
    FOREIGN KEY (Mgr_ssn) REFERENCES EMPLOYEE(Ssn) );
```

To specify Dnumber is a Primary Key

To specify Dname is an alternate/secondary key

Specifying Key and Referential Integrity Constraints (Cont.)

THE NORTHCAP UNIVERSITY

```
CREATE TABLE DEPT_LOCATIONS
    ( Dnumber INT NOT NULL,
    Dlocation VARCHAR(15) NOT NULL,
    PRIMARY KEY (Dnumber, Dlocation),
    FOREIGN KEY (Dnumber) REFERENCES DEPARTMENT (Dnumber) ON DELETE CASCADE
    ON UPDATE CASCADE);
```

It implies Dnumber of DEPT_LOCATIONS table is referencing Dnumber of DEPARTMENT table

If a value is deleted/updated in Dnumber of DEPARTMENT table, what action is to be taken for Dnumber of DEPT_LOCATIONS table.

Options include SET NULL, CASCADE, and SET DEFAULT

ALTER Command

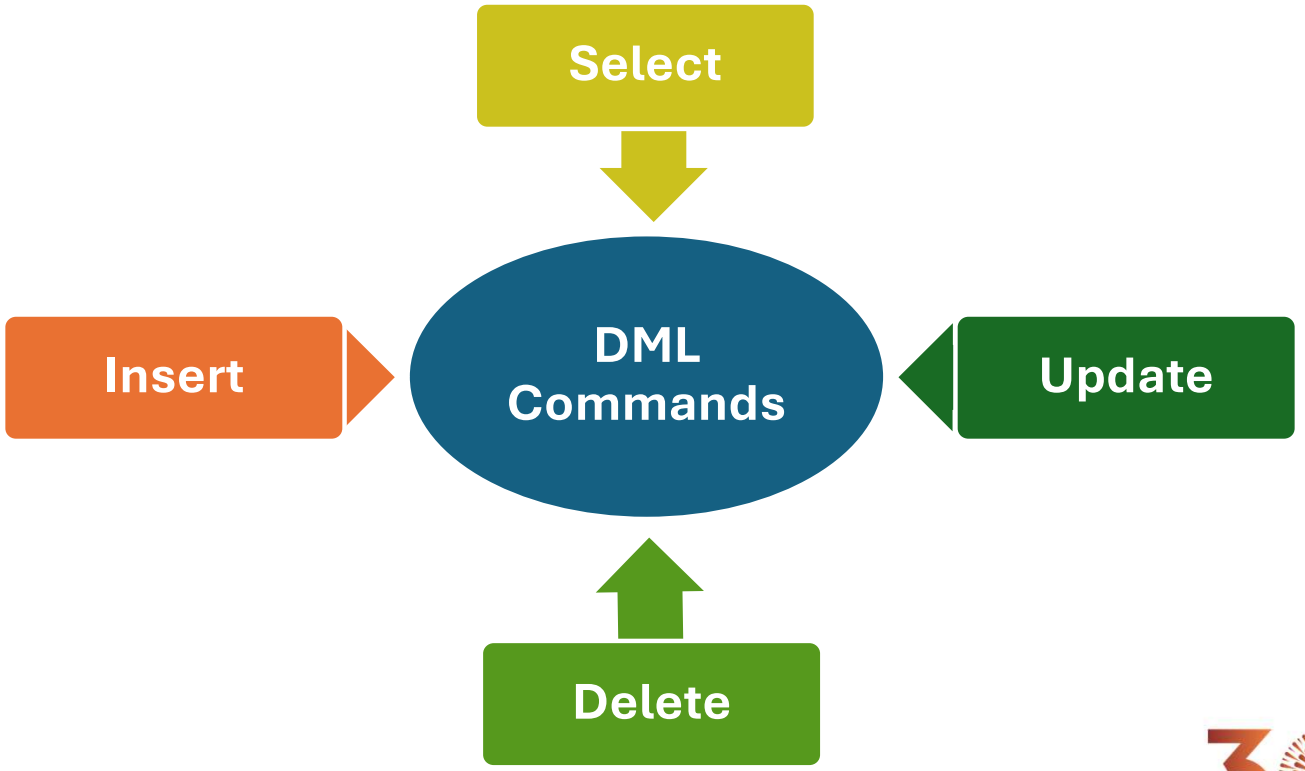
- **Alter table actions** include:
  - Adding or dropping a column (attribute)
  - Changing a column definition
  - Adding or dropping table constraints
- **Example:**

```
ALTER TABLE COMPANY.EMPLOYEE ADD COLUMN Job VARCHAR(12);
```

Example - ALTER Command

- **ALTER TABLE** COMPANY.EMPLOYEE **DROP COLUMN** Address **CASCADE**;
- **ALTER TABLE** COMPANY.DEPARTMENT **ALTER COLUMN** Mgr_ssn **DROP DEFAULT**;
- **ALTER TABLE** COMPANY.DEPARTMENT **ALTER COLUMN** Mgr_ssn **SET DEFAULT** '333445555'
- **ALTER TABLE** COMPANY.EMPLOYEE **DROP CONSTRAINT** EMPSUPERFK **CASCADE**

DML Commands



The INSERT Command

- **CREATE TABLE** EMPLOYEE

  (Fname VARCHAR(15) **NOT NULL**,
  Minit CHAR,
  Lname VARCHAR(15) **NOT NULL**,
  Ssn CHAR(9) **NOT NULL**,
  Bdate DATE,
  Address VARCHAR(30),
  Sex CHAR,
  Salary DECIMAL(10,2),
  Super_ssn CHAR(9),
  Dno INT **NOT NULL**,
  PRIMARY KEY (Ssn),
  FOREIGN KEY (Super_ssn) **REFERENCES** EMPLOYEE(Ssn),
  FOREIGN KEY (Dno) **REFERENCES** DEPARTMENT(Dnumber));

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

Creating new table in database

Inserting entries in EMPLOYEE table

INSERT Command (Cont.)

- **INSERT INTO** EMPLOYEE (Fname, Lname, Dno, Ssn)

  VALUES (‘Richard’, ‘Marini’, 4, ‘653298653’);

values must include all attributes with NOT NULL specification and no default value.

- **INSERT INTO** EMPLOYEE (Fname, Lname, Ssn, Dno)

  VALUES (‘Robert’, ‘Hatcher’, ‘980760540’, 2);

rejected as EMPLOYEE Dno is foreign key of DEPARTMENT table and there is no department 2

- **INSERT INTO** EMPLOYEE (Fname, Lname, Dno)

  VALUES (‘Robert’, ‘Hatcher’, 5);

rejected as Ssn is defined NOT NULL

Basic Retrieval Queries - Select

Dname	Dnumber	Dlocation
Headquarters	1	Houston
Administration	4	Stafford
Research	5	Bellaire
Research	5	Sugarland
Administration	4	Stafford
Research	5	Houston

Two or more tuples that are identical in all their attribute values are allowed in SQL

SELECT-FROM-WHERE Structure

SELECT <attribute list>

FROM <table list>

WHERE <condition>;

- <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.

Example

- **Query:** Retrieve the birth date 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’;

- **Output:**

Bdate	Address
1965-01-09	731Fondren, Houston, TX

Example

- **Query:** 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;

- **Output:**

Fname	Lname	Address
John	Smith	731 Fondren, Houston, TX
Franklin	Wong	638 Voss, Houston, TX
Ramesh	Narayan	975 Fire Oak, Humble, TX
Joyce	English	5631 Rice, Houston, TX

Example

- **Query:** For every project located in ‘Stafford’, list the project number, the controlling department number, and the department manager’s last name, address, and birth date.

  SELECT Pnumber, Dnum, Lname, Address, Bdate

  FROM PROJECT, DEPARTMENT, EMPLOYEE

  WHERE Dnum = Dnumber **AND** Mgr_ssn = Ssn **AND** Plocation=‘Stafford’;

- **Output:**

Pnumber	Dnum	Lname	Address	Bdate
10	4	Wallace	291Berry, Bellaire, TX	1941-06-20
30	4	Wallace	291Berry, Bellaire, TX	1941-06-20

Ambiguous Attribute Names

- Same name can be used for two (or more) attributes as long as the attributes are in different relations

SELECT Fname, EMPLOYEE.Name, Address

FROM EMPLOYEE, DEPARTMENT

WHERE DEPARTMENT.Name = 'Research' **AND**

DEPARTMENT.Dnumber = EMPLOYEE.Dnumber;

Specify table name along with attribute name whenever two tables have same attribute names

Aliasing, Renaming and Tuple Variables

- Query: 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.Super_ssn = S.Ssn;
```

Aliasing table names.

- **Output:**

E.Fname	E.Lname	S.Fname	S.Lname
John	Smith	Franklin	Wong
Franklin	Wong	James	Borg
Alicia	Zelaya	Jennifer	Wallace
Jennifer	Wallace	James	Borg
Ramesh	Narayan	Franklin	Wong
Joyce	English	Franklin	Wong
Ahmad	Jabbar	Jennifer	Wallace

Table & Attribute Aliasing:
EMPLOYEE AS E (Fn, Mi, Ln, Ssn, Bd, Addr, Sex, Sal, Sssn, Dno)

Unspecified WHERE Clause

- **Query:** Select all EMPLOYEE Ssns

  SELECT Ssn

  FROM EMPLOYEE;

no condition on tuple selection

- **Output:**

Ssn
123456789
333445555
999887777
453453453
666884444
987654321
987987987
888665555

Unspecified WHERE Clause (Cont.)

- **Query:** Select all combinations of EMPLOYEE Ssn and DEPARTMENT Dname in the database.

```
SELECT Ssn, Dname
FROM EMPLOYEE, DEPARTMENT;
```

CROSS PRODUCT - All possible tuple combinations

- **Output:**

Ssn	Dname
123456789	Research
333445555	Research
999887777	Research
987654321	Research
666884444	Research
453453453	Research
987987987	Research
888665555	Research
123456789	Administration
333445555	Administration
999887777	Administration
987654321	Administration
666884444	Administration
453453453	Administration
987987987	Administration
888665555	Administration
123456789	Headquarters
333445555	Headquarters
999887777	Headquarters
987654321	Headquarters
666884444	Headquarters
453453453	Headquarters
987987987	Headquarters
888665555	Headquarters

Use of the Asterisk

- **Query:** Retrieve all the attribute values of any EMPLOYEE who works in DEPARTMENT number 5

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

Retrieve all the attribute values of the selected tuples

- **Output:**

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno
John	B	Smith	123456789	1965-09-01	731 Fondren, Houston, TX	M	30000	333445555	5
Franklin	T	Wong	333445555	1955-12-08	638 Voss, Houston, TX	M	40000	888665555	5
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

What is output?

- **Query 1:**

  SELECT *

  FROM EMPLOYEE, DEPARTMENT

  WHERE Dname = ‘Research’ **AND** Dno = Dnumber;

- **Query 2:**

  SELECT *

  FROM EMPLOYEE, DEPARTMENT;

Solution

- **Query 1 Output:**

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ ssn	Dno	Dname	Dnumber	Mgr_ ssn	Mgr_ start_date
…	…	…	…	…	…	…	…	…	5	Research	5	…	…
…	…	…	…	…	…	…	…	…	5	Research	5	…	…
…	…	…	…	…	…	…	…	…	5	Research	5	…	…
…	…	…	…	…	…	…	…	…	5	Research	5	…	…
…	…	…	…	…	…	…	…	…	5	Research	5	…	…

- **Query 2 Output:**

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ ssn	Dno	Dname	Dnumber	Mgr_ ssn	Mgr_ start_date
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…
…	…	…	…	…	…	…	…	…	…	…	…	…	…

Tables as Sets

THE NORTHCAP UNIVERSITY

- **Query:** Retrieve the salary of every employee.

  SELECT ALL Salary

  FROM EMPLOYEE;

- **Output:**

Salary
30000
40000
25000
43000
38000
25000
25000
55000

SQL does not automatically eliminate duplicate tuples in query results

30 YEARS OF EXCELLENCE

NAAC Accredited

Tables as Sets (Cont.)

- **Query:** Retrieve all distinct salary values.

```
SELECT DISTINCT Salary
FROM EMPLOYEE;
```

Allows only distinct tuples in the result

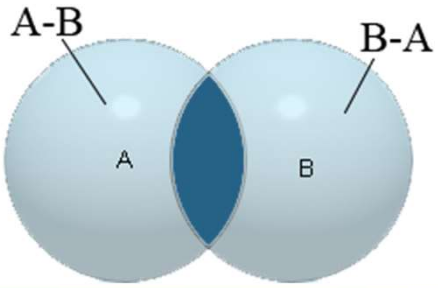
- **Output:**

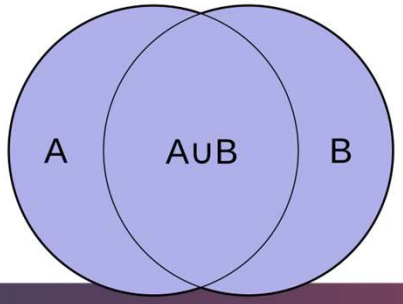
Salary
30000
40000
25000
43000
38000
55000

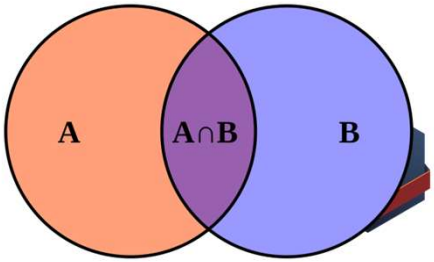
Tables as Sets (Cont.)

Applicable to union-compatible relations i.e. the two relations should have the same attributes and in the same order in both relations.

SET OPERATIONS		
Operation	**Relational Algebra**	**SQL**
Difference	-	EXCEPT
Union	U	UNION
Intersection	∩	INTERSECT







Example

- **Query:** 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, DEPARTMENT, EMPLOYEE
WHERE Dnum=Dnumber AND Mgr_ssn=Ssn AND Lname=‘Smith’ )
UNION
( SELECT DISTINCT Pnumber
FROM PROJECT, WORKS_ON, EMPLOYEE
WHERE Pnumber=Pno AND Essn=Ssn AND Lname=‘Smith’ );
```

- **Output:**

Pnumber
1
2

Substring Pattern Matching

THE NORTHCAP UNIVERSITY

- **Query:** Retrieve all employees whose address is in Houston, Texas.

```
SELECT Fname, Lname
FROM EMPLOYEE
WHERE Address LIKE '%Houston,TX%';
```

Used for string ***pattern matching***

replaces an arbitrary number of zero or more characters

- **Output:**

Fname	Lname
John	Smith
Franklin	Wong
Joyce	English
Ahmad	Jabbar
James	Borg

30 YEARS OF EXCELLENCE | NAAC Accredited

Substring Pattern Matching (Cont.)

- **Query:** Find all employees who were born during the 1950s.

```
SELECT Fname, Lname
FROM EMPLOYEE
WHERE Bdate LIKE '_ _ 5 _ _ _ _ _ _ _';
```

Used for string ***pattern matching***

replaces a single character

- **Output:**

Fname	Lname
Franklin	Wong

Arithmetic Operators

- **Query:** Show the resulting salaries if every employee working on the 'ProductX' project is given a 10 percent raise.

```
SELECT E.Fname, E.Lname, 1.1 * E.Salary AS Increased_sal
FROM EMPLOYEE AS E, WORKS_ON AS W, PROJECT AS P
WHERE E.Ssn=W.Essn AND W.Pno=P.Pnumber AND
      P.Pname='ProductX';
```

- **Output:**

Fname	Lname	Increased_sal
John	Smith	33000
Joyce	English	27500

Arithmetic Operators (Cont.)

- **Query:** 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;
```

((Salary >= 30000) **AND** (Salary <= 40000))

- **Output:**

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
Ramesh	K	Narayan	666884444	1962-09-15	975 Fire Oak, Humble, TX	M	38000	333445555	5

Ordering of Query Results

- **Query:** Retrieve a list of employees and the projects they are working on, ordered by department and, within each department, ordered alphabetically by last name, then first name.

```
SELECT D.Dname, E.Lname, E.Fname, P.Pname
FROM DEPARTMENT D, EMPLOYEE E, WORKS_ON W, PROJECT P
WHERE D.Dnumber=E.Dno AND E.Ssn=W.Essn AND W.Pno=
                                    P.Pnumber
ORDER BY D.Dname, E.Lname, E.Fname;
```

- *Default order is ascending.*
- *First order on Dname, if Dname same then on Lname, if Lname same then on Fname*
- *Use ASC or DESC to change default order:* ORDER BY D.Dname DESC, E.Lname ASC, E.Fname ASC

Ordering of Query Results (Cont.)

- **Output:**

Dname	Lname	Fname	Pname
Administration	Jabbar	Ahmad	Computerization
Administration	Jabbar	Ahmad	NewBenefits
Administration	Wallace	Jennifer	NewBenefits
Administration	Wallace	Jennifer	Reorganization
Administration	Zelaya	Alicia	Computerization
Administration	Zelaya	Alicia	NewBenefits
Headquarters	Borg	James	Reorganization
Research	English	Joyce	ProductX
Research	English	Joyce	ProductY
Research	Narayan	Ramesh	ProductZ
Research	Smith	John	ProductX
Research	Smith	John	ProductY
Research	Wong	Franklin	ProductY
Research	Wong	Franklin	ProductZ
Research	Wong	Franklin	Computerization
Research	Wong	Franklin	Reorganization

Summary

SELECT <attribute list>

FROM <table list>

[**WHERE** <condition>]

[**ORDER BY** <attribute list>];

OPTIONAL

The DELETE Command

- **DELETE FROM** EMPLOYEE

  WHERE Lname='Brown';

- **DELETE FROM** EMPLOYEE

  WHERE Ssn='123456789';

- **DELETE FROM** EMPLOYEE;

To specify the tuples to be deleted

missing WHERE clause specifies that all tuples in the relation are to be deleted. The table remains in the database as an empty table.

The UPDATE Command

- **UPDATE** PROJECT

  SET Plocation = ‘Bellaire’, Dnum = 5

  WHERE Pnumber=10;

- **UPDATE** EMPLOYEE

  SET Salary = Salary * 1.1

  WHERE Dno = 5;

Modify attribute values of one or more selected tuples

To specify attributes to be modified and new values

THE NORTHCAP UNIVERSITY

Thanks!!

30 YEARS OF EXCELLENCE

NAAC Accredited