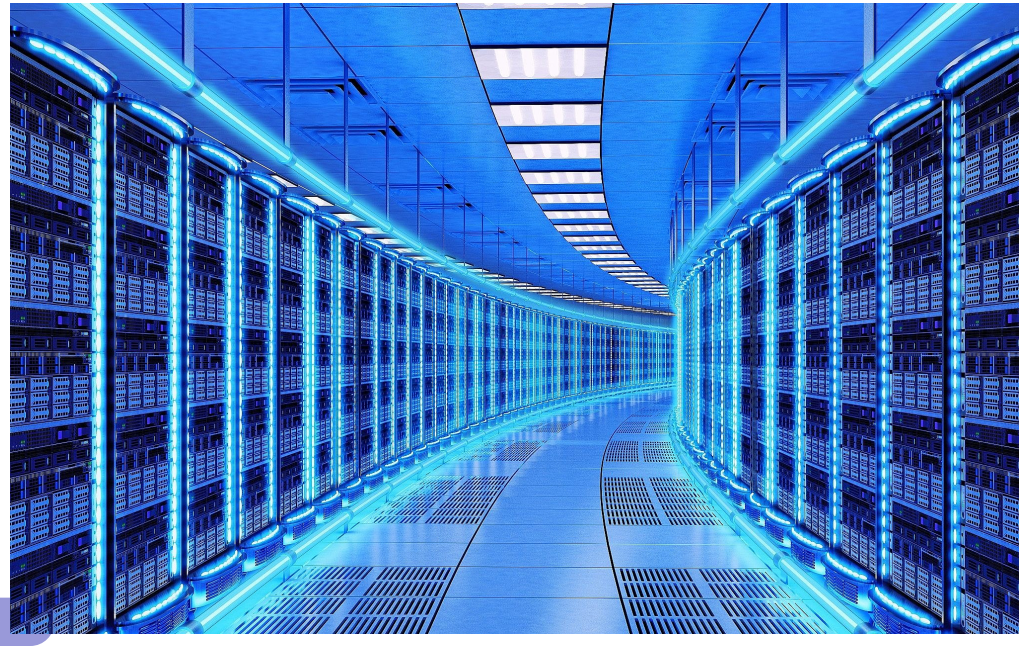
CSC 261/461
Introduction to Databases

Eustrat Zhupa



UNIVERSITY of ROCHESTER

SELECT

SQL has one basic statement for retrieving information from a database: the SELECT statement.

- SELECT statement is *not* the same as the σ operation of relational algebra
- Note: SQL allows a table to have tuples that are identical in all their attribute values.
  - an SQL table is not a set of tuples, but a **multiset** (bag) of tuples.

SELECT

- The basic form of the SELECT statement:

```
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 expression that identifies the tuples to be retrieved by the query.

SELECT

Query 0. Retrieve the *birth date* and *address* of the employee(s) whose name is 'John B. Smith'.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

```
1        SELECT Bdate, Address
2        FROM Employee
3        WHERE Fname='John' AND Minit='B' AND Lname='Smith';
```

SELECT

```
SELECT Bdate, Address
FROM Employee
WHERE Fname='John' AND Minit='B' AND Lname='Smith';
```

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

SELECT

Query 1. Retrieve the *name* and *address* of all employees who work for the 'Research' department.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

```
SELECT Fname, Lname, Address
FROM Employee, Department
WHERE Dname='Research' AND Dnumber=Dno;
```

SELECT

```
SELECT Fname, Lname, Address
FROM Employee, Department
WHERE Dname='Research' AND Dnumber=Dno;
```

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

SELECT

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 birth date.

EMPLOYEE

Fname	Minit	Lname	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dno

DEPARTMENT

Dname	Dnumber	Mgr_ssn	Mgr_start_date

PROJECT

Pname	Pnumber	Plocation	Dnum

```
SELECT Pnumber, Dnum, Lname, Address, Bdate
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE Dnum=Dnumber
        AND Mgr_ssn=Ssn AND Plocation='Stafford';
```

Ambiguities

Query 1. Retrieve the *name* and *address* of all employees who work for the 'Research' department.

EMPLOYEE

Fname	Minit	Name	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dnumber

DEPARTMENT

Name	Dnumber	Mgr_ssn	Mgr_start_date

Note: Suppose `Dno` and `Lname` attributes of the `Employee` relation were called `Dnumber` and `Name`, and the `Dname` attribute of `Department` was also called `Name`.

```
SELECT Fname, Name, Address
FROM Employee, Department
WHERE Name='Research'
    AND Dnumber=Dnumber;
```

Ambiguities

Query 1. Retrieve the *name* and *address* of all employees who work for the 'Research' department.

EMPLOYEE

Fname	Minit	Name	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dnumber

DEPARTMENT

Name	Dnumber	Mgr_ssn	Mgr_start_date

Note: Suppose `Dno` and `Lname` attributes of the `Employee` relation were called `Dnumber` and `Name`, and the `Dname` attribute of `Department` was also called `Name`.

```
SELECT Fname, Employee.Name, Address
FROM Employee, Department
WHERE Department.Name='Research'
    AND Department.Dnumber=Employee.Dnumber;
```

Ambiguities

Query 1. Retrieve the *name* and *address* of all employees who work for the 'Research' department.

EMPLOYEE

Fname	Minit	Name	Ssn	Bdate	Address	Sex	Salary	Super_ssn	Dnumber

DEPARTMENT

Name	Dnumber	Mgr_ssn	Mgr_start_date

Note: Suppose `Dno` and `Lname` attributes of the `Employee` relation were called `Dnumber` and `Name`, and the `Dname` attribute of `Department` was also called `Name`.

```
SELECT Fname, Employee.Name, Address
FROM Employee, Department
WHERE Department.Name='Research'
    AND Department.Dnumber=Employee.Dnumber;
```

No-where

Queries 4 and 5. Select all Employee Ssns (Q4) and all combinations of Employee Ssn and Department Dname (Q5) in the database.

```
SELECT Ssn
FROM Employee;

SELECT Ssn, Dname
FROM Employee, Department;
```

All Attributes

```
SELECT *
FROM Employee
WHERE Dno=5

SELECT *
FROM Employee, Department
WHERE Dname='Research' AND Dno=Dnumber

SELECT *
FROM Employee, Department
```

Sets and Multisets

- SQL usually treats a table not as a set
- duplicate tuples can appear more than once in a table, and in the result of a query.
- SQL keeps duplicate tuples. Why?
  - Duplicate elimination is an expensive operation. One way to implement it is to sort the tuples first and then eliminate duplicates.
  - The user may want to see duplicate tuples in the result of a query.
  - When an aggregate function is applied to tuples, in most cases we do not want to eliminate duplicates.

No duplicates

- An SQL table with a key is restricted to being a set
  - the key value must be distinct in each tuple.
- to get rid of duplicates use DISTINCT in the SELECT clause,
  - only distinct tuples should remain in the result.
- SELECT ALL is equivalent to SELECT

No duplicates

- Q11 retrieves the salary of every employee

```
SELECT ALL Salary
FROM Employee;
```

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

Structured Query Language

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

```
SELECT ALL Salary
FROM Employee;

SELECT DISTINCT Salary
FROM EMPLOYEE;
```

(a)

Salary
30000
40000
25000
43000
38000
25000
25000
55000

(b)

Salary
30000
40000
25000
43000
38000
55000

Structured Query Language

- **Query 12:** 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' );
```

Arithmetic Operators

- Standard arithmetic operators can be applied to numeric values
  - addition (+)
  - subtraction (-)
  - multiplication (*)
  - division (/)

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';
```

Structured Query Language

- Another comparison operator is BETWEEN.

  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;
```

- The condition (Salary BETWEEN 30000 AND 40000) is equivalent to ((Salary $>=$ 30000) AND (Salary $<=$ 40000))

Structured Query Language

- In SQL you can order tuples in the result by the values of one or more of the attributes that appear in the query result, with ORDER BY.
- **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;
```

- To change order
  ORDER BY D.Dname DESC, E.Lname ASC, E.Fname ASC