Outline

- Data definition
- Query
- Data update
- View definition
Basic Retrieval Queries in SQL

- Basic form of the SQL SELECT statement is called a *mapping* or a *SELECT-FROM-WHERE block*

  \[
  \text{SELECT} \quad \langle \text{attribute list} \rangle \\
  \text{FROM} \quad \langle \text{table list} \rangle \\
  \text{WHERE} \quad \langle \text{condition} \rangle \\
  \]

- Declarative as relational calculus
  \[
  \{ \langle \text{attribute list} \rangle \mid \langle \text{table list and condition} \rangle \}
  \]

- Can be translated to relational algebra expression
  \[
  \pi_{\langle \text{attribute list} \rangle} \sigma_{\langle \text{condition} \rangle} (\text{Cartesian product of } \langle \text{table list} \rangle)
  \]

- Does not remove duplicates as SELECT in relational algebra
Figure 9.2
Result of mapping the COMPANY ER schema into a relational database schema.
Simple SQL Queries

Query 0: Retrieve the birthdate and address of the employee whose name is 'John B. Smith'.

Q0: SELECT BDATE, ADDRESS
    FROM EMPLOYEE
    WHERE FNAME='John' AND MINIT='B'
    AND LNAME='Smith'

- Similar to a SELECT-PROJECT pair of relational algebra operations;
- However, the result of the query may contain duplicate tuples
A missing WHERE-clause indicates no condition; hence, all tuples of the relations in the FROM-clause are selected. This is equivalent to the condition WHERE TRUE

Query 9: Retrieve the SSN values for all employees.

Q9: SELECT SSN
    FROM EMPLOYEE

What if more than one relation is specified in the FROM-clause and there is no join condition?

Q10: SELECT SSN, DNAME
    FROM EMPLOYEE, DEPARTMENT

How to retrieve the ssn and department name of each employee?
Query 1: Retrieve the ssn and name of all employees who work for the 'Research' department.

Q1: SELECT SSN, FNAME, LNAME
    FROM EMPLOYEE, DEPARTMENT
    WHERE DNAME='Research' AND DNUMBER=DNO

- Similar to a SELECT-PROJECT-JOIN sequence of relational algebra operations
- (DNAME='Research') is a selection condition (corresponds to a SELECT operation in relational algebra)
- (DNUMBER=DNO) is a join condition (corresponds to a JOIN operation in relational algebra)
Simple SQL Queries (cont.)

- **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.

  Q2: SELECT PNUMBER, DNUM, LNAME, BDATE, ADDRESS
       FROM PROJECT, DEPARTMENT, EMPLOYEE
       WHERE DNUM=DNUMBER AND MGRSSN=SSN
       AND PLOCATION='Stafford'

- The join condition DNUM=DNUMBER relates a project to its controlling department
- The join condition MGRSSN=SSN relates the controlling department to the employee who manages that department
Qualifying Attribute Names

- In SQL, we can use the same name for two (or more) attributes as long as the attributes are in different relations.
- A query that refers to two or more attributes with the same name must qualify the attribute name with the relation name by prefixing the relation name to the attribute name.

Example:

EMPLOYEE.LNAME, DEPARTMENT.DNAME
USE OF *

To retrieve all the attribute values of the selected tuples, a * is used, which stands for all the attributes

Examples:

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

```
SELECT * FROM EMPLOYEE, DEPARTMENT
WHERE DNAME='Research' AND DNO=DNUMBER
```
Aliasing Relation Names

- Some queries need to refer to the same relation twice and *aliases* are given to the relation name.
- **Query 8**: For each employee, retrieve the employee's name, and the name of his or her immediate supervisor.
- **Q8**: SELECT E.FNAME, E.LNAME, S.FNAME, S.LNAME FROM EMPLOYEE E, EMPLOYEE S WHERE E.SUPERSSN=S.SSN

- The alternate relation names E and S are called *aliases* or *tuple variables* for the EMPLOYEE relation.
USE OF DISTINCT

- SQL does not treat a relation as a set; *duplicate tuples can appear*
- To eliminate duplicate tuples in a query result, the keyword `DISTINCT` is used
- For example, the result of Q11 may have duplicate SALARY values whereas Q11A does not have any duplicate values

Q11: `SELECT SALARY
      FROM EMPLOYEE`
Q11A: `SELECT DISTINCT SALARY
      FROM EMPLOYEE`
IN Operator – Set Comparison

- It is possible to use a set of values in the WHERE-clause
- Query 17: Retrieve the social security numbers and project numbers of all employees who work on project number 1, 2, or 3.

Q17: SELECT ESSN, PNO
     FROM WORKS_ON
     WHERE PNO IN (1, 2, 3)

Retrieve distinct social security numbers?
LIKE Operator - SUBSTRING COMPARISON

- The **LIKE** comparison operator is used to compare partial strings.
- Two reserved characters are used: '%' replaces an arbitrary number of characters, and '_' replaces a single arbitrary character.
- Retrieve all employees whose address is in Houston, Texas. Here, the value of the ADDRESS attribute must contain the substring 'Houston,TX'.

**Q12:**

```
SELECT FNAME, LNAME
FROM EMPLOYEE
WHERE ADDRESS LIKE '%Houston,TX%'
```
Retrieve all employees who were born during the 1950s. Here, '5' must be the 8th character of the string (according to our format for date), so the BDATE value is '________5_', with each underscore as a place holder for a single arbitrary character.

```
SELECT FNAME, LNAME
FROM EMPLOYEE
WHERE BDATE LIKE '________5_'
```
The standard arithmetic operators '+', '-', '*', and '/' (for addition, subtraction, multiplication, and division, respectively) can be applied to numeric values in an SQL query result.

Show the effect of giving all employees who work on the 'ProductX' project a 10% raise.

Q13: SELECT FNAME, LNAME, 1.1*SALARY FROM EMPLOYEE, WORKS_ON, PROJECT WHERE SSN=ESSN AND PNO=PNUMBER AND PNAME='ProductX'
SQL has directly incorporated some set operations. There is a union operation (UNION), and in some versions of SQL there are set difference (MINUS) and intersection (INTERSECT) operations.

The resulting relations of these set operations are sets of tuples; duplicate tuples are eliminated from the result.

The set operations apply only to union compatible relations; the two relations must have the same attributes and the attributes must appear in the same order.
Query 4: Make a list of all project numbers for projects that involve an employee whose last name is 'Smith' as a worker or as a manager of the department that controls the project.

Q4: (SELECT PNAME
FROM PROJECT, DEPARTMENT, EMPLOYEE
WHERE DNUM=DNUMBER
AND MGRSSN=SSN
AND LNAME='Smith'
) UNION
(SELECT PNAME
FROM PROJECT, WORKS_ON, EMPLOYEE
WHERE PNUMBER=PNO
AND ESSN=SSN
AND LNAME='Smith')
Query 6: Retrieve the names of employees who have no dependents.

Q6: 

(SELECT FNAME, LNAME 
FROM EMPLOYEE) 
MINUS 
(SELECT FNAME, LNAME 
FROM EMPLOYEE, DEPENDENT 
WHERE ESSN = SSN)
NESTING OF QUERIES

- A complete SELECT query, called a *nested query*, can be specified within the WHERE-clause of another query, called the *outer query*.

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

  Q1A: SELECT FNAME, LNAME, ADDRESS
  FROM EMPLOYEE
  WHERE DNO IN (SELECT DNUMBER
      FROM DEPARTMENT
      WHERE DNAME='Research')

- The comparison operator **IN** compares a value v with a set (or multi-set) of values V, and evaluates to **TRUE** if v is one of the elements in V.

- A reference to an *unqualified attribute* refers to the relation declared in the *innermost nested query*. 
If a condition in the WHERE-clause of a nested query references an attribute of a relation declared in the outer query, the two queries are said to be correlated.

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

Q16: SELECT E.FNAME, E.LNAME
FROM EMPLOYEE AS E
WHERE E.SSN IN
  (SELECT ESSN
   FROM DEPENDENT
   WHERE ESSN=E.SSN
   AND E.FNAME=NAME)

Execution of correlated queries:
FOR each tuple (X) in the outer query DO
{ EXECUTE the inner query USING the attribute values of tuple X in the outer query. }
CORRELATED NESTED QUERIES

- A query written with nested SELECT... FROM... WHERE... blocks and using the = or IN comparison operators can always be expressed as a single block query.

Q16A: SELECT E.FNAME, E.LNAME
FROM EMPLOYEE E, DEPENDENT D
WHERE E.SSN=D.ESSN
AND E.FNAME=D.NAME
THE EXISTS FUNCTION

- EXISTS is used to check whether the result of a correlated nested query is empty (contains no tuples) or not.
- **Query 6**: Retrieve the names of employees who have no dependents.

Q6:  

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

- Recall Query 6 can be written using set difference.
SQL uses **IS** or **IS NOT** to compare NULLs because it considers each NULL value distinct from other NULL values, so equality comparison is not appropriate.

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

Q18: SELECT FNAME, LNAME
    FROM EMPLOYEE
    WHERE SUPERSSN IS NULL

Note: If a join condition is specified, tuples with NULL values for the join attributes are not included in the result.
The Three-Value Logic

<table>
<thead>
<tr>
<th>AND</th>
<th>TRUE</th>
<th>FALSE</th>
<th>UNKNOWN</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>FALSE</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>FALSE</td>
<td>FALSE</td>
<td>FALSE</td>
<td>FALSE</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>UNKNOWN</td>
<td>FALSE</td>
<td>UNKNOWN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>OR</th>
<th>TRUE</th>
<th>FALSE</th>
<th>UNKNOWN</th>
</tr>
</thead>
<tbody>
<tr>
<td>TRUE</td>
<td>TRUE</td>
<td>TRUE</td>
<td>TRUE</td>
</tr>
<tr>
<td>FALSE</td>
<td>TRUE</td>
<td>FALSE</td>
<td>UNKNOWN</td>
</tr>
<tr>
<td>UNKNOWN</td>
<td>TRUE</td>
<td>UNKNOWN</td>
<td>UNKNOWN</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>NOT</th>
<th>TRUE</th>
<th>FALSE</th>
<th>UNKNOWN</th>
</tr>
</thead>
<tbody>
<tr>
<td>--- &gt;</td>
<td>FALSE</td>
<td>TRUE</td>
<td>UNKNOWN</td>
</tr>
</tbody>
</table>
Practice

- P1. List the names of all employees who are directly supervised by ‘Frank Wong’
- P2. Retrieve the names of all employees who work on any project
- P3. Retrieve the names of all employees who do not work on any project
- P4. List the names of all employees who have dependent named ‘Alice’