Chapter 7 & 8

SQL Language

SQL Language and Queries

- Structured Query Language (SQL) is a programming language for processing data in relational databases, initially created in 1970s.
- SQL commands are classified as
  - Data Definition Language (DDL)
  - Data Manipulation Language (DML)
  - Data Query Language
  - Data and Transaction Control Languages
- SQL is used for the following
  - Retrieve information from database tables
  - Add, update, and delete data on existing tables
  - Create and modify tables and database structures such as views, index, etc.

SQL Query Examples

- Get color and city for “nonParis” parts with weight greater than ten pounds.

```
SELECT COLOR, CITY
FROM P
WHERE CITY <> ‘Paris’
AND WEIGHT > 10.0;
```

SQL Examples

- For all parts, get the part number and the weight of that part in grams

```
SELECT P_NUM, WEIGHT*454 AS GMWT
FROM P;
```

SQL Examples

- Get all combinations of supplier and part information such that the supplier and part in question are colocated

```
SELECT S.*, P.P_NUM, P.PNAME, P.COLOR, P.WEIGHT
FROM S, P
WHERE S.CITY = P.CITY;
```

SQL Examples

- Get all pairs of supplier numbers such that the two suppliers concerned are colocated

```
SELECT A.S_NUM SA, B.S_NUM SB
FROM S A, S B
WHERE A.CITY = B.CITY
AND A.S_NUM > B.S_NUM;
```

  - Eliminate pairs like (x, x)
  - Guarantees the pairs (x, y) and (y, x) will not both appear

SQL Examples

- Get the total number of suppliers

```
SELECT COUNT(*)  NUM_SUPPLIERS
FROM S;
```

  - Do not use “AS” in mySQL database

SQL Examples

- Get the maximum and minimum quantity for part 2

```
SELECT MAX(QTY) MAXQ, MIN(QTY) MINQ
FROM SPJ
WHERE P_NUM = 'P2';
```

  - Other aggregate functions include AVG, SUM, COUNT

SQL Examples

- For each part supplied, get the part number and the total shipment quantity.

```
SELECT P_NUM, SUM(QTY) TOTQTQ
FROM SPJ
GROUP BY P_NUM;
```

SQL Examples

- Get the part number for parts supplied by more than one supplier.

```
SELECT P_NUM, COUNT(S_NUM)
FROM SPJ
GROUP BY P_NUM
HAVING COUNT(S_NUM) > 1;
```

  - HAVING is used to eliminate groups as WHERE is used to eliminate rows.

SQL Examples

- Get supplier names for suppliers who supply part P2.

```
SELECT DISTINCT S.SNAME
FROM S
WHERE S.S_NUM IN
    (
      SELECT SPJ.S_NUM
      FROM SPJ
      WHERE SPJ.P_NUM = 'P2' )
```

SQL Examples

- Get supplier names for suppliers who supply part P2.

```
SELECT DISTINCT S.SNAME
FROM S
WHERE EXISTS
    (  SELECT *
       FROM SPJ
       WHERE SPJ.P_NUM = 'P2' AND SPJ.S_NUM = S.S_NUM )
```

EXISTS – evaluated to true if and only if (…) is not empty.

SQL Examples

- Get supplier names for suppliers who do not supply part P2.

```
SELECT DISTINCT S.SNAME
FROM S
WHERE NOT EXISTS
    (  SELECT *
       FROM SPJ
       WHERE SPJ.P_NUM = 'P2' AND SPJ.S_NUM = S.S_NUM )
```

NOT EXISTS – evaluated to true if and only if (…) is empty.

SQL Examples

- Get part numbers for parts that either weigh more than 16 pounds or are supplier by supplier P2, or both.

```
SELECT P_NUM
FROM P
WHERE WEIGHT > 16.0
UNION
SELECT P_NUM
FROM SPJ
WHERE S_NUM = 'S2'
```

INTERSECT and MINUS are not available in MySQL, but they can be simulated by other existing operations.

SQL Examples

- Get supplier names for suppliers who supply all parts

```
SELECT DISTINCT S.SNAME
FROM S
WHERE NOT EXISTS
   ( SELECT *
     FROM P
     WHERE NOT EXISTS
        ( SELECT * FROM SPJ
           WHERE SPJ.S_NUM = S.S_NUM AND
                    SPJ.P_NUM = P.P_NUM
        )
   );
```