

Exercise 2

SQL Built-in Functions

Consider the database given below.

Supplier (scode: int, sname: string, scity: string, turnover: int)

Part (pcode: string, weight: int, color: string, cost: int, sellingprice: int)

Supplier_Part (scode: int, pcode: string, qty: int)

Demonstrate with appropriate SQL Statement for the following:

1. Create the above tables
2. Insert at least five suitable tuples for each relation
3. Get the supplier number and part number in ascending order of supplier number.
4. Get the details of supplier who operate from certain place (say Kolkata) with turnover 50.
5. Get the total number of supplier.
6. Get pairs of supplier number such that both operate from the same city.
7. Get the part number (code) weighing between 25 and 35.
8. Get the supplier number whose turnover is null.
9. Get the part number that cost 20, 30 or 40 rupees.
10. Get the supplier number of supplier are located in the same city (say Ahmedabad).
11. Get the part number for part whose cost is less than the current maximum cost.
12. Get the part number whose cost is greater than the average cost.
13. For each part supplied get part number and names of all cities supplying the part.
14. Get the supplier number who supply at least one part supplied by supplier 5.
15. Get the part number for the part supplied by more than one supplier.
16. Get the names of suppliers who supply from city where there is at least one more supplier.
17. Get the part number of all the part being supplied with no duplicates.

Objective:

The objective of this exercise is to revise and enable you to use SQL built-in functions for performing calculations on data using various clauses in certain conditions along with operators.

Procedure and description:

SQL is rich in built-in functions *like Numeric functions, String functions, String / Number Conversion Functions, Formats for TO_CHAR Function, Group Functions, Date and Time Functions, aggregate functions, scalar functions* etc. We use built-in functions for *performing calculations on data*. For e.g., SQL aggregate functions return a single value, calculated from values in a column. SQL contains Aggregate Functions like: AVG(), COUNT(), FIRST(), LAST(), MAX(), MIN(), SUM() etc.

SQL scalar functions return a single value, based on the input value. SQL contains Scalar functions like: UCASE(), LCASE(), MID(), LEN(), ROUND(), NOW(), FORMAT() etc.

Algorithm: The steps for this exercise are given below:

Step – 1: Start

Step – 2: Create tables using CREATE commands with its essential attributes.

Step – 3: Insert the values using INSERT INTO statements. (Insert the suitable values that is required for demonstration.)

Step – 4: Execute different Commands and extract information from the table. (Hint: use built-in functions like MAX, MIN, COUNT etc.. You can use suitable operators like AND & OR for certain conditions to meet. Use various statements of clauses like ORDER BY, GROUP BY etc.)

Expected Output: Creation of tables and its modification through SQL commands.