Lab 3

Update Query, Delete query & Sub Query

CARTESIAN JOIN

The CARTESIAN JOIN or CROSS JOIN returns the cartesian product of the sets of records from the two or more joined tables. Thus, it equates to an inner join where the join-condition always evaluates to True or where the join condition is absent from the statement.

Syntax:

The basic syntax of INNER JOIN is as follows:

SELECT table1.column1, table2.column2...

FROM table1, table2 [, table3]

Example:

SELECT ID, NAME, AMOUNT, DATE

FROM CUSTOMERS, ORDERS;

UPDATE Query

The SQL UPDATE Query is used to modify the existing records in a table.You can use WHERE clause with UPDATE query to update selected rows, otherwise all the rows would be affected.
Syntax:
UPDATE table_name
SET column1 = value1, column2 = value2...., columnN = valueN
WHERE [condition];
You can combine N number of conditions using AND or OR operators.

Example:
UPDATE CUSTOMERS
SET ADDRESS = 'Pune'
WHERE ID = 6;
Output:

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
6	Komal	22	Pune	4500.00
7	Muffy	24	Indore	10000.00

Example:

```
UPDATE CUSTOMERS
SET ADDRESS = 'Pune', SALARY = 1000.00;
```

Output:

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Pune	1000.00
2	Khilan	25	Pune	1000.00
3	kaushik	23	Pune	1000.00
4	Chaitali	25	Pune	1000.00
5	Hardik	27	Pune	1000.00
6	Komal	22	Pune	1000.00
7	Muffy	24	Pune	1000.00

SQL DELETE Query

The SQL DELETE Query is used to delete the existing records from a table.You can use WHERE clause with DELETE query to delete selected rows, otherwise all the records would be deleted.

Syntax:

```
DELETE FROM table_name
WHERE [condition];
```

You can combine N number of conditions using AND or OR operators.

Example:

```
DELETE FROM CUSTOMERS
WHERE ID = 6;
```

Output:

ID	NAME	AGE	ADDRESS	SALARY
1	Ramesh	32	Ahmedabad	2000.00
2	Khilan	25	Delhi	1500.00
3	kaushik	23	Kota	2000.00
4	Chaitali	25	Mumbai	6500.00
5	Hardik	27	Bhopal	8500.00
7	Muffy	24	Indore	10000.00

SQL TRUNCATE TABLE

The SQL TRUNCATE TABLE command is used to delete complete data from an existing table.

You can also use DROP TABLE command to delete complete table but it would remove complete table structure form the database and you would need to re-create this table once again if you wish you store some data.

Syntax:

The basic syntax of TRUNCATE TABLE is as follows:

```
TRUNCATE TABLE table_name;
```

Example:
TRUNCATE TABLE CUSTOMERS
Output:
SELECT * FROM CUSTOMERS;
Empty set (0.00 sec)

SQL Sub Queries

A Sub query or Inner query or Nested query is a query within another SQL query and embedded within the WHERE clause.
A sub query is used to return data that will be used in the main query as a condition to further restrict the data to be retrieved.
Sub queries can be used with the SELECT, INSERT, UPDATE, and DELETE statements along with the operators like =, <, >, >=, <=, IN, BETWEEN etc.
There are a few rules that sub queries must follow:
- Subqueries must be enclosed within parentheses.
- A subquery can have only one column in the SELECT clause, unless multiple columns are in the main query for the subquery to compare its selected columns.
- An ORDER BY cannot be used in a subquery, although the main query can use an ORDER BY. The GROUP BY can be used to perform the same function as the ORDER BY in a subquery.
- Subqueries that return more than one row can only be used with multiple value operators, such as the IN operator.
- The SELECT list cannot include any references to values that evaluate to a BLOB, ARRAY, CLOB, or NCLOB.
- A subquery cannot be immediately enclosed in a set function.
- The BETWEEN operator cannot be used with a subquery; however, the BETWEEN operator can be used within the subquery.

Subqueries with the SELECT Statement:
Subqueries are most frequently used with the SELECT statement. The basic syntax is as follows:

```
SELECT column_name [, column_name ]
FROM table1 [, table2 ]
WHERE column_name OPERATOR
 (SELECT column_name [, column_name ]
FROM table1 [, table2 ]
[WHERE])
```

Example:

```
SELECT *
FROM CUSTOMERS
WHERE ID IN (SELECT ID
FROM CUSTOMERS
WHERE SALARY > 4500) ;
```

Subqueries with the INSERT Statement:

Subqueries also can be used with INSERT statements. The INSERT statement uses the data returned from the subquery to insert into another table. The selected data in the subquery can be modified with any of the character, date or number functions.
The basic syntax is as follows:

```
INSERT INTO table_name [ (column1 [, column2 ]) ]
SELECT [ *|column1 [, column2 ]
FROM table1 [, table2 ]
[ WHERE VALUE OPERATOR ]
```

Example:

```
INSERT INTO CUSTOMERS_BKP
SELECT * FROM CUSTOMERS  WHERE ID IN (SELECT ID FROM CUSTOMERS) ;
```

Subqueries with the UPDATE Statement:
The subquery can be used in conjunction with the UPDATE statement. Either single or multiple columns in a table can be updated when using a subquery with the UPDATE statement.
The basic syntax is as follows:

```
UPDATE table
SET column_name = new_value
[ WHERE OPERATOR [ VALUE ]
(SELECT COLUMN_NAME
FROM TABLE_NAME)
[ WHERE) ]
```

Example:
Update SALARY by 0.25 times in CUSTOMERS table for all the customers whose AGE is greater than or equal to 27:

```
SQL> UPDATE CUSTOMERS
SET SALARY = SALARY * 0.25
WHERE AGE IN (SELECT AGE FROM CUSTOMERS_BKP
WHERE AGE >= 27 );
```

Subqueries with the DELETE Statement:
The subquery can be used in conjunction with the DELETE statement like with any other statements mentioned above.
The basic syntax is as follows:

```
DELETE FROM TABLE_NAME
[ WHERE OPERATOR [ VALUE ]
(SELECT COLUMN_NAME
```

FROM TABLE_NAME)
[WHERE)]

Example:

Assuming, we have CUSTOMERS_BKP table available which is backup of CUSTOMERS table.
Following example deletes records from CUSTOMERS table for all the customers whose AGE is greater than or equal to 27:

```
SQL> DELETE FROM CUSTOMERS
WHERE AGE IN (SELECT AGE FROM CUSTOMERS_BKP
WHERE AGE > 27 );
```

Lab Task:

1. Find all the bookname, publication name and author name where publication name is “Ekta”.
2. Find the teachers name and faculty who issued book on jan 1 ,2015.
3. Find the employee name whose salary is greater then 10000 and faculty is “computer”.
4. Add attribute bid on Issues relation .
5. Insert the data in bid column.
6. Find the Teacher’s name , and book name issued by the teacher whose name starts with “S”.
7. Update all salary by 10 %.
8. Update book name DBMS as DATABASE.
9. Update the salary of all employee by 20% whose salary is less than 5000.
10. Provide 5% increment to all salaries whose salary is greater than 20000 and 20% increment in rest of salaries.
    (use CASE WHEN <CONDITION>THEN <STATEMENT> ELSE<STATEMENT> END).
11. Delete the records from employee table whose eid is 111.
12. Use sub query to find all teachers name and faculty whose date of employee is jan 2., 2011
13. Use sub query to find all the book name and author name whose publication is “shaja prakashan”.