Lab 5 – DML Commands

Objective: To understand basic DML. Insert rows using insert command, Update using update command and delete using delete command.

Exercise -1:

1. **Insert 5 rows into the tables for the database created of your case study.**

Ans 1) Assuming we have set up the tables (AD_ACADEMIC_SESSIONS, AD_DEPARTMENTS, AD_PARENT_INFORMATION).

```
CREATE TABLE AD_ACADEMIC_SESSIONS (
    ID INT PRIMARY KEY,
    NAME VARCHAR(255)
);

CREATE TABLE AD_DEPARTMENTS (
    ID INT PRIMARY KEY,
    NAME VARCHAR(255),
    HEAD VARCHAR(255)
);
```

```
CREATE TABLE AD_PARENT_INFORMATION (
    ID INT PRIMARY KEY,
    PARENT1_FN VARCHAR(255),
    PARENT1_LN VARCHAR(255),
    PARENT2_FN VARCHAR(255),
    PARENT2_LN VARCHAR(255)
);
```

AD_ACADEMIC_SESSIONS

ID	NAME
empty	

AD_DEPARTMENTS

ID	NAME	HEAD
empty		

AD_PARENT_INFORMATION

ID	PARENT1_FN	PARENT1_LN	PARENT2_FN	PARENT2_LN
empty				

Here are the SQL commands:

Inserting 5 rows into AD_ACADEMIC_SESSIONS

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (400, 'WINTER SESSION');

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (500, 'AUTUMN SESSION');

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (600, 'MONSOON SESSION');

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (700, 'RAINY SESSION');

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (800, 'DRY SEASON SESSION');

AD_ACADEMIC_SESSIONS

ID	NAME
400	WINTER SESSION
500	AUTUMN SESSION
600	MONSOON SESSION
700	RAINY SESSION
800	DRY SEASON SESSION

Inserting 5 rows into AD_DEPARTMENTS

INSERT INTO AD_DEPARTMENTS (ID, NAME, HEAD) VALUES (50, 'PHYSICS', 'JOHN DOE');

INSERT INTO AD_DEPARTMENTS (ID, NAME, HEAD) VALUES (60, 'CHEMISTRY', 'JANE DOE');

INSERT INTO AD_DEPARTMENTS (ID, NAME, HEAD) VALUES (70, 'MATHEMATICS', 'ALAN SMITH');

INSERT INTO AD_DEPARTMENTS (ID, NAME, HEAD) VALUES (80, 'HISTORY', 'EMMA JONES');

INSERT INTO AD_DEPARTMENTS (ID, NAME, HEAD) VALUES (90, 'GEOGRAPHY', 'OLIVER BLACK');

AD_DEPARTMENTS

ID	NAME	HEAD
50	PHYSICS	JOHN DOE
60	CHEMISTRY	JANE DOE
70	MATHEMATICS	ALAN SMITH
80	HISTORY	EMMA JONES
90	GEOGRAPHY	OLIVER BLACK

Inserting 5 rows into AD_PARENT_INFORMATION

INSERT INTO AD_PARENT_INFORMATION (ID, PARENT1_FN, PARENT1_LN, PARENT2_FN, PARENT2_LN) VALUES (650, 'LUCAS', 'MARTIN', 'EMILY', 'MARTIN');

INSERT INTO AD_PARENT_INFORMATION (ID, PARENT1_FN, PARENT1_LN, PARENT2_FN, PARENT2_LN) VALUES (660, 'JAMES', 'MOORE', 'MARY', 'MOORE');

INSERT INTO AD_PARENT_INFORMATION (ID, PARENT1_FN, PARENT1_LN, PARENT2_FN, PARENT2_LN) VALUES (670, 'LILY', 'ANDERSON', 'JOHN', 'ANDERSON');

INSERT INTO AD_PARENT_INFORMATION (ID, PARENT1_FN, PARENT1_LN, PARENT2_FN, PARENT2_LN) VALUES (680, 'DAVID', 'BROWN', 'SARA', 'BROWN');

INSERT INTO AD_PARENT_INFORMATION (ID, PARENT1_FN, PARENT1_LN, PARENT2_FN, PARENT2_LN) VALUES (690, 'HARRY', 'CLARK', 'LISA', 'CLARK');

AD_PARENT_INFORMATION

ID	PARENT1_FN	PARENT1_LN	PARENT2_FN	PARENT2_LN
650	LUCAS	MARTIN	EMILY	MARTIN
660	JAMES	MOORE	MARY	MOORE
670	LILY	ANDERSON	JOHN	ANDERSON
680	DAVID	BROWN	SARA	BROWN
690	HARRY	CLARK	LISA	CLARK

Exercise -2:

1. **Create the tables mentioned below and insert the rows as shown. Please assume the datatype and constraints as required.**

AD_ACADEMIC_SESSIONS:

ID	NAME
100	SPRING SESSION
200	FALL SESSION
300	SUMMER SESSION

AD_DEPARTMENTS:

ID	NAME	HEAD
10	ACCOUNTING	MARK SMITH
20	BIOLOGY	DAVE GOLD
30	COMPUTER SCIENCE	LINDA BROWN
40	LITERATURE	ANITA TAYLOR

AD_PARENT_INFORMATION: (Hint: must return to READ/WRITE status)

ID	PARENT1_FN	PARENT1_LN	PARENT2_FN	PARENT2_LN
600	NEIL	SMITH	DORIS	SMITH
610	WILLIAM	BEN	NITA	BEN
620	SEAN	TAYLOR	RHEA	TAYLOR
630	DAVE	CARMEN	CATHY	CARMEN
640	JOHN	AUDRY	JANE	AUDRY

2. **Add 2 new rows in AD_ACADEMICS_SESSIONS table with name as "Summer Break Session" and "Winter Break Session".**

Ans 2)

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (900, 'Summer Break Session');

INSERT INTO AD_ACADEMIC_SESSIONS (ID, NAME) VALUES (1000, 'Winter Break Session');

AD_ACADEMIC_SESSIONS

ID	NAME
100	SPRING SESSION
200	FALL SESSION
300	SUMMER SESSION
900	Summer Break Session
1000	Winter Break Session

3. **Update the name “Computer Science” with “Computer Science and Engineering” in AD_DEPARTMENTS table.**

Ans 3)
```
UPDATE AD_DEPARTMENTS
SET NAME = 'Computer Science and Engineering'
WHERE NAME = 'COMPUTER SCIENCE';
```

AD_DEPARTMENTS

ID	NAME	HEAD
10	ACCOUNTING	MARK SMITH
20	BIOLOGY	DAVE GOLD
30	Computer Science and Engineering	LINDA BROWN
40	LITERATURE	ANITA TAYLOR

4. **Update the PARENT1_LN as NULL for ID 620 in AD_PARENT_INFORMATION table.**

Ans 4)
```
UPDATE AD_PARENT_INFORMATION
SET PARENT1_LN = NULL
WHERE ID = 620;
```

AD_PARENT_INFORMATION

ID	PARENT1_FN	PARENT1_LN	PARENT2_FN	PARENT2_LN
600	NEIL	SMITH	DORIS	SMITH
610	WILLIAM	BEN	NITA	BEN
620	SEAN		RHEA	TAYLOR
630	DAVE	CARMEN	CATHY	CARMEN
640	JOHN	AUDRY	JANE	AUDRY

5. **Delete the 2 new rows added in question 2.**

Ans 5)
```
DELETE FROM AD_ACADEMIC_SESSIONS
WHERE NAME IN ('Summer Break Session', 'Winter Break Session');
```

AD_ACADEMIC_SESSIONS

ID	NAME
100	SPRING SESSION
200	FALL SESSION
300	SUMMER SESSION

Lab 6 – DQL and Sorting Data

Objective: To understand basics of DQL commands. Select, Conditional retrieval, operators, pattern matching, order by clause.

Use the default schema of EMP Table & DEPT Table of the database and implement the listed queries:

Exercise -1: Queries based on Conditional Retrieval of Rows

1. **List department names and location from the department table.**

Ans 1) SELECT DNAME, LOC FROM DEPT;

2. **List the employees belonging to the department 20.**

Ans 2) SELECT * FROM EMP WHERE DEPTNO = 20;

3. **List the name and salary of the employees whose salary is more than 1000.**

Ans 3) SELECT ENAME, SAL FROM EMP WHERE SAL > 1000;

4. **List the employee number and name of managers.**

Ans 4) SELECT EMPNO, ENAME FROM EMP WHERE JOB = 'MANAGER';

5. **List the name of clerks working in the department 20.**

Ans 5) SELECT ENAME FROM EMP WHERE JOB = 'CLERK' AND DEPTNO = 20;

6. **List the names of analysts and salesmen.**

Ans 6) SELECT ENAME FROM EMP WHERE JOB IN ('ANALYST', 'SALESMAN');

7. **List the details of the employees who have joined before the end of September 1981.**

Ans 7) SELECT * FROM EMP WHERE HIREDATE < '1981-09-30';

8. **List the names of employees who are not managers.**

Ans 8) SELECT ENAME FROM EMP WHERE JOB <> 'MANAGER';

Exercise -2: Special Operators IN and BETWEEN

1. **List the name of the employee whose employee numbers are 7369,7521,7839,7934, 7788.**

Ans 1) SELECT ENAME FROM EMP WHERE EMPNO IN (7369, 7521, 7839, 7934, 7788);

2. **List the employee details not belonging to the department 10, 30 and 40.**

Ans 2) SELECT * FROM EMP WHERE DEPTNO NOT IN (10, 30, 40);

3. **List the employee name and salary, whose salary is between 1000 and 2000.**

Ans 3) SELECT ENAME, SAL FROM EMP WHERE SAL BETWEEN 1000 AND 2000;

4. **List employee names, who have joined before 30th June 81 and after December 81.**

Ans 4)

SELECT ENAME FROM EMP WHERE HIREDATE < '1981-06-30' OR HIREDATE > '1981-12-31';

Exercise -3: DISTINCT Clause with SELECT

1. **List the different jobs (designations) available in the EMP table.**

Ans 1) SELECT DISTINCT JOB FROM EMP;

Exercise -4: Working with NULL Values

1. **List the employee names, who are not eligible for commission.**

Ans 1) SELECT ENAME FROM EMP WHERE COMM IS NULL;

2. **List the name of the employees and designation (job) of the employee, who does not report to anybody (manager is NULL).**

Ans 2) SELECT ENAME, JOB FROM EMP WHERE MGR IS NULL;

3. **List the employees not assigned to any department.**

Ans 3) SELECT * FROM EMP WHERE DEPTNO IS NULL;

4. **List the employees who are eligible for commission.**

Ans 4) SELECT ENAME FROM EMP WHERE COMM IS NOT NULL;

5. **List the details of employees, whose salary is greater than 2000 and commission is NULL.**

Ans 5) SELECT * FROM EMP WHERE SAL > 2000 AND COMM IS NULL;

Exercise -5: Matching Pattern with Column

1. **List the employees whose names start with an “S”.**

Ans 1) SELECT ENAME FROM EMP WHERE ENAME LIKE 'S%';

2. **List the employees names ending with an “S”.**

Ans 2) SELECT ENAME FROM EMP WHERE ENAME LIKE '%S';

3. **List the names of employees whose names have exactly 5 Characters.**

Ans 3) SELECT ENAME FROM EMP WHERE LENGTH(ENAME) = 5;

4. List the employee names having "I" as the second character.

Ans 4) SELECT ENAME FROM EMP WHERE ENAME LIKE '_I%';

Exercise -6: Using Expression with Column

1. List the name, salary and PF amount of all the employees (PF is calculated as 10% of salary).

Ans 1) SELECT ENAME, SAL, SAL * 0.1 AS PF_AMOUNT FROM EMP;

2. List the names of employees, who are more than 2 years old in the organization.

Ans 2) SELECT ENAME FROM EMP WHERE HIREDATE <= DATEADD(YEAR, -2, GETDATE());

Exercise -7: Ordering the Results of a query

ORDER BY clause to impose an order on the result of a query,

ORDER BY clause used with **SELECT** statement.

SYNTAX: SELECT [DISTINCT] <col list>| <exp> FROM table name WHERE cond ORDER BY col [ASC|DESC].

One or more column can be specified in ORDER BY clause.

1. List the empno, ename, sal in ascending order of salary.

Ans 1) SELECT EMPNO, ENAME, SAL FROM EMP ORDER BY SAL ASC;

2. List the employee name and hiredate in descending order of Hiredate.

Ans 2) SELECT ENAME, HIREDATE FROM EMP ORDER BY HIREDATE DESC;

3. List the employee name, salary, job and department No. in ascending order of Department No. in ascending order of Department No and then on descending descending order of salary.

Ans 3) SELECT ENAME, SAL, JOB, DEPTNO FROM EMP ORDER BY DEPTNO ASC, SAL DESC;