# **Lab Assignment 8**

1. Create table production\_details consisting attributes: emp\_id, f\_name, l\_name, job\_type, and salary.

🡺 CREATE TABLE production\_details AS SELECT emp\_id, f\_name, l\_name, job\_type, salary FROM employee WHERE D\_Name = 'Production';

1. Populate the production\_details table with values from the employee table with details of the employees who are working in the production department.

🡺 INSERT INTO production\_details SELECT emp\_id, f\_name, l\_name, job\_type, salary FROM employee WHERE D\_Name = 'Production';

1. Update Chitra's Job\_type to CEO in production\_details table.

🡺 UPDATE production\_details SET job\_type = 'CEO' WHERE f\_name = 'Chitra';

1. Update employee 3's job and salary in production\_details table to match that of employee 4's of employee table.

🡺 UPDATE production\_details pd SET job\_type = (SELECT job\_type FROM employee WHERE emp\_id = 4), salary = (SELECT salary FROM employee WHERE emp\_id = 4)

WHERE emp\_id = 3;

1. Delete employee 5's details from production\_details.

🡺 DELETE FROM production\_details WHERE emp\_id = 5;

1. Delete the employee's details in production\_details where the salary is greater than of Dheeraj's salary in the employee table.

🡺 DELETE FROM production\_details WHERE salary > (SELECT salary FROM employee WHERE f\_name = 'Dheeraj');

1. Similar to question 1 and 2, Create and populate a different table for the Marketing department (named marketing\_details) with attributes emp\_id, f\_name, job\_type and salary.

🡺 CREATE TABLE marketing\_details AS SELECT emp\_id, f\_name, job\_type, salary FROM employee WHERE D\_Name = 'Marketing';

1. Update table marketing\_details and Make Barun 'CEO' of the company.

🡺 UPDATE marketing\_details SET job\_type = 'CEO' WHERE f\_name = 'Barun';

1. Increase the salary of the employees by 50% who are working as salesman in the marketing\_details table.

🡺 UPDATE marketing\_details SET salary = salary \* 1.5 WHERE job\_type = 'Salesman';

1. Update the employee table using the marketing\_details table. (using merge statement)

🡺 MERGE INTO employee e USING marketing\_details md ON (e.emp\_id = md.emp\_id) WHEN MATCHED THEN UPDATE SET e.job\_type = md.job\_type, e.salary = md.salary;

1. Show the data from production\_details.

🡺 SELECT \* FROM production\_details;

1. Commit the database.

🡺 COMMIT;

1. Delete a tuple from production\_details where emp\_id=1.

🡺 DELETE FROM production\_details WHERE emp\_id = 1;

1. Show the data from production\_details.

🡺 SELECT \* FROM production\_details;

1. Rollback the database to the previous saved state.

🡺 ROLLBACK;

1. Show the data from production\_details.

🡺 SELECT \* FROM production\_details;

1. Create a savepoint (named abc).

🡺 SAVEPOINT abc;

1. Delete a tuple from production\_details where emp\_id=1.

🡺 DELETE FROM production\_details WHERE emp\_id = 1;

19. Show the data from production\_details.

🡺 SELECT \* FROM production\_details;

20. Rollback the database to abc.

🡺 ROLLBACK TO abc;

21. Show the data from production\_details.

🡺 SELECT \* FROM production\_details;