**Asgn 2.** Create an employee database as follows:

Employee (e-id, name, street,city)

Works (e-id, company\_name, salary)

Company (company\_name, city)

Manages (e-id, dept\_name)

1. Create a table for above schemas. (It should include table create, drop, alter and update commands)

2. Create a view for listing all employees working in ‘ABC’ company. Alter this view for listing all employees of ‘ABC’ company having salary greater than 50000 Rs. Rename the view. Perform DML (insert, delete and update) operations on views.

3. Create, alter and drop index on Employee and Company table.

4. Create sequence for required columns.

**Asgn 3.** Design at least 10 SQL queries for suitable database application using SQL DML statements:

Insert, Select, Update, Delete with operators, functions, and set operator.

**Note: All 20 queries are mandatory.**

1. Insert values into above created tables

2. Insert values into only selected columns (e.g. in Employee table insert values for only street and city)

3. Select all values from Employee and Company tables.

4. Select employee ID and name from employee table.

5. Select names of all employees who are managing departments.

6. Assume company is situated in many cities and accordingly have entries into respective table. Now select list of all distinct cities (e.g. “Mumbai”) from Company table.

7. Update the address(street, city) of employee whose E-ID is “12345”.

8. Update (increase) the salary of employees working in “Amazon” by 10000 Rs.

9. Delete records of all employees having salary between 20000 Rs and 50000 Rs.

10. Select all company names where city name starts with ‘P’.

11. List all employees from (“Amazon”, “Flipkart”, “Google”) companies.

12. What is the average salary of employees working in “ABC Corp”.

13. Find total number of employees working in “Amazon”.

14. Find Maximum and Minimum salary of employee in “Google”.

15. Find total amount spent on salaries of employee of “ABC Corp”.

16. List all details of company, sorted ascending by company name and descending by city.

17. List all employee names from “Amazon” and “Google” using union.

18. List all employee names (allow duplicate values for names) from “Amazon” and “Google” using union all

19. List all employee details who are managers (Emulate intersect)

20. List all employee details who are not managers (emulate minus)