1. Write a query to find employees who were hired between January 1, 1982, and December 31, 1983, and whose salaries are between 1000 and 3000.

```
- SELECT *
  FROM EMP
  WHERE hiredate between '1982-01-01' AND '1983-12-31'
  AND
  Sal between 1000 AND 3000;
```

2. Write a query to find employees hired between February 1, 1981, and March 1, 1981, and display their names, jobs, and salaries, ordered by hire date.

```
- SELECT ename, job, sal
  FROM emp
  WHERE hiredate BETWEEN '1981-02-01' AND '1981-03-01'
  Order by hiredate;
```

3. Write a query to find employees whose manager ID is between 7698 and 7839 and who earn a salary between 1000 and 2000.

```
- Select * from emp
  Where mgr between 7698 and 7839
  AND
  Sal between 1000 and 2000;
```

4. Write a query to select employees hired between December 1, 1981, and January 31, 1982, and whose commission is between 0 and 500.

```
- Select * from emp
  Where hiredate between '1981-12-01' and '1982-01-31'
  And
  comm between 0 and 500;
```

5. Write a query to find employees in department 20 with hire dates between 1981-01-01 and 1981-12-31 and whose salaries are between 1000 and 3000.

```
- Select * from emp
  Where deptno = 20
  AND
  Hiredate between '1981-01-01' and '1981-12-31'
  AND
  Sal between 1000 and 3000;
```

6. Write a query to find employees whose job is either 'CLERK', 'MANAGER', or 'SALESMAN', and whose salary is greater than 1000.

```
- SELECT * FROM EMP
  WHERE JOB IN ( 'CLERK' ,' MANAGER' ,' SALESMAN' )
  AND
  SAL > 1000;
```

7. Write a query to select employees from departments 10, 20, and 30 whose commission is not NULL.

```
- SELECT * FROM EMP
  WHERE DEPTNO IN (10,20,30)
  AND
  COMM IS NOT NULL;
```

8. Write a query to find employees whose manager ID is in (7839, 7698, 7788), and whose job is 'CLERK' or 'ANALYST'.

```
- SELECT * FROM EMP
  WHERE MGR IN (7839, 7698, 7788)
  AND
  JOB IN ( 'CLERK' , 'ANALYST' );
```

9. Write a query to select all employees whose hire date is either 1981-09-08, 1981-02-20, or 1981-12-03 and display their names, jobs, and salaries.

```
- SELECT ENAME, JOB,SAL
  FROM EMP
  WHERE HIREDATE IN (‘1981-09-08’, ‘1981-02-20’,
  ‘1981-12-03’);
```

10. Write a query to display employees whose salary is in (1250, 3000, 1600, 1100) and whose job is 'SALESMAN'.

```
- SELECT * FROM EMP
  WHERE SAL IN (1250,3000,1600,1100)
  AND
  JOB = ‘SALESMAN’;
```

DDL(DATA DEFINITION LANGUAGE)

1. CREATE STATEMENT

Creates new table, database or index*

```
CREATE TABLE DEPARTMENT (
   DEPTNO INT Primary key,
   DNAME VARCHAR(50)
);
```

CREATED A DEPARTMENT TABLE WITH TWO COLUMNS: DEPTNO, DNAME.

2. ALTER STATEMENT

alter statement modifies the structure of existing table.

Operations: adding, modifying, dropping columns

Ex. To add a column

```
ALTER TABLE EMP ADD BONUS DECIMAL(7,2);
```

-Adds a new column BONUS of type decimal to emp table.

Ex. Modify a column

```
Alter table emp modify sal decimal(10,2);
```

-changes the sal column to hold 10 digits and 2 decimal places

Ex. To drop a column

```
ALTER TABLE EMP DROP COLUMN BONUS;
```

Removes the BONUS column from emp table.

3. DROP Statement

The drop statement permanently removes table or database.

```
DROP TABLE emp;
```

Delete the emp table along with its data and structure.

4. TRUNCATE Statement

It removes al rows from a table and keeps the table structure.

```
TRUNCATE TABLE EMP;
```

Features	drop	Truncate
Operation	Delete structure and data	Delete data and keeps structure
Rollback	Cannot be rolled back	Cannot be rolled back
Use case	When table is no longer needed	When clearing all rows of data is required.

- To add an email column in emp table.

  ALTER TABLE EMP ADD EMAIL VARCHAR(100);

- To drop email column

  Alter table emp drop column email;

- To modify the existing column

  To change the limit of ename characters to 100.

  ALTER TABLE EMP MODIFY ENAME VARCHAR(100);

- Rename the table emp-> employees
  Alter table emp rename to employees;
  RENAME table EMPLOYEES TO EMPLOYEE;

- To rename the column name
- ALTER TABLE employee RENAME COLUMN ename TO empname;

Single Row Functions

1. Sqrt(num)
   Returns the square root of a number
   Ex. SELECT sqrt(num);

2. Abs(num)
   Returns the absolute value of the number
   Ex. Select abs(-5);

3. Round(num, n)
   Rounds up the number to n decimal places
   Select round(123.4567,2);

4. Truncate(num,n)
   Truncates a number to n decimal places without rounding.
   Ex. SELECT truncate(123.4567,2);

5. Ceil(num)
   Returns the smallest integer greater than or equal to the number
   - Rounds numbers up to nearest integer.
   - Works even if the number is already an integer (no change)
   Ex. Select ceil(12.3);
   Select ceil(12.0);

6. Floor(num)
   Returns the largest integer less than or equal to the number.
   Select floor(12.8);
   Select floor(12.0);

7. Ascii(str)
   Returns the ascii value of the first character
   Select ascii(‘A’); -> 65

Select ascii('Hello'); -> 72

8. Substr(str, start, length)
   Extracts the substring starting at a specified position.
   Ex. Select substr(‘database’ ,2,4); ->atab
   Select substr(‘Hello’ ,1,2); -> He

9. Concat(str1,str2)
   Joins two strings
   Ex. Select concat(‘Hello’ ,’ TEAM’);
   SELECT CONCAT('NAME: ',EMPNAME) FROM EMPLOYEE;

10. LOWER(STR)
    CONVERT A STRING TO LOWERCASE.
    SELECT LOWER(‘FLYNAUT’);

11. UPPER(STR)
    CONVERT A STRING TO UPERCASE
    SELECT UPPER(‘FlYnAuT’);

12. Instr(str,substr)
    Returns the position of first
    occurrence of substr

    Ex. Select instr(‘Database’ ,
     ‘base’);

13. Trim(str)
    Removes all the leading and trailing
    spaces from a string.

    Ex. Select trim(‘   Hello   ’);

14. Lpad(str,length,padstr)
    Left pads a string with specified
    characters to a given length.
    Ex. SELECT lpad(‘Hello’ ,8,’ *’);

15. Rpad(str, length, padstr)
    right pads a string with specified
    characters to a given length.

16. Length(str)
    select length('Flynaut');-> 7

17. Format(num,d)
    Formats a number to d decimal places
    Rounds the number to the specified decimal places.
    Select format(12345.6789);