Assignment:

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;
```

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;
```

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;
```

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;
```

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;
```

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;
```

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;
```

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’);
```

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’);
```

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

   CREATE new table, database or index*

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

   - Create a department table with two columns: DEPTNO, DEPTNAME

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 the table emp

   EX> MODIFY A COLUMN

```
ALTER TABLE EMP MODIFY SAL DECIMAL(10,2);
```

   - CHANGES THE SALARY 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

It permanently removes the table or database.

```
DROP TABLE EMP;
```

DELETE THE EMP TABLE ALONG WITH ITS DATA AND STRUCTURE.

Flashback*

4. TRUNCATE STATEMENT

IT REMOVES ALL THE ROWS FROM TABLE AND KEEPS THE TABLE STRUCTURE.

```
TRUNCATE TABLE EMP;
```

Features	DROP	TRUNCATE
Operation	DELETE STRUCTURE AND DATA	DELETE THE DATA AND KEEPS THE STRUCTURE
ROLLBACK	Cannot be rolled back	Cannot be rolled back
USE CASE	WHEN TABLE IS NOT REQUIRED ANYMORE	WHEN CLEARING ALL ROWS OF DATA IS REQUIRED.

- TO ADD THE EMAIL COLUMN IN EMP TABLE.

```
ALTER TABLE EMP
ADD EMAIL VARCHAR(30);
```

- 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);
  ```

- To RENAME THE TABLE
  EMP -> EMPLOYEES

  ```
  ALTER TABLE EMP
  RENAME TO EMPLOYEES;
  RENAME TABLE EMP TO EMPLOYEES;
  ```

- TO RENAME THE COLUMN

  ```
  ALTER TABLE EMP
  RENAME COLUMN ENAME TO EMPNAME;
  ```

SINGLE ROW FUNCTIONS

1. SQRT(NUM)

   RETURNS THE SQUARE ROOT OF
   THE NUM.
   EX. SELECT SQRT(NUM);

   ```
   mysql> SELECT SQRT(16);
   +----------+
   | SQRT(16) |
   +----------+
   |        4 |
   +----------+
   ```

2. ABS(NUM)

RETURNS THE ABSOLUTE VALUE OF THE NUMBER

EX. SELECT ABS(-45);

```
mysql> SELECT ABS(-45);
+----------+
| ABS(-45) |
+----------+
|       45 |
+----------+
```

3. ROUND(NUM, N)

ROUNDS UP THE NUM TO N DECIMAL PLACES

SELECT ROUND(123.4567, 2);

```
mysql> SELECT
ROUND(123.4567, 2);
+--------------------+
| ROUND(123.4567, 2) |
+--------------------+
|             123.46 |
+--------------------+
1 row in set (0.00 sec)
```

4. TRUNCATE(NUM,N)

TRUNCATES A NUMBER TO N
DECIMAL PLACES WITHOUT
ROUNDING UP.

EX. SELECT
TRUNCATE(123.4567,2);

5. CONCAT(STR1,STR2)

JOINS TWO STRINGS

EX. SELECT
CONCAT(‘HELLO’,’TEAM’);

```
mysql> SELECT CONCAT('HELLO','TEAM');
+------------------------+
| CONCAT('HELLO','TEAM') |
+------------------------+
| HELLOTEAM              |
+------------------------+
1 row in set (0.00 sec)

mysql> select concat ('NAME: ',empname)
from emp;
+---------------------------+
| concat ('NAME: ',empname) |
+---------------------------+
| NAME: SMITH               |
| NAME: ALLEN               |
| NAME: WARD                |
| NAME: JONES               |
| NAME: MARTIN              |
| NAME: BLAKE               |
| NAME: CLARK               |
| NAME: SCOTT               |
| NAME: KING                |
| NAME: TURNER              |
| NAME: ADAMS               |
| NAME: JAMES               |
| NAME: FORD                |
| NAME: MILLER              |
+---------------------------+
14 rows in set (0.00 sec)
```