DBMS Lab Session-1

-106122076 CSE-B

a. Write an SQL Query to find the year from date.

ANSWER:-

```
SELECT YEAR(CURRENT_DATE()) AS YEAR;
```

OUTPUT:-

```
+----------+
| YEAR   |
+----------+
| 2024     |
+----------+
```

b. Check whether date passed to Query is the date of a given format or not.

ANSWER:-

```
SELECT CASE
        WHEN STR_TO_DATE('2023-09-13','%Y-%m-%d') IS NOT NULL
        THEN 'Valid date'
        ELSE 'Invalid'
END AS CheckFormat;
```

OUTPUT:-

```
+------------------+
| CheckFormat  |
+------------------+
| Valid date       |
+------------------+
```

c. Find the size of the SCHEMA/USER.

ANSWER:-

```
SELECT SUM(ROUND(((DATA_LENGTH + INDEX_LENGTH) / 1024 / 1024), 2))
        AS "SIZE IN MB"
        FROM INFORMATION_SCHEMA.TABLES
        WHERE
        TABLE_SCHEMA = "USER";
```

OUTPUT:-

```
+---------------+
| SIZE IN MB |
+---------------+
|      NULL      |
+---------------+
```

d. Display the current time.

ANSWER:-

```
SELECT CURRENT_TIME();
```

OUTPUT:-

```
+-------------------------+
| CURRENT_TIME() |
+-------------------------+
| 15:42:52                |
+-------------------------+
```

e. Given a date, retrieve the next day's date.

ANSWER:-

```
SELECT DATE_ADD('2024-07-26',INTERVAL 1 DAY) AS Next_day;
```

OUTPUT:-

```
+---------------+
| Next_day      |
+---------------+
| 2024-07-27  |
+--------------- +
```

f. Get database's date.

ANSWER:-

```
SELECT CURRENT_DATE();
```

OUTPUT:-

```
+-----------------------------+
| CURRENT_DATE()     |
+-----------------------------+
| 2024-07-26                  |
+-----------------------------+
```

g. Returns the default(current) database name.

ANSWER:-

```
SELECT DATABASE() AS Database_Name;
```

OUTPUT:-

```
+--------------------+
| DatabaseName |
+--------------------+
| test_db             |
+--------------------+
```

h. Retrieve the current MySQL user name and host name.

ANSWER:-

```
SELECT CURRENT_USER() AS CurrentUser;
```

OUTPUT:-

```
+--------------------+
| CurrentUser     |
+--------------------+
| user_1             |
+--------------------+
```

i. Find the string that tells the MySQL server version.

ANSWER:-

```
 SELECT VERSION() AS Version;
```

OUTPUT:-

```
+---------+
| Version |
+---------+
| 8.0.27  |
+---------+
```

j. Perform Bitwise OR, Bitwise XOR and Bitwise AND.

ANSWER:-

```
SELECT 4 | 2; SELECT 4 ^ 2; SELECT 4 & 2;
```

OUTPUT:-

```
+-------+
| 4 | 2  |
+-------+
|     6   |
+-------+

+-------+
| 4 ^ 2 |
+-------+
|     6   |
+-------+

+-------+
| 4 & 2 |
+-------+
|     0   |
+-------+
```

k. Find the difference between two dates and print in terms of the number of days.
ANSWER:-

```
SELECT DATEDIFF("2024-08-29", "2024-07-25") AS Days;
```

OUTPUT:-

```
+--------+
| Days  |
+--------+
|  35   |
+--------+
```

l. Add one day to the current date.
ANSWER:-

```
SELECT DATE(CURRENT_DATE()+1) AS Tomorrow;
```

OUTPUT:-

```
+----------------+
| Tomorrow    |
+----------------+
| 2024-07-27 |
+----------------+
```

m. Add two hours and 5000 minutes to the current date and print the new date.
ANSWER:-

```
SELECT DATE(CURRENT_TIMESTAMP()+ INTERVAL 2 HOUR + INTERVAL 5000
MINUTE) AS FutureDate;
```

OUTPUT:-

```
+----------------+
| FutureDate  |
+----------------+
| 2024-07-30 |
+----------------+
```

n. Find the floor and ceil values of a floating point number. Also operate on the power, log, modulus, round off and truncate functions.
ANSWER:-

```
SELECT FLOOR(2.5);
SELECT CEIL(2.5);
SELECT POWER(4, 2);
SELECT LOG(8);
SELECT LOG10(100);
SELECT 10 % 4;
SELECT ROUND(2.565, 2);
SELECT TRUNCATE(3.756, 2);
```

OUTPUT:-

```
+----------------+
| FLOOR(2.5) |
+----------------+
|          2          |
+----------------+
+-------------+
| CEIL(2.5) |
+-------------+
|         3      |
+-------------+
+-------------------+
| POWER(4, 2) |
+-------------------+
|          16          |
+-------------------+
+-----------------------------+
| LOG(8)                      |
+-----------------------------+
| 2.0794415416798357 |
+-----------------------------+
+----------------+
| LOG10(100) |
+----------------+
|          2          |
+----------------+
+----------+
| 10 % 4 |
+----------+
|      2      |
+----------+
+-----------------------+
| ROUND(2.565, 2) |
+-----------------------+
|            2.57            |
+-----------------------+
+---------------------------+
| TRUNCATE(3.756, 2) |
+---------------------------+
|              3.75              |
+---------------------------+
```

o. Compare two strings and print the value ‘yes’ if they are equal, else print ‘no’.
ANSWER:-

```
SELECT CASE WHEN 'abc' = 'abc' THEN 'yes' ELSE 'no' END AS Compare;
```

OUTPUT:-

```
+-------------+
| Compare |
+-------------+
| yes          |
+-------------+
```

p. Simulate the “IF... ELSE” construct in MySQL for a mark and grade setup.
ANSWER:-

```
SELECT
    CASE
        WHEN 85 BETWEEN 90 AND 100 THEN 'S'
        WHEN 85 BETWEEN 80 AND 89 THEN 'A'
        WHEN 85 BETWEEN 70 AND 79 THEN 'B'
        WHEN 85 BETWEEN 60 AND 69 THEN 'C'
        WHEN 85 BETWEEN 50 AND 59 THEN 'D'
        ELSE 'F'
    END AS grade
```

OUTPUT:-

```
+--------+
| grade |
+--------+
| A       |
+--------+
```

q. Use IFNULL to check whether a mathematical expression gives a NULL value or not.
ANSWER:-

```
SELECT IFNULL((85/0),"Expression results in NULL") AS Result;
```

OUTPUT:-

```
+-----------------------------------+
| Result                            |
+-----------------------------------+
| Expression results in NULL  |
+-----------------------------------+
```