

DBMS LAB SESSION 1

1. Write SQL queries in MySQL for the following.

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

SQL Query: select year(current_date);

Output:

```
+-----+
| year(current_date) |
+-----+
|                2024 |
+-----+
```

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

SQL Query:

select if(date_format(current_date,'%d-%m-%Y') = current_date, 'Yes', 'No');

Output:

```
+-----+
| if(date_format(current_date,'%d-%m-%Y') = current_date, 'Yes' , 'No') |
+-----+
| No |
+-----+
```

c. Find the size of the SCHEMA/USER.

SQL Query: SELECT SUM(DATA_LENGTH + INDEX_LENGTH) AS size
FROM information_schema.TABLES
WHERE TABLE_SCHEMA = 'mysql';

Output:

```
+-----+
| size |
+-----+
| 2752512 |
+-----+
```

d. Display the current time.

SQL Query: SELECT(CURRENT_TIME);

Output:

```
+-----+
| (current_time) |
+-----+
| 15:15:20 |
+-----+
```

e. Given a date, retrieve the next days date.

SQL Query: SELECT DATE_ADD(current_date,INTERVAL 1 DAY);

Output:

```
+-----+
| DATE_ADD(current_date,INTERVAL 1 DAY) |
+-----+
| 2024-07-26 |
+-----+
```

f. Get database date.

SQL Query: select curdate() as database_date;

Output:

```
+-----+
| database_date |
+-----+
| 2024-07-25 |
+-----+
```

DBMS LAB SESSION 1

g. Returns the default(current) database name.

SQL Query: select database();

Output:

```
+-----+
| database() |
+-----+
| Neel_106122030 |
+-----+
```

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

SQL Query: SELECT USER() AS mysql_user_host;

Output:

```
+-----+
| mysql_user_host |
+-----+
| root@localhost |
+-----+
```

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

SQL Query: SELECT VERSION() AS mysql_server_version;

Output:

```
+-----+
| mysql_server_version |
+-----+
| 8.0.37-0ubuntu0.20.04.3 |
+-----+
```

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

SQL Query: SELECT 2 | 3 AS bitwise_or, 2 ^ 3 AS bitwise_xor, 2 & 3 AS bitwise_and;

Output:

```
+-----+-----+-----+
| bitwise_or | bitwise_xor | bitwise_and |
+-----+-----+-----+
|          3 |          1 |          2 |
+-----+-----+-----+
```

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

SQL Query: SELECT DATEDIFF('2007-12-31 23:59:59', '2007-12-30');

Output:

```
+-----+
| DATEDIFF('2007-12-31 23:59:59', '2007-12-30') |
+-----+
| 1 |
+-----+
```

l. Add one day to the current date.

SQL Query: SELECT DATE_ADD(current_date, INTERVAL 1 DAY);

Output:

```
+-----+
| DATE_ADD(current_date, INTERVAL 1 DAY) |
+-----+
| 2024-07-26 |
+-----+
```

DBMS LAB SESSION 1

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

SQL Query: SELECT DATE_ADD(current_date,INTERVAL '2:5000' HOUR_MINUTE);

Output:

```
+-----+
| DATE_ADD(current_date,INTERVAL '2:5000' HOUR_MINUTE) |
+-----+
| 2024-07-28 13:20:00 |
+-----+
```

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

SQL Query: select floor(5.3), ceil(5.3);

Output:

```
+-----+-----+
| floor(5.3) | ceil(5.3) |
+-----+-----+
|          5 |          6 |
+-----+-----+
```

SQL Query: SELECT POWER(2, 3), LOG10(100);

Output:

```
+-----+-----+
| POWER(2, 3) | LOG10(100) |
+-----+-----+
|          8 |          2 |
+-----+-----+
```

SQL Query: SELECT MOD(10, 3), ROUND(3.14159, 2), TRUNCATE(3.14159, 2);

Output:

```
+-----+-----+-----+
| MOD(10, 3) | ROUND(3.14159, 2) | TRUNCATE(3.14159, 2) |
+-----+-----+-----+
|          1 |          3.14 |          3.14 |
+-----+-----+-----+
```

o. In the first name of the employee, match the following using regular expressions.

SQL Query: SELECT

```
-> CASE
->     WHEN 'Neel' REGEXP '^n' THEN 'Name starts with n'
->     ELSE 'Name does not start with n'
-> END AS result;
```

Output:

```
+-----+
| result |
+-----+
| Name starts with n |
+-----+
```

p. Compare two strings and print the value 'yes' if they are equal, else print 'no'.

SQL Query: SELECT CASE WHEN 'apple' = 'banana' THEN 'yes' ELSE 'no' END AS result;

Output:

```
+-----+
| result |
+-----+
| no |
+-----+
```

DBMS LAB SESSION 1

q. Simulate the construct in MySQL for a mark and grade setup.

SQL Query: SELECT
 -> 85 AS marks,
 -> CASE
 -> WHEN 85 >= 90 AND 85 <= 100 THEN 'A'
 -> WHEN 85 >= 80 AND 85 < 90 THEN 'B'
 -> WHEN 85 >= 70 AND 85 < 80 THEN 'C'
 -> WHEN 85 >= 60 AND 85 < 70 THEN 'D'
 -> WHEN 85 >= 0 AND 85 < 60 THEN 'F'
 -> ELSE 'Invalid marks'
 -> END AS grade;

Output:

marks	grade
85	B

r. Use IFNULL to check whether a mathematical expression gives a NULL value or not

SQL Query: SELECT IFNULL (10 / 5, 'Result is NULL') AS result;

Output:

result
2.0000