NUMERIC FUNCTIONS

In MySQL, you can use various mathematical and rounding functions to perform operations on numeric values. Here are some commonly used mathematical and rounding functions:

ROUND(): The ROUND() function is used to round a numeric value to a specified number of decimal places.

SELECT ROUND(12.345, 2); -- Result: 12.35

TRUNCATE(): The TRUNCATE() function is used to truncate a numeric value to a specified number of decimal places (similar to rounding, but it discards digits).

SELECT TRUNCATE(12.345, 2); -- Result: 12.34

CEILING(): The CEILING() function is used to round a numeric value up to the nearest integer greater than or equal to the value.

SELECT CEILING(12.345); -- Result: 13

FLOOR(): The FLOOR() function is used to round a numeric value down to the nearest integer less than or equal to the value.

SELECT FLOOR(12.345); -- Result: 12

ABS(): The ABS() function is used to return the absolute (positive) value of a numeric expression.

SELECT ABS(-5); -- Result: 5

RAND(): The RAND() function is used to generate a random floating-point number between 0 (inclusive) and 1 (exclusive).

SELECT RAND(); -- Result: A random number between 0 and 1

STRING FUNCTIONS

1. LENGTH
2. UPPER
3. LOWER
4. LTRIM
5. RTRIM
6. TRIM
7. LEFT("KINDER", 4) ---- KIND
8. RIGHT("KINDER", 4) --- NDER
9. SUBSTRING(STRING, START_POSITION, LENGTH)
10.   LOCATE("N", "KINDERGARDEN") ---- 3
11.   CONCAT(STRING, STRING)

DATES FUNCTIONS
1. NOW() --- Current Date and Time
2. CURDATE() --- Current Date
3. CURTIME() ---- Current Time
4. YEAR(NOW()) --- Current Year
5. MONTH(NOW()) --- Current Month
6. DAY(NOW()) --- Current Day
7. HOUR(NOW()) --- Current Hour
8. MINUTE(NOW()) --- Current Minute
9. SECOND(NOW()) --- Current Second
10.   EXTRACT(DAY FROM NOW())
11.   EXTRACT(YEAR FROM NOW())
12.   EXTRACT(MONTH FROM NOW())
13.   EXTRACT(HOUR FROM NOW())
14.   EXTRACT(MINUTE FROM NOW())
15.   EXTRACT(SECOND FROM NOW())

DATE AND TIME FORMAT
1. SELECT DATE_FORMAT(NOW(), "%y") --- 23
2. SELECT DATE_FORMAT(NOW(), "%Y") --- 2023
3. SELECT DATE_FORMAT(NOW(), "%m %Y") --- 09 2023
4. SELECT DATE_FORMAT(NOW(), "%M %Y") --- SEPTEMBER 2023
5. SELECT DATE_FORMAT(NOW(), "%M %d %Y") --- SEPTEMBER 23 2023
6. SELECT TIME_FORMAT(NOW(), "%H:%i %p") --- 09:37 PM

CALCULATING DATES AND TIMES

```sql
SELECT DATE_ADD(NOW(), INTERVAL 1 YEAR); --- 1 YEAR FROM NOW
SELECT DATE_ADD(NOW(), INTERVAL -1 YEAR); --- 1 YEAR BACK FROM NOW
SELECT DATE_SUB(NOW(), INTERVAL -1 YEAR); --- 1 YEAR FROM NOW
SELECT DATE_SUB(NOW(), INTERVAL 1 YEAR); --- 1 YEAR BACK FROM NOW
SELECT DATEDIFF("2019-01-05", "2019-01-01");
SELECT DATEDIFF(LATER_DATE, EARLIER_DATE);
SELECT TIME_TO_SEC(LATER_TIME) - TIME_TO_SEC(EARLIER_TIME);
```

IFNULL
```sql
SELECT ORDER_ID,
      IFNULL(SHIPPER_ID, "NOT ASSIGNED") AS SHIPPER
FROM ORDERS;
```

COALESCE
```sql
SELECT ORDER_ID,
      COALESCE(SHIPPER_ID, COMMENTS, "NOT ASSIGNED") AS SHIPPER
FROM ORDERS;
```

IF
```sql
IF(EXPRESSION, FIRST_VALUE, SECOND_VALUE);

SELECT ORDER_ID, ORDER_DATE,
      IF(YEAR(ORDER_DATE) = YEAR(NOW()), "ACTIVE", "ARCHIVED")
      AS ORDER_STATUS
FROM ORDERS;
```

CASE

```sql
SELECT ORDER_ID, ORDER_DATE,
      CASE
            WHEN YEAR(ORDER_DATE) = YEAR(NOW()) THEN "ACTIVE"
            WHEN YEAR(ORDER_DATE) = YEAR(NOW())-1 THEN "LAST YEAR"
            WHEN YEAR(ORDER_DATE) < YEAR(NOW())-1 THEN "ARCHIVE"
            ELSE "FUTURE"
      END
      AS ORDER_STATUS
FROM ORDE RS;
```

SELECT ORDER_ID, ORDER_DATE,
      CASE
            WHEN YEAR(ORDER_DATE) = YEAR(NOW()) THEN "ACTIVE"
            WHEN YEAR(ORDER_DATE) = YEAR(NOW())-1 THEN "LAST YEAR"