

Numeric Functions:

ABS - Returns the absolute value of a number

Syntax: ABS (numeric_value)

ROUND - Rounds a value to a specified precision

Syntax: ROUND (numeric_value, n)

TRUNCATE - Truncates a value to a specified precision

Syntax: TRUNC (numeric_value, n)

MOD - Returns the remainder of division

Syntax: MOD (m, n)

POWER(x,y) - Calculates x to the power y

STRING FUNCTIONS:

CONCAT - Concatenates data from two different character columns and returns a single column

Syntax: CONCAT (str_val, str_val).

UPPER / LOWER - Returns a string in all capital or all lowercase

letters

Syntax: UPPER(str_val), LOWER(str_val)

SUBSTR - Returns a substring or part
of a given string parameter.
If a given string parameter

Syntax: SUBSTR (str_val, p, l)

LENGTH - Returns a number of characters
in a string value
Syntax: LENGTH (str_val)

LEFT - Returns N number of characters
from the left side of the
string Syntax: LEFT (string, N)

RIGHT - Returns N number of characters
from the right side of
the string Syntax: RIGHT (str, N)

INSTR - Returns the position of the
first occurrence of the
substring in the given
string. Returns 0, if the
substring is not present
in the string
Syntax: INSTR (str, Substr).

Car No	Carmate	Original Police	Gst - amount
1001	Car 1	582613	69913.6
1002	Car 1	673112	80443.4
2001	Car 2	564031	68043.7
2002	Car 2	647852	7743

LTRIM

- Returns the given string after removing leading white space characters
Syntax: LTRIM (string)

RTRIM

- Returns the given string after removing trailing white space characters
Syntax: RTRIM (string)

TRIM

- Returns the given string after removing both leading and trailing white space characters
Syntax: TRIM (string)

- i) Calculate GST at 12% of price and display the result after rounding it off to one decimal place.

Select custid, customer, price as original_price, ROUND(0.12 * price, 1) as gst_amount from inventory;

	ConName	finalprice
2) carid	car1	652527
1001		
1002	car1	153885
1003	car1	153885
1004	car1	153885
1005	car1	153885
1006	car1	153885
1007	car1	153885
1008	car1	153885
1009	car1	153885
1001	car1	725601
1002	car1	725601
1003	car1	725601
1004	car1	725601
1005	car1	725601
1006	car1	725601
1007	car1	725601
1008	car1	725601
1009	car1	725601
3) carid	ConName	Monthly- Installment
1001	car1	6525000
1002	car1	15389000
1003	car1	15389000
1004	car1	15389000
1005	car1	15389000
1006	car1	15389000
1007	car1	15389000
1008	car1	15389000
1009	car1	15389000
4) carid	ConName	Remaining amount
1001	car1	65000
1002	car1	2526.625
1003	car1	2526.625
1004	car1	2526.625
1005	car1	2526.625
1006	car1	2526.625
1007	car1	2526.625
1008	car1	2526.625
1009	car1	2526.625
1001	car1	3888.875
1002	car1	3888.875
1003	car1	3888.875
1004	car1	3888.875
1005	car1	3888.875
1006	car1	3888.875
1007	car1	3888.875
1008	car1	3888.875
1009	car1	3888.875
1001	car1	5074.6875
1002	car1	5074.6875
1003	car1	5074.6875
1004	car1	5074.6875
1005	car1	5074.6875
1006	car1	5074.6875
1007	car1	5074.6875
1008	car1	5074.6875
1009	car1	5074.6875
1001	car1	5601
1002	car1	5601
1003	car1	5601
1004	car1	5601
1005	car1	5601
1006	car1	5601
1007	car1	5601
1008	car1	5601
1009	car1	5601

- 2) Add a new column final price to the table inventory which will have the value as sum of price and 12% of the GST.
- update inventory set finalprice = round(price + (price * 0.12), 1);
- 3) Calculate and display the amount to be paid each month which is to be calculated after dividing the final price of the car into 10 installments.

Select carid, carname, round(finalprice/6) * 1000 as monthly - installment from inventory;

- 4) After dividing the amount into EMI's find out the remaining amount to be paid immediately, by performing modular division.

Select carid, carname, round((finalprice - mod(finalprice, 10000)) / 10, 0) "EMI", mod(finalprice, 1000) "remainingAmount" from inventory;

5) email length email-prefix

amitsaha@gmail.com 19 amitsaha

rehnuma@hotmail.com 19 rehnuma

charnei123@yahoo.com 19 charnei123

b) lower-case-name upper-case-email

amitsaha AMITSAHAS@GMAIL.COM

rehnuma REHNUHA@HOTMAIL.COM

charneimaryyar CHARV1122@YAHOO.COM

c) instname Area-code

charneimaryyar 1169

5) Display the length of the email and part of the email from the email id before the character '@'. Note - do not print '@'.

Select email, length(email) as email_length, substr(email, 1, index(email, '@') - 1) as email_prefix from customer;

6) Display customer name in lower case and customer email in upper case from table customer

Select lower(customername) as custname, upper(email) as upper_email from customer;

7) Let us assume that four-digit area code is reflected in the mobile number starting from position 3. For E.g. 2638 is area code of mobile number 472 6309212. Display the area code of customer living in Rohini.

Select custname, substr(iphone, 3, 4) as area_code from customer where custadd = "10/9, RR, Rohini";

8)	custName	email - without extension
	Amitsaha	amitsaha2@gmail.com
	Rehnuma	rehnuma@hotmail.com
9)	custId	custName email
	10003	chaeninayyar chaenib23@yahoo.com
	10004	Gurpreet gur Singh gur-singh@yahoo.com
10)	money	
	2023-11-09	11:22:29
2)	verdate	
	2023-11-09	

8) Display email after removing the domain name extension ".com" from customer

Select custname, replace (.email,
".com", "") as email without
extension from customer;

9) Display details of all the
customers having yahoo emails
only.

Select * from customer where email
like "%@yahoo.com";

DATE FUNCTIONS:

1) Write a query to retrieve the
current date and time

Select now();

2) Write a query to retrieve
current date.

Select curdate();

3) `(current_time())`

`11: 23 : 23`

4) `day ("2023-11-09")`

`9`

5) `month ("2023-11-09")`

`11`

6) `year ("2023-11-09")`

`2023`

7) `last-day-month`

`2023-11-30`

3) Write a query to retrieve current time.

Select curtime();

4) Write a query to retrieve day of the given date.

Select day ("2023-11-09");

5) Write a query to retrieve month of the given date.

Select month ("2023 - 11-09");

6) Write a query to retrieve year of the given date.

Select year ("2023 - 11-09");

7) Write a query to retrieve last day of the month from the given date.

Select last_day ("2023-11-13") as last_day_of_month;

3) new-date

2023-11-20

9) new-date-with-honesties

2023-11-13 15:00:00

10) entry-date

2023-11-13

11) date-difference

7

20) new-for-standard-within-principle

8) Write a query to determine date leased in the interval days given.

Select date.add("2023-11-13", interval 7 day) as new_date;

9) Write a query to determine the add the number of hour specified.

Select addtime("2023-11-13 15:00:00", "03:00:00") as newdate with hours;

10) Write a query to determine the date from the given date and time.

Select date("2023-11-13 12:00:00") as only_date;

11) Write a query the difference between the given dates.

Select datediff("2023-11-20", "2023-11-13") as date_difference;

(12) day-of-week

2

In example, the value 2 is selected
representing Saturday.

(13) day-of-year

317

An example, the value 317 is selected
representing December 31st.

(14) year-value

2023

(15) original-format vs custom-format

2023-11-19 15:30:00

23-11-19 05:30 AM

12) Write a query to retrieve the day of the week for the given day.

Select `day_of_week("2023-11-13")` as day_of_week;

13) Write a query to retrieve the day of the year for the given day.

Select `day_of_year("2023-11-13")` as day_of_year;

14) Write a query to retrieve the year of the given day.

Select `extract(year from "2023-11-13")` as year value;

15) Write a query to retrieve the date time in different formats

Select `"2023-11-13 15:30:00"` as original_format, `date_format("2023-11-13 15:30:00", "%Y-%m-%d %h:%i %p")` as custom format,