IT Department

Built-in Functions in MySQL

Soma Soleiman Zadeh

Database Systems II (IT 226)

Spring 2024 - 2025

Week 4

March 04, 2025

Outline

- Functions
- **Built-In** Functions vs. **User-Defined** Functions
- Some Built-In Functions in MySQL
  - **MySQL String Functions**
  - MySQL Numeric Functions
  - MySQL Date Functions

Functions

- A **MySQL function** is a **named, reusable sequence of SQL statements**.
- It takes **zero** or **more arguments** as **input**, performs specific operations, and **returns a value**.
- **Functions** can be utilized within **SQL statements**, making them a powerful tool for data manipulation and calculation.

Built-in Functions vs. User-defined Functions

- **Built-in functions** are predefined functions that can be used to perform operations on data at any time.
  - They are usually categorized according to the data types (strings, date, numeric).
- **User-Defined Functions (UDFs)** are one of the most useful features in **MySQL**, allowing users to extend MySQL functionality by creating custom functions.

MySQL String Functions

Function	Description
CHAR_LENGTH()	Returns the length of a string (in characters)
CONCAT( )	Adds two or more expressions (strings) together
CONCAT_WS( )	Adds two or more expressions (strings) together with a separator
POSITION( )	Returns the position of the first occurrence of a substring in a string
REPLACE( )	Replaces all occurrences of a substring within a string, with a new substring
STRCMP( )	Compares two strings
TRIM( )	Removes leading and trailing spaces from a string

CHAR_LENGTH( ) Function

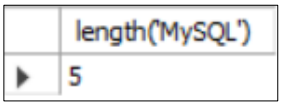
- **CHAR_LENGTH( )** returns the length of a given string in characters (number of characters).

Syntax:

CHAR_LENGTH(***str***)

Example:

SELECT *CHAR_LENGTH ('MySQL')*;



CONCAT( ) Function

- **CONCAT( )** returns the string that results from concatenating the arguments.

Syntax:

CONCAT(***str1, str2, …***)

Example:

```
SELECT  CONCAT( 'My' , 'S' , 'QL' );
```
➡ MySQL

```
SELECT  CONCAT( 'Hello' , NULL , 'ALL');
```
➡ NULL

CONCAT( ) Function

Example: Use **CONCAT( )** function to get the following output from the student table.

student

SID	firstName	lastName
1	Sara	Bahram
2	Noor	Ali
3	Ali	Omar

Output

SID	fullName
1	Sara Bahram
2	Noor Ali
3	Ali Omar

```
SELECT  SID, CONCAT( firstName , ' ' , lastName) AS fullName
FROM  student;
```

CONCAT_WS( ) Function

- **CONCAT_WS( )** stands for **C**oncatenate **W**ith **S**eparator and is a special form of **concat( ).**
- **CONCAT_WS( )** returns the string that results from concatenating the arguments with the separator added between them.

Syntax:

CONTACT_WS(***separator, str1, str2, …***)

Example:

SELECT CONTACT_WS('_', 'Spring' , '2024' , '2025'); → Spring_2024_2025

CONCAT_WS( ) Function

Example: Use **CONCAT_WS( )** function to get the following output from the student table.

student

SID	firstName	lastName
1	Sara	Bahram
2	Noor	Ali
3	Ali	Omar

Output

SID	fullName
1	Sara Bahram
2	Noor Ali
3	Ali Omar

```
SELECT  SID, CONCAT_WS( ' ', firstName , lastName) AS fullName
FROM  student;
```

POSITION( ) Function

- **POSITION( )** returns the position of the first occurrence of a substring in a string.
- If the substring is not found within the original string, this function returns **0**.

Syntax:

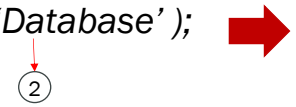
```
POSITION(substr IN str)
```

Example:

```
SELECT POSITION( 'a' IN 'Database' );
```

1 4 6

2



2

REPLACE( ) Function

- **REPLACE( )** replaces all occurrences of a substring within a string, with a new substring.

Syntax:

```
REPLACE(str , old_substr , new_substr)
```

Example:

```
SELECT REPLACE( 'This is the Fall semester.' , 'Fall' , 'Spring' );
```

This is the Spring Semester.

TRIM( ) Function

- **TRIM( )** function removes all spaces from the beginning and end of a string.

Syntax:

TRIM(***str***)

Example:

SELECT TRIM(‘ *This is the Spring semester!* ’);

This is the Spring semester!

STRCMP( ) Function

- **STRCMP( )** compares two strings:
  - If **string1 = string2**, the function returns **0**.
  - If **string1 ≠ string2**, the function returns **1** or **-1** according to the sum of ASCII and the position of the characters.

Syntax:

STRCMP(***str1 , str2***)

STRCMP( ) Function

Example:

```
SELECT STRCMP( 'MySQL' , 'MySQL' );
```

→ 0

```
SELECT STRCMP( 'MySQL' , 'Access' );
```

→ 1

```
SELECT STRCMP( 'Access' , 'MySQL' );
```

→ -1

Some MySQL Numeric Functions

Function	Description
AVG()	Returns the average value of an expression
COUNT( )	Returns the number of records returned by a select query
MAX( )	Returns the maximum value in a set of values
MIN( )	Returns the minimum value in a set of values
SUM( )	Returns the sum of a set of values