

Built-in Functions

Functions and Wildcards in MySQL Server



SoftUni Team
Technical Trainers



SoftUni



Software University

<https://softuni.bg>

sli.do

#java-db

Table of Contents

1. Functions in MySQL
2. String Functions
3. Math Functions
4. Date Functions
5. Wildcards





Functions in MySQL

- **String** Functions – for **manipulating text**, both from table values or user input
 - E.g. concatenate column values
- **Math** Functions – calculations and working with aggregate data
 - E.g. perform geometry and currency operations
- **Date and Time** Functions
 - E.g. find length of timespan
- Other





String Functions

- **SUBSTRING()** – extracts part of a string

SUBSTRING(*String*, *Position*)

SUBSTRING(*String*, *Position*, *Length*)

SUBSTRING(*String* **FROM** *Position* **FOR** *Length*)

SUBSTRING – Example

- Get short summary of article

```
SELECT `article_id`, `author`, `content`,  
       SUBSTRING(`content`, 1, 200) AS 'Summary'  
FROM `articles`;
```


Problem: Find Book Titles

- Write a query to find all book titles that start with "The"
 - Query book_library database

	title
▶	The Mysterious Affair at Styles
	The Big Four
	The Murder at the Vicarage
	The Mystery of the Blue Train
	The Ring
	The Alchemist
	The Fifth Mountain
	The Zahir
	The Dead Zone
	The Hobbit
	The Adventures of Tom Bombadil

Solution: Find Book Titles

```
SELECT title FROM books WHERE  
SUBSTRING(title, 1, 3) = "The";
```



	title
▶	The Mysterious Affair at Styles
	The Big Four
	The Murder at the Vicarage
	The Mystery of the Blue Train
	The Ring
	The Alchemist
	The Fifth Mountain
	The Zahir
	The Dead Zone
	The Hobbit
	The Adventures of Tom Bombadil

- **REPLACE** – replaces specific string with another
 - Performs a case-sensitive match

REPLACE(*String*, *Pattern*, *Replacement*)

String to replace

Field from table

Replacement pattern

REPLACE – Example

- Censor the word **blood** from album names

```
SELECT REPLACE(`title`, 'blood', '*****')  
  AS 'Title'  
FROM `album`;
```

Problem: Replace Titles

- Write a query to find all book titles that start with "The" and replace the substring with "***"
- Query book_library database

	Title
▶	*** Mysterious Affair at Styles
	*** Big Four
	*** Murder at the Vicarage
	*** Mystery of the Blue Train
	*** Ring
	*** Alchemist
	*** Fifth Mountain
	*** Zahir
	*** Dead Zone
	*** Hobbit
	*** Adventures of Tom Bombadil

Solution: Replace Titles

```
SELECT REPLACE(`title`, 'The', '***')  
AS 'Title' FROM `books`  
WHERE SUBSTRING(title, 1, 3) = 'The';
```



	Title
▶	*** Mysterious Affair at Styles
	*** Big Four
	*** Murder at the Vicarage
	*** Mystery of the Blue Train
	*** Ring
	*** Alchemist
	*** Fifth Mountain
	*** Zahir
	*** Dead Zone
	*** Hobbit
	*** Adventures of Tom Bombadil

- **LTRIM** & **RTRIM** – remove **spaces** from either side of string

```
LTRIM(String)
```

```
RTRIM(String)
```

- **CHAR_LENGTH** – count number of characters

```
CHAR_LENGTH(String)
```

- **LENGTH** – get number of used bytes (double for Unicode)

```
LENGTH(String)
```

- **LEFT & RIGHT** – get characters from beginning or end of string

```
LEFT(String, Count)
```

```
RIGHT(String, Count)
```

- Example: name shorthand (first 3 letters)

```
SELECT `id`, `start`,  
      LEFT(`name`, 3) AS 'Shorthand'  
FROM `games`;
```


- **LOWER & UPPER** – change letter casing

```
LOWER(String)
```

```
UPPER(String)
```

- **REVERSE** – reverse order of all characters in string

```
REVERSE(String)
```

- **REPEAT** – repeat string

```
REPEAT(String, Count)
```

- **LOCATE** – locate specific pattern (substring) in string

LOCATE(*Pattern, String, [Position]*)

If omitted, begins at 1

- **INSERT** – insert substring at specific position

INSERT(*String, Position, Length, Substring*)

Number of characters
to delete



Math Functions

Arithmetical Operators and Numeric Functions

- Supported common arithmetic operators

Name	Description
DIV	Integer division
/	Division operator
-	Minus Operator
%, MOD	Modulo operator
+	Addition operator
*	Multiplication operator
- (arg)	Change sign of argument

- Used primarily for numeric **manipulation** and/or mathematical **calculations**
- **PI** – get the value of Pi (15 –digit precision)

```
SELECT PI() +0.0000000000000000
```

- **ABS** – absolute value

```
ABS(Value)
```

- **SQRT** – square root

SQRT(*Value*)

- **POW** – raise value to desired exponent

POW(*Value*, *Exponent*)

- **CONV** – Converts numbers between different number bases

CONV(*Value*, *from_base*, *to_base*)

- **ROUND** – obtain desired precision

ROUND(*Value*, *Precision*)

Can be negative

- **FLOOR** & **CEILING** – return the nearest integer

FLOOR(*Value*)

CEILING(*Value*)

- **SIGN** – returns +1, -1 or 0, depending on value sign

```
SIGN(Value)
```

- **RAND** – get a random value in range [0,1)
 - If **Seed** is not specified, one is assigned at random

```
RAND()
```

```
RAND(Seed)
```




Date Functions

- **EXTRACT** – extract a segment from a date as an integer

EXTRACT(*Part FROM Date*)

- **TIMESTAMPDIFF** – find difference between two dates

TIMESTAMPDIFF(*Part, FirstDate, SecondDate*)

- **Part** can be any part and format of date or time

year, %Y, %y

month, %M, %m

day, %w, %D

YEAR(*Date*)

MONTH(*Date*)

DAY(*Date*)

- Show employee experience

```
SELECT `employee_id`, `first_name`, `last_name`,  
       TIMESTAMPTDIFF(year, `hire_date`, '2017-05-31')  
       AS 'Years In Service'  
FROM `employees`;
```

Problem: Days Lived

- Write a query to calculate how many days have authors lived
 - Use **TIMESTAMPDIFF**
 - Query book_library database

	Full Name	Days Lived
▶	Agatha Christie	31164
	William Shakespeare	18990
	Danielle Schuelein-Steel	NULL
	Joanne Rowling	NULL
	Lev Tolstoy	30021
	Paulo Souza	NULL
	Stephen King	NULL
	John Tolkien	29827
	Erika Mitchell	NULL

Solution: Days Lived

```
SELECT concat(first_name, ' ', last_name) AS 'Full Name',  
TIMESTAMPDIFF(DAY, born, died) AS 'Days Lived'  
FROM authors;
```



	Full Name	Days Lived
▶	Agatha Christie	31164
	William Shakespeare	18990
	Danielle Schuelein-Steel	NULL
	Joanne Rowling	NULL
	Lev Tolstoy	30021
	Paulo Souza	NULL
	Stephen King	NULL
	John Tolkien	29827
	Erika Mitchell	NULL

- **DATE_FORMAT** – formats the date value according to the format

```
SELECT DATE_FORMAT('2017/05/31', '%Y %b %D') AS 'Date';
```

- **NOW** – obtain current date and time

```
SELECT NOW();
```



WILDCARD

Wildcards

Selecting Results by Partial Match

Wildcards

- Used to substitute any other character(s) in a string
 - '%' - represents zero, one, or multiple characters
 - '_' - represents a single character
 - Can be used in combinations
- Used with **LIKE** operator in a **WHERE** clause
 - Similar to **Regular Expressions**



Wildcards – Examples

- Find any values that start with "a"

```
WHERE customer_name LIKE 'a%';
```

- Find any values that have "r" in second position

```
WHERE customer_name LIKE '_r%';
```

- Finds any values that starts with "a" and ends with "o"

```
WHERE customer_name LIKE 'a%o';
```

- Supported characters also include:
 - `\` – specify prefix to treat special characters as normal
 - `[charlist]` – specifying which characters to look for
 - `[!charlist]` – **excluding** characters

```
SELECT * FROM `customers`  
WHERE `city` REGEXP '[a-c]%' ;
```

"a", "b", or "c"

Problem: Harry Potter Books

- Write a query to retrieve information about the titles of all Harry Potter books
 - Use **Wildcards**
 - Query book_library database

	title
▶	Harry Potter and the Philosopher's S...
	Harry Potter and the Chamber of Se...
	Harry Potter and the Prisoner of Azk...
	Harry Potter and the Goblet of Fire
	Harry Potter and the Order of the P...
	Harry Potter and the Half-Blood Prince
	Harry Potter and the Deathly Hallows
	Harry Potter and the Deathly Hallows

Solution: Harry Potter Books

```
SELECT title FROM books  
WHERE title LIKE 'Harry Potter%';
```



	title
▶	Harry Potter and the Philosopher's Stone
	Harry Potter and the Chamber of Secrets
	Harry Potter and the Prisoner of Azkaban
	Harry Potter and the Goblet of Fire
	Harry Potter and the Order of the Phoenix
	Harry Potter and the Half-Blood Prince
	Harry Potter and the Deathly Hallows
	Harry Potter and the Deathly Hallows

- **REGEXP** - pattern matching using regular expressions

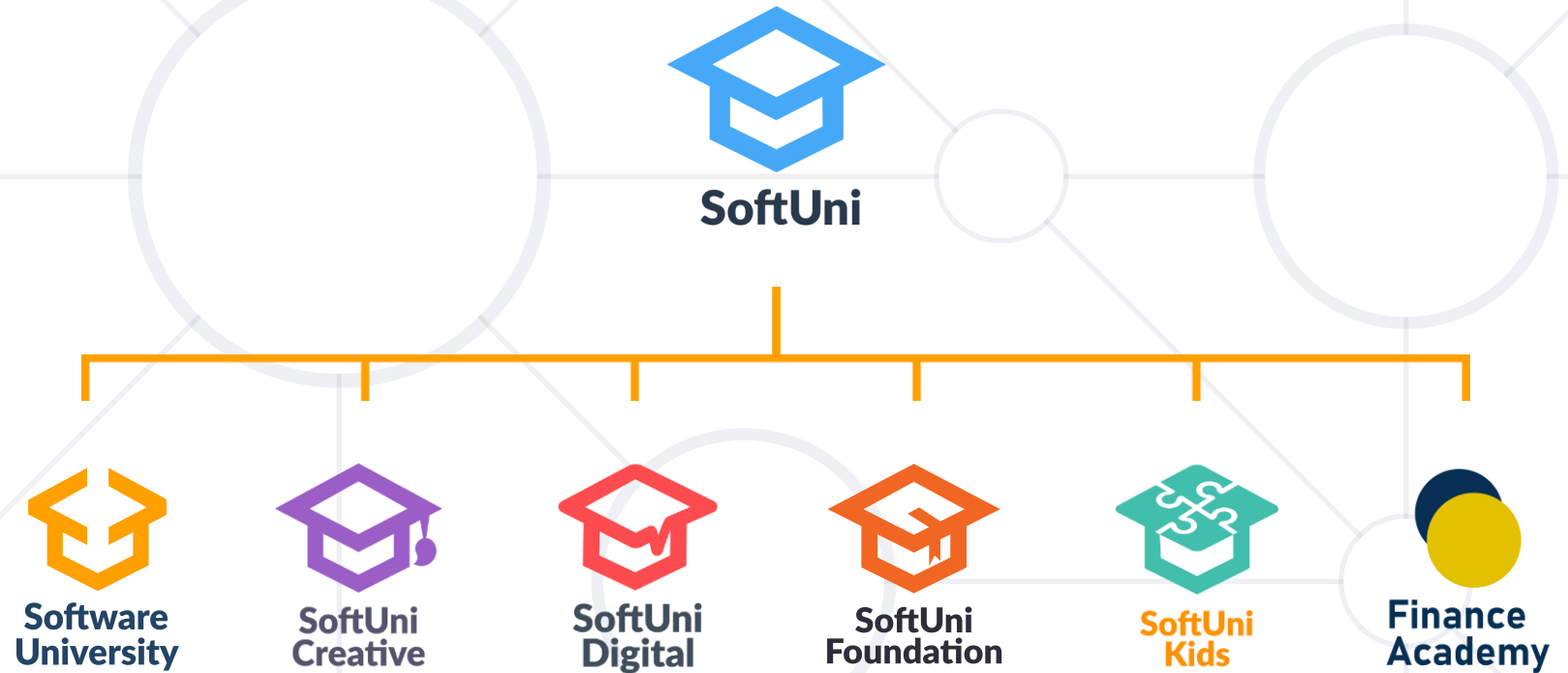
```
SELECT `employee_id`, `first_name`, `last_name`  
FROM `employees`  
WHERE `first_name` REGEXP '^\\[^K\\]{3}\\$';
```

Regular expression

- MySQL Server provides various built-in functions
 - **Numerical** functions
 - **String** functions
- Using Wildcards, we can obtain results by **partial string matches**
 - Regular expressions



Questions?



SoftUni Diamond Partners



- Software University – High-Quality Education, Profession and Job for Software Developers

- softuni.bg, about.softuni.bg

- Software University Foundation

- softuni.foundation

- Software University @ Facebook

- facebook.com/SoftwareUniversity

- Software University Forums

- forum.softuni.bg



- This course (slides, examples, demos, exercises, homework, documents, videos and other assets) is **copyrighted content**
- Unauthorized copy, reproduction or use is illegal
- © SoftUni – <https://about.softuni.bg/>
- © Software University – <https://softuni.bg>

