CHAR_LENGTH(str)

Returns the length of the string str, measured in characters. A multi-byte character counts as a single character. This means that for a string containing five two-byte characters, LENGTH() returns 10, whereas CHAR_LENGTH() returns 5.

```
mysql> SELECT CHAR_LENGTH("text");

+---------------------------------------------------------+

| CHAR_LENGTH("text")                                     |

+---------------------------------------------------------+

| 4                                                       |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

CONCAT(str1,str2,...)

Returns the string that results from concatenating the arguments. May have one or more arguments. If all arguments are non-binary strings, the result is a non-binary string. If the arguments include any binary strings, the result is a binary string. A numeric argument is converted to its equivalent binary string form; if you want to avoid that, you can use an explicit type cast, as in this example:

```
mysql> SELECT CONCAT('My', 'S', 'QL');

+---------------------------------------------------------+

| CONCAT('My', 'S', 'QL')                                 |

+---------------------------------------------------------+

| MySQL                                                   |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

INSERT(str,pos,len,newstr)

Returns the string str, with the substring beginning at position pos and len characters long replaced by the string newstr. Returns the original string if pos is not within the length of the string. Replaces the rest of the string from position pos if len is not within the length of the rest of the string. Returns NULL if any argument is NULL.

```
mysql> SELECT INSERT('Quadratic', 3, 4, 'What');

+---------------------------------------------------------+

| INSERT('Quadratic', 3, 4, 'What')                       |

+---------------------------------------------------------+

| QuWhattic                                               |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

LCASE(str)

LCASE() is a synonym for LOWER().

LEFT(str,len)

Returns the leftmost len characters from the string str, or NULL if any argument is NULL.

```
mysql> SELECT LEFT('foobarbar', 5);

+---------------------------------------------------------+

| LEFT('foobarbar', 5)                                    |

+---------------------------------------------------------+

| fooba                                                   |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

LENGTH(str)

Returns the length of the string str, measured in bytes. A multi-byte character counts as multiple bytes. This means that for a string containing five two-byte characters, LENGTH() returns 10, whereas CHAR_LENGTH() returns 5.

```
mysql> SELECT LENGTH('text');

+---------------------------------------------------------+

| LENGTH('text')                                          |

+---------------------------------------------------------+
| 4                                                          |
+-----------------------------------------------------------+
1 row in set (0.00 sec)
```

LOWER(str)

Returns the string str with all characters changed to lowercase according to the current character set mapping.

```
mysql> SELECT LOWER('QUADRATICALLY');
+-----------------------------------------------------------+
| LOWER('QUADRATICALLY')                                     |
+-----------------------------------------------------------+
| quadratically                                              |
+-----------------------------------------------------------+
1 row in set (0.00 sec)
```

LPAD(str,len,padstr)

Returns the string str, left-padded with the string padstr to a length of len characters. If str is longer than len, the return value is shortened to len characters.

```
mysql> SELECT LPAD('hi',4,'??');
+-----------------------------------------------------------+
| LPAD('hi',4,'??')                                          |
+-----------------------------------------------------------+
| ??hi                                                       |
+-----------------------------------------------------------+
1 row in set (0.00 sec)
```

LTRIM(str)

Returns the string str with leading space characters removed.

```
mysql> SELECT LTRIM('  barbar');
+-----------------------------------------------------------+
| LTRIM('  barbar')                                          |
+----------------------------------------------------------+
| barbar                                                   |
+----------------------------------------------------------+
1 row in set (0.00 sec)
```

MID(str,pos,len)

MID(str,pos,len) is a synonym for SUBSTRING(str,pos,len).

REPEAT(str,count)

Returns a string consisting of the string str repeated count times. If count is less than 1, returns an empty string. Returns NULL if str or count are NULL.

```
mysql> SELECT REPEAT('MySQL', 3);
+----------------------------------------------------------+
| REPEAT('MySQL', 3)                                       |
+----------------------------------------------------------+
| MySQLMySQLMySQL                                          |
+----------------------------------------------------------+
1 row in set (0.00 sec)
```

REPLACE(str,from_str,to_str)

Returns the string str with all occurrences of the string from_str replaced by the string to_str. REPLACE() performs a case-sensitive match when searching for from_str.

```
mysql> SELECT REPLACE('www.mysql.com', 'w', 'Ww');
+----------------------------------------------------------+
| REPLACE('www.mysql.com', 'w', 'Ww')                      |
+----------------------------------------------------------+
| WwWwWw.mysql.com                                         |
+----------------------------------------------------------+
1 row in set (0.00 sec)
```

REVERSE(str)

Returns the string str with the order of the characters reversed.

```
mysql> SELECT REVERSE('abcd');

+---------------------------------------------------------+

| REVERSE('abcd')                                         |

+---------------------------------------------------------+

| dcba                                                    |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

RIGHT(str,len)

Returns the rightmost len characters from the string str, or NULL if any argument is NULL.

```
mysql> SELECT RIGHT('foobarbar', 4);

+---------------------------------------------------------+

| RIGHT('foobarbar', 4)                                   |

+---------------------------------------------------------+

| rbar                                                    |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

RPAD(str,len,padstr)

Returns the string str, right-padded with the string padstr to a length of len characters. If str is longer than len, the return value is shortened to len characters.

```
mysql> SELECT RPAD('hi',5,'?');

+---------------------------------------------------------+

| RPAD('hi',5,'?')                                        |

+---------------------------------------------------------+

| hi???                                                   |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

RTRIM(str)

Returns the string str with trailing space characters removed.

```
mysql> SELECT RTRIM('barbar   ');

+---------------------------------------------------------+

| RTRIM('barbar   ')                                      |

+---------------------------------------------------------+

| barbar                                                  |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

STRCMP(str1, str2)

Compares two strings and returns 0 if both strings are equal, it returns -1 if the first argument is smaller than the second according to the current sort order otherwise it returns 1.

```
mysql> SELECT STRCMP('MOHD', 'MOHD');

+---------------------------------------------------------+

| STRCMP('MOHD', 'MOHD')                                  |

+---------------------------------------------------------+

| 0                                                       |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

Another example is:

```
mysql> SELECT STRCMP('AMOHD', 'MOHD');

+---------------------------------------------------------+

| STRCMP('AMOHD', 'MOHD')                                 |

+---------------------------------------------------------+

| -1                                                      |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

Let's see one more example:

```
mysql> SELECT STRCMP('MOHD', 'AMOHD');

+---------------------------------------------------------+

| STRCMP('MOHD', 'AMOHD')                                 |

+---------------------------------------------------------+

| 1                                                       |

+---------------------------------------------------------+

1 row in set (0.00 sec)
```

SUBSTRING(str,pos) SUBSTRING(str FROM pos) SUBSTRING(str,pos,len) SUBSTRING(str FROM pos FOR len)

The forms without a len argument return a substring from string str starting at position pos. The forms with a len argument return a substring len characters long from string str, starting at position pos. The forms that use FROM are standard SQL syntax. It is also possible to use a negative value for pos. In this case, the beginning of the substring is pos characters from the end of the string, rather than the beginning. A negative value may be used for pos in any of the forms of this function.

```
mysql> SELECT SUBSTRING('Quadratically',5);

+---------------------------------------------------------+

| SSUBSTRING('Quadratically',5)                           |

+---------------------------------------------------------+

| ratically                                               |

+---------------------------------------------------------+

1 row in set (0.00 sec)


mysql> SELECT SUBSTRING('foobarbar' FROM 4);

+---------------------------------------------------------+

| SUBSTRING('foobarbar' FROM 4)                           |

+---------------------------------------------------------+

| barbar                                                  |
+---------------------------------------------------------+
1 row in set (0.00 sec)


mysql> SELECT SUBSTRING('Quadratically',5,6);
+---------------------------------------------------------+
| SUBSTRING('Quadratically',5,6)                          |
+---------------------------------------------------------+
| ratica                                                  |
+---------------------------------------------------------+
1 row in set (0.00 sec)
```

SUBSTRING_INDEX(str,delim,count)

Returns the substring from string str before count occurrences of the delimiter delim. If count is positive, everything to the left of the final delimiter (counting from the left) is returned. If count is negative, everything to the right of the final delimiter (counting from the right) is returned. SUBSTRING_INDEX() performs a case-sensitive match when searching for delim.

```
mysql> SELECT SUBSTRING_INDEX('www.mysql.com', '.', 2);
+---------------------------------------------------------+
| SUBSTRING_INDEX('www.mysql.com', '.', 2)                |
+---------------------------------------------------------+
| www.mysql                                               |
+---------------------------------------------------------+
1 row in set (0.00 sec)
```

TRIM([{BOTH | LEADING | TRAILING} [remstr] FROM] str)

TRIM([remstr FROM] str)

Returns the string str with all remstr prefixes or suffixes removed. If none of the specifiers BOTH, LEADING, or TRAILING is given, BOTH is assumed. remstr is optional and, if not specified, spaces are removed.

```
mysql> SELECT TRIM('  bar   ');
+---------------------------------------------------------+
| TRIM('  bar   ')                                        |
+---------------------------------------------------------+
| bar                                                     |
+---------------------------------------------------------+
1 row in set (0.00 sec)

mysql> SELECT TRIM(LEADING 'x' FROM 'xxxbarxxx');
+---------------------------------------------------------+
| TRIM(LEADING 'x' FROM 'xxxbarxxx')                      |
+---------------------------------------------------------+
| barxxx                                                  |
+---------------------------------------------------------+
1 row in set (0.00 sec)

mysql> SELECT TRIM(BOTH 'x' FROM 'xxxbarxxx');
+---------------------------------------------------------+
| TRIM(BOTH 'x' FROM 'xxxbarxxx')                         |
+---------------------------------------------------------+
| bar                                                     |
+---------------------------------------------------------+
1 row in set (0.00 sec)

mysql> SELECT TRIM(TRAILING 'xyz' FROM 'barxxyz');
+---------------------------------------------------------+
| TRIM(TRAILING 'xyz' FROM 'barxxyz')                     |
+---------------------------------------------------------+
| barx                                                    |
+---------------------------------------------------------+
1 row in set (0.00 sec)
```

UCASE(str)

UCASE() is a synonym for UPPER().

UPPER(str)

Returns the string str with all characters changed to uppercase according to the current character set mapping.

```
mysql> SELECT UPPER('Mysql’);

+---------------------------------------------------------+
```