MySQL | Regular expressions (Regexp)

shreyanshi_arun

Read   Discuss   Courses   Practice

MySQL supports another type of pattern matching operation based on the regular expressions and the REGEXP operator.

1. It provide a powerful and flexible pattern match that can help us implement power search utilities for our database systems.
2. REGEXP is the operator used when performing regular expression pattern matches. RLIKE is the synonym.
3. It also supports a number of metacharacters which allow more flexibility and control when performing pattern matching.
4. The backslash is used as an escape character. It's only considered in the pattern match if double backslashes have used.
5. Not case sensitive.

Pattern	What the Pattern matches
*	Zero or more instances of string preceding it
+	One or more instances of strings preceding it

Engineering Mathematics   Discrete Mathematics   Digital Logic and Design   Computer Organization and Architecture

?	Match zero or one instances of the strings preceding it.
^	caret(^) matches Beginning of string
$	End of string
[abc]	Any character listed between the square brackets
[^abc]	Any character not listed between the square brackets
[A-Z]	match any upper case letter.

▲

Pattern	What the Pattern matches
[a-z]	match any lower case letter
[0-9]	match any digit from 0 through to 9.
[[:<:]]	matches the beginning of words.
[[:>:]]	matches the end of words.
[:class:]	matches a character class i.e. [:alpha:] to match letters, [:space:] to match white space, [:punct:] is match punctuations and [:upper:] for upper class letters.
p1|p2|p3	Alternation; matches any of the patterns p1, p2, or p3
{n}	n instances of preceding element
{m,n}	m through n instances of preceding element

Examples with explanation :

- **Match beginning of string(^):** Gives all the names starting with 'sa'.Example- sam,samarth.

```
SELECT name FROM student_tbl WHERE name REGEXP '^sa';
```

- **Match the end of a string($):** Gives all the names ending with 'on'.Example – norton,merton.

```
SELECT name FROM student_tbl WHERE name REGEXP 'on$';
```

- **Match zero or one instance of the strings preceding it(?):** Gives all the titles containing 'com'.Example – comedy , romantic comedy.

```
SELECT title FROM movies_tbl WHERE title REGEXP 'com?';
```

- **matches any of the patterns p1, p2, or p3(p1|p2|p3):** Gives all the names containing 'be' or 'ae'.Example – Abel, Baer.

```
SELECT name FROM student_tbl WHERE name REGEXP 'be|ae' ;
```

- **Matches any character listed between the square brackets([abc]):** Gives all the names containing 'j' or 'z'.Example – Lorentz, Rajs.

```
SELECT name FROM student_tbl WHERE name REGEXP '[jz]' ;
```

- **Matches any lower case letter between 'a' to 'z'- ([a-z]) ([a-z] and (.)):** Retrieve all names that contain a letter in the range of 'b' and 'g', followed by any character, followed by the letter 'a'.Example – Tobias, sewall. Matches any single character(.)

```
SELECT name FROM student_tbl WHERE name REGEXP '[b-g].[a]' ;
```

- **Matches any character not listed between the square brackets.([^abc]):** Gives all the names not containing 'j' or 'z'. Example – nerton, sewall.

```
SELECT name FROM student_tbl WHERE name REGEXP '[^jz]' ;
```

- **Matches the end of words[[:>:]]:** Gives all the titles ending with character "ack". Example – Black.

```
SELECT title FROM movies_tbl WHERE REGEXP 'ack[[:>:]]';
```

- **Matches the beginning of words[[:<:]]:** Gives all the titles starting with character "for". Example – Forgetting Sarah Marshal.

```
SELECT title FROM movies_tbl WHERE title REGEXP '[[:<:]]for';
```

- **Matches a character class[:class:]:** i.e [:lower:]- lowercase character ,[:digit:] – digit characters etc. Gives all the titles containing alphabetic character only. Example – stranger things, Avengers.

```
SELECT title FROM movies_tbl WHERE REGEXP '[:alpha:]' ;
```

- **Matches the beginning of all words by any character listed between the square brackets. (^[abc]):** Gives all the names starting with 'n' or 's'. Example – nerton, sewall.

```
SELECT name FROM student_tbl WHERE name REGEXP '^[ns]' ;
```

Last Updated : 05 Sep, 2022   116

Similar Reads

1.   MySQL | Recursive CTE (Common Table Expressions)

2.   MySQL | Common MySQL Queries