Regular Expression (Metacharacter)

→ Metacharacters are the building blocks of regular expressions. Characters in Regular expression are understood to be either a metacharacter with a special meaning or a regular character with a literal

meaning.

Following are some common metacharacters in regular expressions.

CHARACTER	Explanation	Example
.	This metacharacter allows any and exactly one character.	matches any character
+	This character allows one or more preceding term.	/ah+x/ matches "ahx" or "ahhhhx"
*	This character allows zero or more preceding term.	/ah*x/ matches "aax", "ahx" or "ahhhhx"
?	This character allows zero or on preceding term.	/ah?x/ matches "aahx" and "ahx" but not "ahhx"
\.	This character allows a period (.) in the text.	/etc\./ matches "etc."

→ \/	The character allows forward slash in the text.	/.+\/.+/ matches "HOME/MY"
→ *	This character allows asterik in the text.	/My*NAME/ matches "My*NAME"
→ \+	This character allows Plus sign in the text.	/.+\+.+/ matches "FIVE+FOUR"
→ \?	This character allows question mark in the text.	/.+\?/ matches "REALLY?"
→ ^	The string should begin with the specified word written after this character.	/^BYE/ matches "BYE" in "BYE HARRY"
→ $	The string should end with the specified word written before this character.	/HARRY$/ matches "HARRY" in "AND HARRY"

Examples of Metacharacters:

1) Do a global search for "m.d" in a string:

```
LET REG = /m.d/g;
LET STR = "HE'S MAD!" // matches.
```

2) Do a global search for at least one "e":

```
LET REG = /0+/g;
LET STR = "CODEEEE!"; // matches.
```

3) Do a global search for a "5", followed by zero or more "0" characters:

```
LET REG = /50*/g;
LET STR = "5, 500 or 5000?"; // matches.
```

4) Do a global search for "Javascript" at the end of a string:

```
LET REG = /JAVASCRIPT$/g;
LET STR = "TUTORIAL OF JAVASCRIPT"; // matches.
```

5) Do a global search for "JAVASCRIPT" at the begging of a string:

```
LET REG = /^JAVASCRIPT/g;
LET STR = "JAVASCRIPT SUPPORTS OOP"; // matches
```

6) Do a global search for "CODE" followed by "HARRY":

```
LET REG = /CODE(?=HARRY)/g;
LET STR = "CODE WITH HARRY"; // matches
```

Page No. _____
Date | | | |

7) Do a global, case ~~insetive~~ insensitive search for "code" not followed by "JAVASCRIPT":

```
LET REG = /CODE (?! JAVASCRIPT/gi ;
LET STR = "CODE JAVASCRIPT" ; // does not match
```