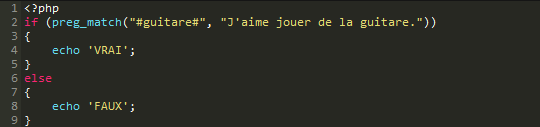
RegEx are used to do high level searches inside strings ; it’s like a buffed up version of find/replace in word. They are used in MYSQL frequently (select \* WHERE name = %a%). Here are a few examples of use:

* verifying that the user input is a valid email address
* changing a date in American format to a rest-of-the-world format
* changing automatically all string that starts with a http:// with a clickable link
* the BBcode (<http://www.lovelyreality.com/html-vs-bbcode/>)

Regexes of the PCRE type stem from Perl. All expressions of this type are prefixed with **preg\_**.

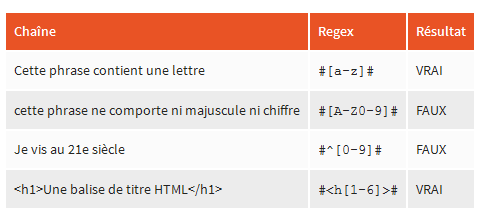


Maybe you wanna know, for some reason, whether a string starts with the word “Hello” or not, or whether another string ends with a .com. You use ^ or $

* ^ (accent circonflexe) : indique le début d'une chaîne ;
* $ (dollar) : indique la fin d'une chaîne.

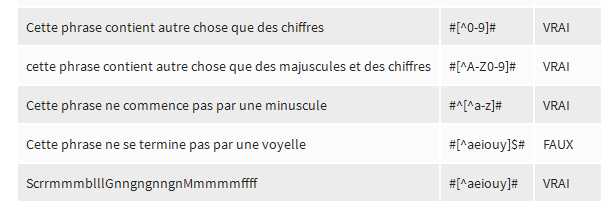






**Important: searching whether a string doesn’t have a particular character**

you use ^ to specify you don’t want a character in your string. This works only if the ^ is inside the brackets



#^[^0-9]# --> here you wanna make sure the string does **not** begin with an integer.

**Quantifiers**

There are three main quantifiers:

* ? returns true if the char is present 0 or 1 times
* + returns present if the char is present 1 or more times
* \* returns true if the char.. well whatever the case is.

For example, you are not sure how to spell Marry (some Marrys are spelled with two Rs, other with one). So you will use :

#[M-m]arr?y#~

Another example, if you wanna check whether the user input is a laugh, but you don’t know how many “ha”s his hahaha will have:

#ha (ha)\*# (at least one, for the sassy “ha”, and maybe a million ha’s, if the guy’s a lunatic).

This one demands that the string starts and finishes with yahoo(with an unspecified number of “o”)



This one wants to know whether the beginning of the string is www.

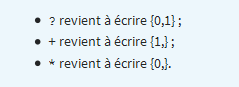
#^(w{3}).#

{3,5} is the interval [3,5]

{3,} is minimum 3, max infinite.

This one checks whether it starts with https or ftp, followed by www. and followed by alphanumerical values (to check whether it is a valid address)

#(((https?|ftp)://(w{3}\.)?)(?<!www)(\w+-?)\*\.([a-z]{2,4}))#



This one checks whether it is a string of exactly 6 integers:

#^[0-9]{6}^#

**Metacharacters**

# Delimite regex

!

^ “At the beginning of the stack” OR, if inside brackets “Is not in”

$ “At the end of the stack”

() “creates a variable”

[] “range/type of characters ([0-9]) / creates a class”

{} “used for .times method”

? “optional (‘Is in stack 0 or 1 times’)”

+ “is in stack 1 or more times”

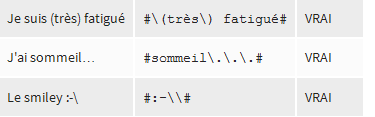
\* “Whatever”

. “”

\ “same as in php, breaks special characters (\? = interrogation point)”

| “this value or that value”

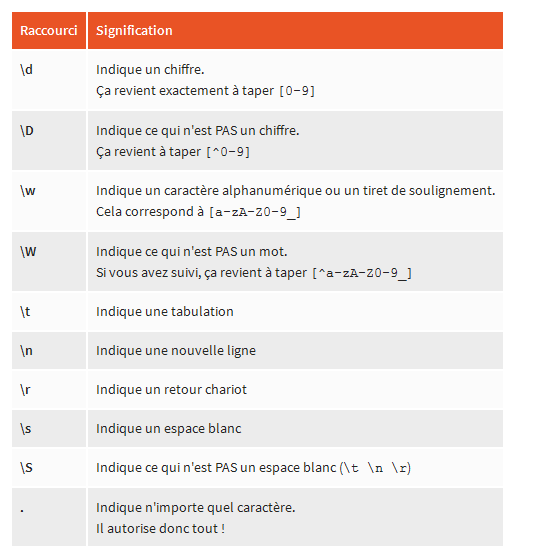
eg:

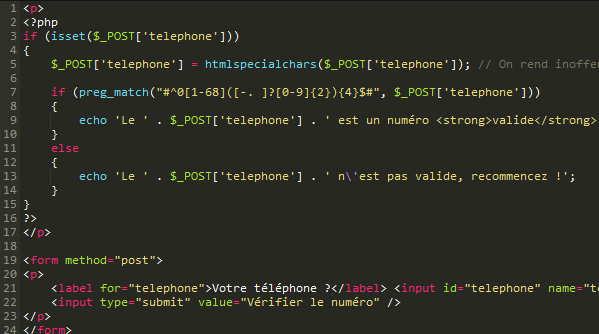


http://i.gyazo.com/7b9916c9854958a4673f71ca64dc4873.pngis ok, because it is inside a “class” of characters. However, if you were to put # or ] in your class, you need

**Abridged classes**

Some classes you don’t need to write them (English wooohoo!)





Here it is in action. Note that the digits of a phone number can be separated by either a space, a ., a -, or nothing, so we use the class [-. ]. Here is another example for verifying whether it is a valid email address.

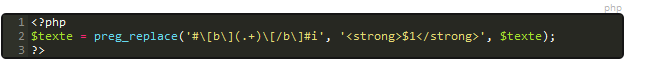
#^[a-z0-9-\_.]+@[a-z0-9-\_]{2,}\.[a-z]{2,4}$#

So that was Regex. Also available in Mysql

http://i.gyazo.com/581cde238909dc2d440ec78f45054406.png

**Search and Replace**

Imagine you are building a forum, and you wanna change a url to a link everytime a message has one. So you gotta replace some characters with other. That’s where preg\_replace comes in handy. The way it works is, you create a variable using the paranthesises; (.+) means some characters, at least one, and then the replace part of the regex will target the chars inside the parenthesises. Here we replace text that is in bold (bbcode). The parenthesises are stored in PHP variables ($1,$2,$3…) which index is determined by the order of the parenthesises. $0 is created by default, it relates to the whole regex:



That’s actually how bbcode is made. The miracle of life…

The I after the reg ex makes it so it also works if the b’s are in capital letters.

