1. <https://regex101.com/> (this site can be used to test our regular epression)
2. Cheatsheet : https://www.rexegg.com/regex-quickstart.html
3. Best example : <https://mkyong.com/regular-expressions/10-java-regular-expression-examples-you-should-know/>

^ : it is for starting of line or sentence

^Ram - > sentence start from Ram

But - > [^Ram] - > it scape the Ram from the sentence , hat used inside a set of bracket **[^...]** for excluding characters

$ : - > dollar sign used to represent the last word of sentence

Eg. ^Mission:Successful$ - > meaning sentence start from Mission and end with Successful

1. Best to understand : <https://jkorpela.fi/perl/regexp.html>

phone number regex:

650-555-2345

(416)555-3456

202 555 4567

4035555678

Regex : 1?[\s-]?\(?(\d{3})\)?[\s-]?\d{3}[\s-]?\d{4}

1? [\s-]? \(? (\d{3}) \)? [\s-]? \d{3} [\s-]? \d{4}

Explain:

1? is zero times or one times

[\s-]? - white spce or dashes in zero times or one times

\(? - open prentthesis , escaping and difining zero or one times

(\d{3}) - > 3 digits must be

\)? - > escaping right open parentthesis which might occur zero or one time

[\s-]? -> white space or dashes , one or zero occurrences

\d{3} -> digit 3

[\s-]? -> white space or dashes , one or zero occurrences

\d{4} -> digit 4

Email verification:

^[A-Za-z0-9.\_%+-][+@[A-Za-z]+\.[A-Za-z]{3}](mailto:+@[A-Za-z]+\.%5bA-Za-z%5d%7b3%7d)

Shambhu456@gmail.com

^ [A-Za-z0-9.\_%+-][+ @ [A-Za-z]+ \.[A-Za-z]{3}](mailto:+@[A-Za-z]+\.%5bA-Za-z%5d%7b3%7d)

Explanation:

^ [A-Za-z0-9.\_%+-][+](mailto:+@[A-Za-z]+\.%5bA-Za-z%5d%7b3%7d)  - > should begin with provided character and one or more occurrence of given character : Shambhu456

@ - > should have @ sign

[A-Za-z]+ -> after @ should have given character one or more in occurrence

Gmail

\.[A-Za-z]{3} - > special meta character dot . is escaping by backward slash and given character shood occur 3 times

.com

**Grouping and capturing :**

**Grouping : if we put any tegex inside the parenthesis then it becomes the group**

1. **Whole regex is known as group 0**
2. **And all the group created by ( ) parenthesis - > group 1 , 2 and so on**

**$1 and $2 so on can use to retriev the group value**

**Backslash reference : ?**