**REGEX METACHARACTERS**

**.**     the period is a wildcard for any character (letter, number, symbol, etc.).  
**[]**    square brackets are used in character classes.   
**()**    parenthesis are used in capturing groups.   
**{}**    curly brackets are used in quantifiers.   
**\**     the backslash is used to precede a metacharacter or a *predefined character class*. It is also used to indicate a Java escape sequence character.  
**^**     the caret is used for negation or the beginning of a line.  
**-**     the dash is used for ranges.  
**&**     the ampersand is used for intersections.  
**=**     the equals is used for special construct lookahead.  
**$**     the dollar sign is a boundary matcher for the end of a line.  
**!**     the exclamation mark is used for special construct lookbehind.  
**|**     the pipe is used for **or** expressions.  
**?**     the question mark is used in quantifiers and special constructs.  
**\***     the asterisk is used in quantifiers.  
**+**     the plus is used in quantifiers.  
**,**     the comma is used in quantifiers.

**Using some examples**

|  |  |
| --- | --- |
| **package** com.Soham;  **import** java.util.regex.\*;  **public class** Main {   **public static void** main(String[] args) {   *//finds all bcz . is an wildcard character,  // this can be anything  displayFind*(**"."**,**"107.1.53.254"**);  *displayFind*(**"\\."**,**"107.1.53.254"**);  *displayFind*(**"\\Q.\\E"**,**"107.1.53.254"**);  System.***out***.println(**"----------------------------------"**);   *//since . is an wildcard character thats the reason why,  // it will pick any combination of 3 characters  displayFind*(**"..."**,**"Blah blah ... blah blah"**);  *displayFind*(**"\\.\\.\\."**,**"Blah blah ... blah blah"**);  *displayFind*(**"\\Q...\\E"**,**"Blah blah ... blah blah"**);  System.***out***.println(**"----------------------------------"**);   System.***out***.println(**"Searching for backslashes..."**);  System.***out***.println(**"\\\\java\\\\"**);  System.***out***.println(**"c:\\java\\"**);  *displayFind*(**"\\\\java\\\\"**,**"c:\\java\\"**);  System.***out***.println(**"----------------------------------"**);   *//in this case we are searching [41], means, a 4 and a 1,  // not 41, it searches either 4 or either 1  displayFind*(**"[41]"**,**"new int[14]"**);  *//searching string [41]  displayFind*(**"\\Q[41]\\E"**,**"new int[14]"**);  *displayFind*(**"\\Q[41]\\E"**,**"new int[41]"**);  *displayFind*(**"\\[41\\]"**,**"new int[41]"**);  }   **public static void** displayFind(String regex, String find\_me){  **boolean** getmatch = **false**;  Pattern p = Pattern.*compile*(regex);  Matcher m = p.matcher(find\_me);   **while** (m.find()){  System.***out***.println(**"Matcher found "** + m.group() + **" at index "** + m.start() + **" for regex "** + regex + **" with "** + find\_me);  getmatch = **true**;  }   **if** (!getmatch){  System.***out***.printf(**"no match for "** + regex + **" on "** + find\_me);  }  System.***out***.println();  } } | **Matcher found 1 at index 0 for regex . with 107.1.53.254**  **Matcher found 0 at index 1 for regex . with 107.1.53.254**  **Matcher found 7 at index 2 for regex . with 107.1.53.254**  **Matcher found . at index 3 for regex . with 107.1.53.254**  **Matcher found 1 at index 4 for regex . with 107.1.53.254**  **Matcher found . at index 5 for regex . with 107.1.53.254**  **Matcher found 5 at index 6 for regex . with 107.1.53.254**  **Matcher found 3 at index 7 for regex . with 107.1.53.254**  **Matcher found . at index 8 for regex . with 107.1.53.254**  **Matcher found 2 at index 9 for regex . with 107.1.53.254**  **Matcher found 5 at index 10 for regex . with 107.1.53.254**  **Matcher found 4 at index 11 for regex . with 107.1.53.254**  **Matcher found . at index 3 for regex \. with 107.1.53.254**  **Matcher found . at index 5 for regex \. with 107.1.53.254**  **Matcher found . at index 8 for regex \. with 107.1.53.254**  **Matcher found . at index 3 for regex \Q.\E with 107.1.53.254**  **Matcher found . at index 5 for regex \Q.\E with 107.1.53.254**  **Matcher found . at index 8 for regex \Q.\E with 107.1.53.254**  **----------------------------------**  **Matcher found Bla at index 0 for regex ... with Blah blah ... blah blah**  **Matcher found h b at index 3 for regex ... with Blah blah ... blah blah**  **Matcher found lah at index 6 for regex ... with Blah blah ... blah blah**  **Matcher found .. at index 9 for regex ... with Blah blah ... blah blah**  **Matcher found . b at index 12 for regex ... with Blah blah ... blah blah**  **Matcher found lah at index 15 for regex ... with Blah blah ... blah blah**  **Matcher found bl at index 18 for regex ... with Blah blah ... blah blah**  **Matcher found ... at index 10 for regex \.\.\. with Blah blah ... blah blah**  **Matcher found ... at index 10 for regex \Q...\E with Blah blah ... blah blah**  **----------------------------------**  **Searching for backslashes...**  **\\java\\**  **c:\java\**  **Matcher found \java\ at index 2 for regex \\java\\ with c:\java\**  **----------------------------------**  **Matcher found 1 at index 8 for regex [41] with new int[14]**  **Matcher found 4 at index 9 for regex [41] with new int[14]**  **no match for \Q[41]\E on new int[14]**  **Matcher found [41] at index 7 for regex \Q[41]\E with new int[41]**  **Matcher found [41] at index 7 for regex \[41\] with new int[41]** |