
Lab 09

JavaScript Form Validation

Objective:

This lab will introduce you to regular expressions (REGEX) and its practical implementation by demonstrating the use of regular expressions in different views.

Activity Outcomes:

After this lab the students should be able to understand Java Script basics and to validate the form using Java Script.

This lab teaches you the following topics:

- Purpose and use of different regular expressions
- To write functions in html, validate using regular expressions
- Use of REGEX for validating different inputs (email, password etc).
- Use of REGEX for checking certain words in a string/comment.

Instructor Note:

As pre-lab activity, read Chapter 9 from the textbook “Web Programming with HTML5, CSS, and JavaScript Pap/Psc John Dean, Year 2018.

1) Useful Concepts

Introduction

A regular expression is an object that describes a pattern of characters. Regular expressions are nothing more than a sequence or pattern of characters itself. They provide the foundation for pattern-matching functionality. They are used to perform pattern-matching and "search- and-replace" functions on text.

Following link is helpful in creating regular expressions of your requirements.
https://developer.mozilla.org/en-US/docs/Web/JavaScript/Guide/Regular_Expressions

2) Solved Lab Activities

Activity 1:

Develop and demonstrate, using Javascript script, HTML document that collects the user input (the valid format is: A digit from 1 to 4 followed by two upper-case characters followed by two digits followed by two upper-case characters followed by three digits; no embedded spaces allowed) of the user. Event handler must be included for the form element that collects this information to validate the input. Messages in the alert windows must be produced when errors are detected.

Solution:

```
1  <!DOCTYPE html>
2  <html>
3  <head> <title> Input Validation </title> </head>
4  <body>
5  <form>
6  Note: A digit from 1 to 4, Two upper-case characters, Two digits, Two upper-case characters, Three digits.<br/>
7  User Input: <input type="text" name="user" /> <br/>
8  <input type="button" value="Validate" onclick="valid
9  (user)" />
10 </form>
11
12
13 <script type="text/javascript">
14 function valid(user)
15 {
16 var pattern1=/^[1-4][A-Z]{2}[0-9]{2}[A-Z]{2}[0-9]{3}$/
17 if(!user.value.match(pattern1)||user.value.length==0)
18 {
19 alert("Invalid User Input!\nEnter a valid input")
20 return false
21 }
22 else alert("Input is valid!")
23 }
24 </script>
25
26 </body>
27 </html>
28
```

Activity 2:

Using REGEX to perform character recognition in a string. Also use global(g) case sensitive(i) and multiline(m) search.

Solution:

```
1  <!DOCTYPE html>
2  <html>
3  <body>
4
5  <p>Click the button to do a global search for characters NOT inside the brackets [h] in a string.</p>
6
7  <button onclick="myFunction()">Try it</button>
8
9  <p id="demo"></p>
10
11 <script>
12 function myFunction() {
13     var str = "Is this all there is?";
14     var patt1 = /^[^h]/g;
15     var result = str.match(patt1);
16     document.getElementById("demo").innerHTML = result;
17 }
18 </script>
19
20 </body>
21 </html>
```

Activity 3:

Validate email field while creating a signup form. Also show the error messages if email is not valid. Use event handler for input field to validate email.

Solution:

```
1  <script src="https://code.jquery.com/jquery-3.1.1.min.js" integrity="sha256-hVVnYaiADRTO2PzUGmULJr8BLUSjGIZsDYGmIJLv2b8="
2  crossorigin="anonymous"></script>
3
4  <form method="post">
5      <label for="email" id="email-ariaLabel">Your email address:</label>
6      <input id="email" name="email" type="email" class="required"/>
7      <span id="error">Not a valid email address</span>
8      <span id="success">A valid email address!</span>
9  </form>
10
11 <script>
12 $(document).ready(function(){
13     $('#success').hide();
14     $('#error').hide();
15     $('#email').keyup(function () {
16         var email = $('#email').val();
17         // i Perform case-insensitive matching
18         // g Perform a global match (find all matches rather than stopping after the first match)
19         // m Perform multiline matching
20         var re = /^[A-Z0-9._%+-]+@[A-Z0-9.-]+\.[A-Z]{2,4}$/igm;
21         if (re.test(email)) {
22             $('#success').show();
23             $('#error').hide();
24         } else {
25             $('#error').show();
26             $('#success').hide();
27         }
28     });
29 </script>
```

3) Graded Lab Tasks

Lab Task 1

Validate password field while creating signup form. Ask the user to set a strong password by using at least 1 uppercase letter, 1 lowercase letter, 1 digit and 1 ASCII character show the error messages if password doesn't contain any of these expressions.