Experiment Details

Subject: ECOM

Exp No: 03

Experiment Name: Design forms and perform form validation

Resources Required

a. Equipments/Machines: P IV 810 MHz, 20 GB HDD, 128 MB RAM, 1.44 FDD

14" Color Monitor, 101 keyboard, serial mouse, LAN Card

Dot Matrix Printer, Internet Explorer

b. Consumables: Printer Pages *for* printouts

Procedure /Algorithm

Theory :

Any sort of interactive site is going to have form inputs — a place where your users input who they are, what they want to buy, where they live, and so forth. This data is passed to whatever handles back end — a Perl CGI script, a PHP engine, a database like Oracle, or some other technology . Whatever system is back there, it doesn’t appreciate having its time wasted with bogus information, and chances are the user doesn’t appreciate it either. Client-side validation is instantaneous because it doesn’t have to transmit any data. JavaScript catches any erroneous data the user enters before it goes anywhere.

Double-checking the data on the server remains necessary, in case the user has turned JavaScript off or somehow finds a way to circumvent the client-side validation, either maliciously or by accident. For the majority of your users, JavaScript form validation will save a lot of time up front.

This script accompanies an HTML form. When the user clicks the Submit button on the form, the form data is sent to a JavaScript validation function that checks each aspect of the input to make sure that it is in the appropriate format. Each form element is evaluated according to specified criteria. If the script finds an error in one of the fields, it sends back a warning explaining how the string doesn’t conform. The fairly robust string-handling and regular-expression techniques available in JavaScript handle this checking process.

Program

body

{ background-image: url(om.jpg);

background-repeat:no-repeat;

background-size:cover;

}

.login-page

{

width: 360px;

padding:10% 0 0;

margin: auto ;

}

.form

{

position: relative;

z-index: 1;

background: red;

max-width: 360;

margin: 0 auto 100px;

padding: 45px;

}

.form input

{

font-family:"Roboto", sans-serif;

outline:1;

background: #f2f2f2;

width:100%;

border:0;

margin:0 0 15px;

padding:15px;

box-sizing: border-box;

font-size: 14px;

}

.form button

{

font-family:"Roboto", sans-serif;

text-transform: uppercase;

outline: 0;

background: #4CAF50;

border:0;

padding:15px;

color:green;

cursor:Pointer;

}

.form .register-form

{

dispaly: none;

}

<html>

<head>

<title>Login and Registration</title>

<link rel="stylesheet" href="style.css">

</head>

<body>

<div class="login-page">

<div class="form">

<form class="register-form">

<input type="text" placeholder="username" size=65 name="Name"/><br/><br/>

<input type="text" placeholder="password" size=65 name="Password"/><br/><br/>

<input type="text" placeholder="email id" type="text" size=65 name="EMail"/>

<button>Create</button>

<p class="message">Already Registered? <a href="#">LOGIN </a>

</p>

</form>

<form class="login-form">

<input type="text" placeholder="username"><br>

<input type="text" placeholder="password"><br>

<button>login</button>

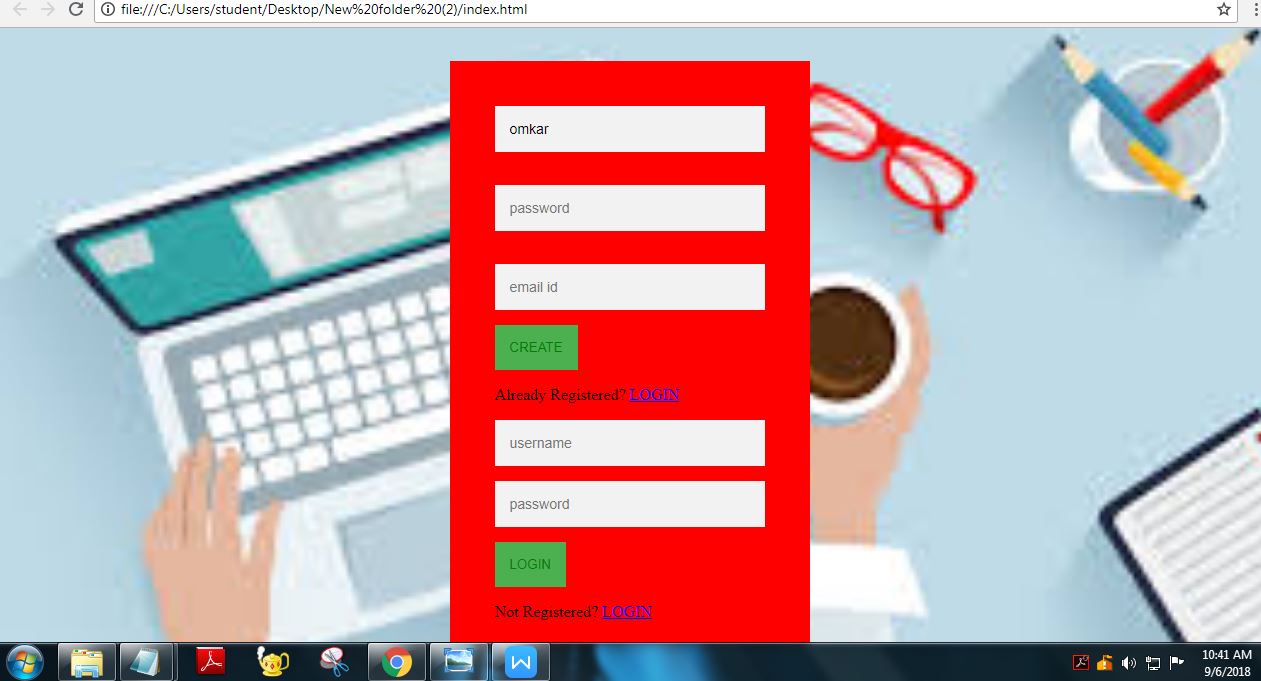
<p class="message">Not Registered? <a href="#">LOGIN </a>

</div>

</body>

</html>

**Standard Observations *I* Output /Results**:



Conclusion:Hence we have sucessfully completed the experiment .