## JavaScript Form Validation:

It is important to validate the form submitted by the user because it can have inappropriate values. So, validation is must to authenticate user.

JavaScript provides facility to validate the form on the client-side so data processing will be faster than server-side validation. Most of the web developers prefer JavaScript form validation.

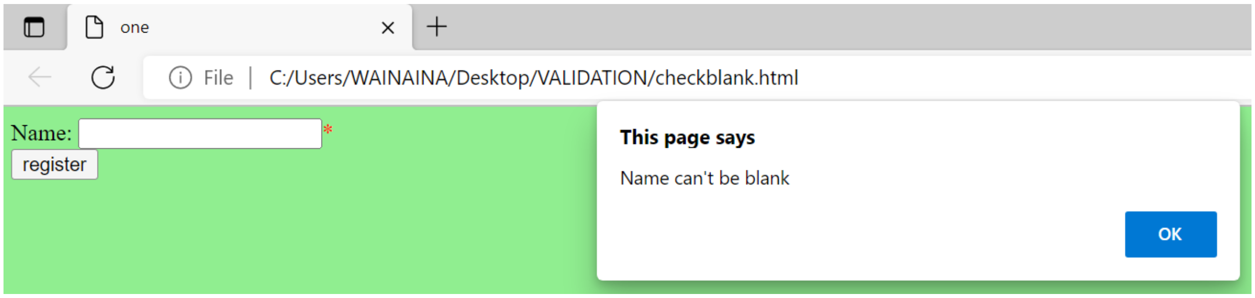
Through JavaScript, we can validate name, password, email, date, mobile numbers and more fields.

### Example one: A JavaScript program to check if a filed is blank.

```
<html>
<head>
<title>one</title>
<script>
function validateform()
{
var me=document.WAINAINA.me.value;

if (me==null || me=="")
{
  alert("Name can't be blank");
  return false;
}
}
</script>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return validateform()" >
Name: <input type="text" name="me"><span style="color:#ff0000">*</span><br/>
<input type="submit" value="register">
</form>
</body>
</html>
```

### OUTPUT:



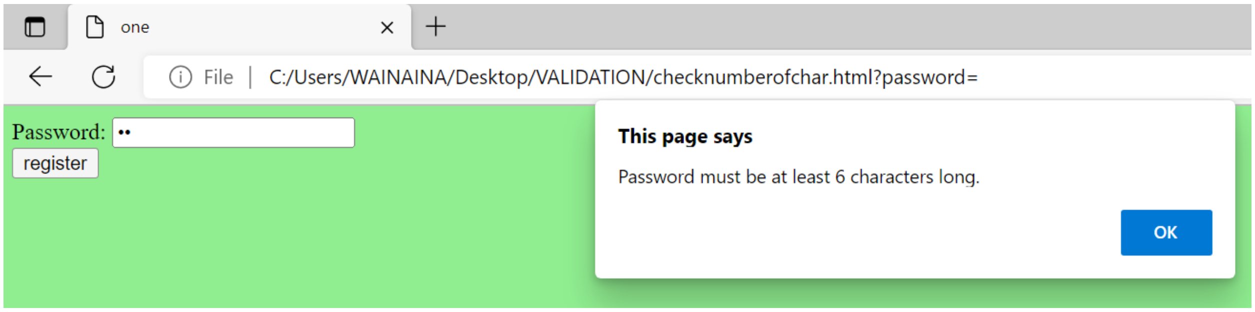
This page says

Name can't be blank

## Example two: A JavaScript program to check number of characters entered in a field.

```
<html>
<head>
<title>one</title>
<script>
function validateform()
{
var password=document.WAINAINA.password.value;
if(password.length<6)
{
  alert("Password must be at least 6 characters long.");
  return false;
  }
}
</script>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return validateform()" >
Password: <input type="password" name="password"><br/>
<input type="submit" value="register">
</form>
</body>
</html>
```

## OUTPUT


one
File
C:/Users/WAINAINA/Desktop/VALIDATION/checknumberofchar.html?password=
register
OK

Password: ••

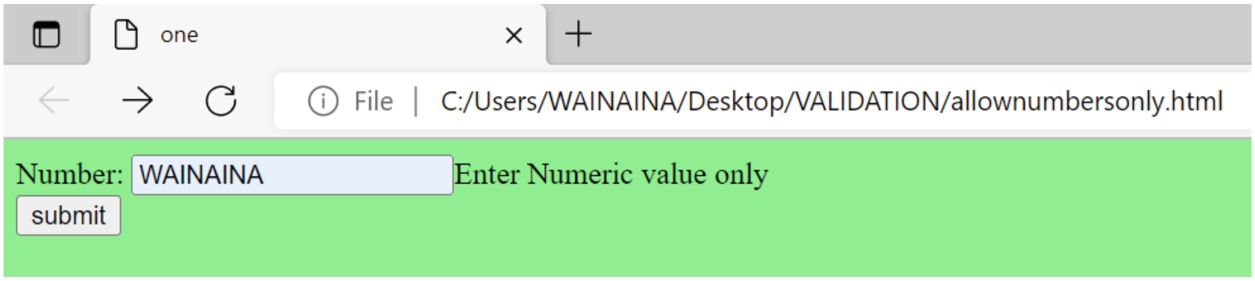
This page says

Password must be at least 6 characters long.

## Example three: A JavaScript program to allow only Numbers to be entered in a field.

```
<html>
<head>
<title>one</title>
<script>
function validate()
{
var num=document.WAINAINA.num.value;
if (isNaN(num)) //short for "Not-a-Number. Returns true if a value is NaN.
{
 document.getElementById("GEO").innerHTML="Enter Numeric value only";
 return false;
}
else
{
 return true;
 }
}
</script>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return validate()" >
Number: <input type="text" name="num"><span id="GEO"></span><br/>
<input type="submit" value="submit">
</form>
</body>
</html>
```

**OUTPUT:**

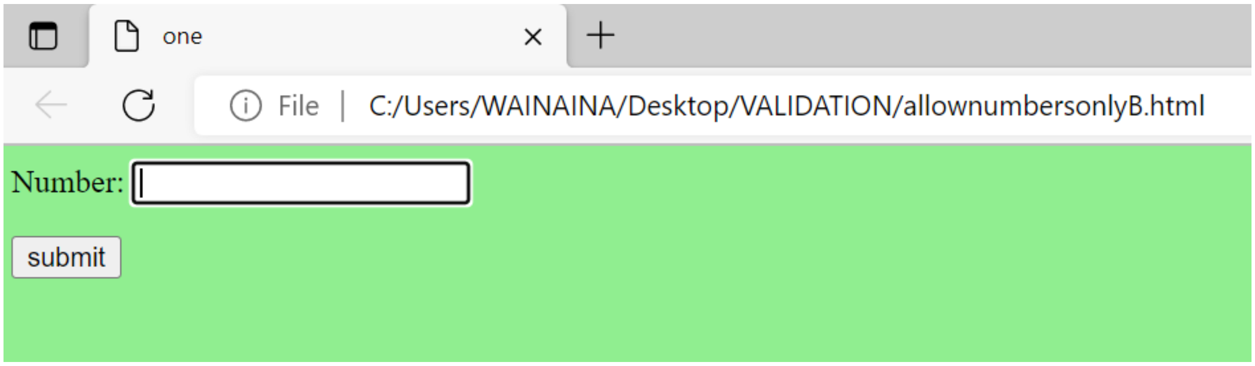


Number: WAINAINA Enter Numeric value only

**Alternatively, you can validate numeric values as follows:**

```
<html>
<head>
<title>one</title>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return validate()" >
Number: <input type="text"
onkeypress="return(event.charCode>=48)&&(event.charCode<=57)">
<p><input type="submit" value="submit">
</form>
</body>
</html>
```

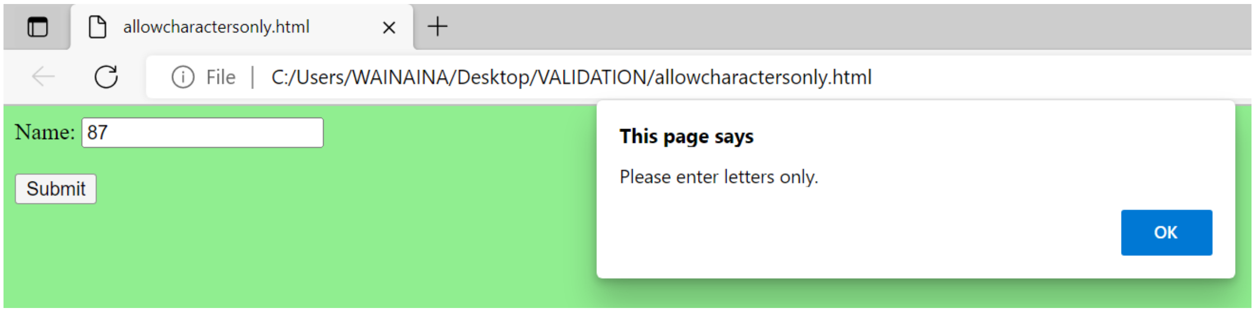
**OUTPUT:**



**Example Four: A JavaScript program to allow only characters to be entered in a field.**

```
<!DOCTYPE html>
<html lang="en">
<head>
<script>
function validate(name)
{
  var check = /^[A-Za-z]+$/;
  if(name.value.match(check))
    {
     return true;
    }
  else
    {
    alert("Please enter letters only.");
    return false;
    }
}
</script>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA">
Name: <input type='text' name='name'/></br></br>
<input type="submit" name="submit" value="Submit"
onclick="validate(document.WAINAINA.name)"/>
</form>
</body>
</html>
```

**OUTPUT:**


allowcharactersonly.html
File
C:/Users/WAINAINA/Desktop/VALIDATION/allowcharactersonly.html
Name: 87
Submit
OK

This page says

Please enter letters only.

**Alternatively, you can validate characters as follows:**

```
<html>
<head>
<title>one</title>
</head>
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return validate()" >
Name: <input type="text" onkeypress="return(event.charCode>=65)&&(event.charCode<=91) ||
(event.charCode>=97)&&(event.charCode<=123)">
<p><input type="submit" value="submit">
</form>
</body>
</html>
```

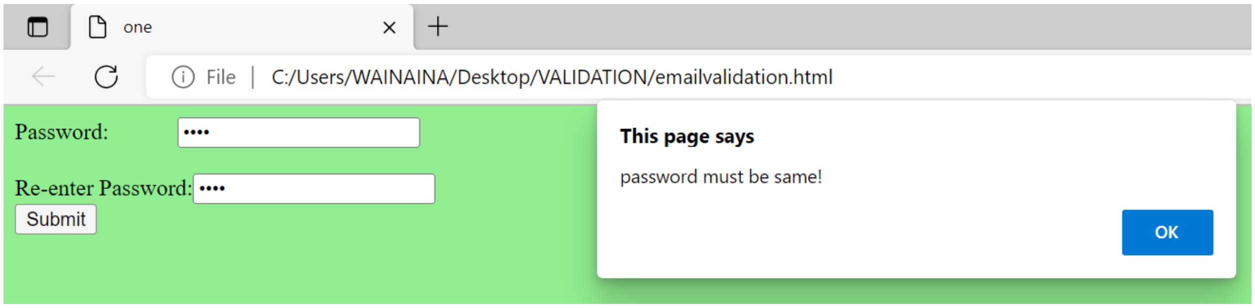
**Example Five: A JavaScript program to validate the password entered is the same.**

```
<html>
<head>
<title>one</title>
<script type="text/javascript">
function matchpass()
{
var firstpassword=document.WAINAINA.password.value;
var secondpassword=document.WAINAINA.password2.value;

if(firstpassword==secondpassword)
{
return true;
}
else
{
alert("password must be same!");
return false;
}
}
</script>
</head>
```

```
<body bgcolor="lightgreen">
<form name="WAINAINA" onsubmit="return matchpass()">
Password:&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;<input type="password" name="password" /><br/></br>
Re-enter Password:<input type="password" name="password2"/><br/>
<input type="submit">
</form>
</body>
</html>
```

**OUTPUT:**



This page says

password must be same!

## Example Six: A JavaScript program to Validate Email.

We can validate the email by the help of JavaScript.

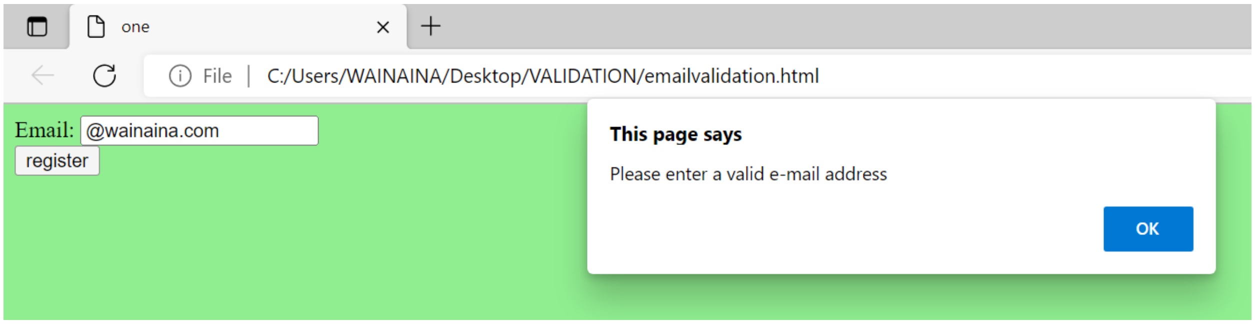
There are many criteria that need to be follow to validate the email id such as:

- email id must contain the @ and . character
- There must be at least one character before and after the @.
- There must be at least two characters after . (dot).

```
<html>
<head>
<title>one</title>
<script>
function validateemail()
{
var x=document.myform.email.value;
var atposition=x.indexOf("@"); //method returns the first index at which a given element can be found in the array, or -1 if it is not present.
var dotposition=x.lastIndexOf("."); //returns the index (position) of the last occurrence of a specified value in a string.
if (atposition<1 || dotposition<atposition+2 || dotposition+2>=x.length)
{
  alert("Please enter a valid e-mail address");
  return false;
  }
}
</script>
<body bgcolor="lightgreen">
<form name="myform"  method="post" action="#" onsubmit="return validateemail();">
Email: <input type="text" name="email"><br/>

<input type="submit" value="register">
</form>
</body>
</html>
```

**OUTPUT:**



C:/Users/WAINAINA/Desktop/VALIDATION/emailvalidation.html

Email: @wainaina.com

This page says

Please enter a valid e-mail address