

# JavaScript

## JavaScript Events



# JavaScript Events

---

- Every element on a web page has certain events which can trigger invocation of event handlers
- Attributes are inserted into HTML tags to define events and event handlers
- Basic idea: the web page waits for the user to do something (generate an event) and respond
- Examples of events:
  - Clicking on a button
  - Loading a web page or an image.
  - Moussing over a spot on the web page
  - Submitting an HTML form

# onload Events

---

- The `onload` and `onUnload` events are triggered when the user enters or leaves the page
- The `onload` event is often used to check the visitor's browser type and browser version, and load the proper version of the web page based on the information
- Both the `onload` and `onUnload` events are also often used to deal with `cookies` that should be set when a user enters or leaves a page.
- `onload` is an event that occurs the page loaded.

```
<body onload="window.alert('Page successfully loaded');">
```

# Example: onload event

---

```
<!DOCTYPE html>
<html>
<head>
<script>
    function myFunction() {
        alert("Page is loaded");
    }
</script>
</head>

<body onload="myFunction()">
    <h2>Hello World!</h2>
</body>

</html>
```

# onclickEvent

---

- **onclick Event Type**
- This is the most frequently used event type which occurs when a user clicks the left button of his mouse. You can put your validation, warning, etc.,

```
<input type="button" onclick="some js..." />
```

# Example: onclick event

---

```
<body>
  <button onclick="displayDate()">The time is</button>
  <p id="p1"></p>

  <script>
function displayDate() {
    var today= new Date();
    document.getElementById("p1").innerHTML = today;}
  </script>

</body>
```

# Example: onclick event

---

```
<html>
<head>
<script >
function ChangeText() {
    document.getElementById("p1").style.color="blue";
    document.getElementById("p1").style.fontSize="30px";
    document.getElementById("p1").style.fontFamily="Arial";
}
</script>
</head>
<body>
<p id="p1">Hello world!</p>
<input type="button" onclick="ChangeText()"  value="Click me
to change text above">
</body>
</html>
```

# onSubmit Event

---

- The onSubmit event is used to validate all form fields before submitting it.
- Example: The **checkForm()** function will be called when the user clicks the submit button in the form. If the field values are not accepted, the submit should be canceled. The function **checkForm()** returns either **true** or **false**. If it returns true the form will be submitted, otherwise the submit will be cancelled:

```
<form method="post" action="xxx.html"  
      onsubmit="return checkForm()">
```

# Example: onSubmit Event

```
<head>  
  <script>  
    function CheckAndSubmit() {  
      // Validate username  
      var userName = document.getElementById("userName");  
      if (userName.value.length < 3) {  
        alert("The name of the user must be at least 3 characters long!");  
        return false; }  
  
      // Validate password  
      var password = document.getElementById("password");  
      var repassword = document.getElementById("repassword");  
  
      if (password.value.length < 6) {  
        alert("The password must be at least 6 characters long!");  
        return false; }
```

# Example: onSubmit Event

```
if (repassword.value != password.value) {  
    alert("The two passwords are different!");  
    return false; }  
// Validate agreement checkbox  
var acceptAgreement =  
document.getElementById("acceptAgreement");  
if (!acceptAgreement.checked) {  
    alert("You must accept the User Agreement to register!");  
    return false; }  
// If all validations pass  
alert('Form Submitted Successfully!');  
return true; }  
</script>  
</head>
```

# Example: onSubmit Event

```
<body>  
  <form id="regForm" method="post" action="#URL#"  
        onsubmit="return CheckAndSubmit ()">  
    User Name: <input type="text" name="userName" id="userName" />  
    <br />  
    Password: <input type="password" name="password"  
              id="password" /> <br />  
    Repeat Password: <input type="password" name="repassword"  
                      id="repassword" /> <br /><br />  
    <input type="checkbox" name="acceptAgreement"  
          id="acceptAgreement" />  
    <label for="acceptAgreement">I accept the User Agreement </label>  
    <br /><br />  
    <input type="submit" value="Register" />  
  </form> </body>
```

# onMouseOver and onMouseOut Events

---

- **onMouseOver** and **onMouseOut** are often used to create "animated" buttons.
- Example: An alert box appears when an onMouseOver event is detected:

```
<button  
    onMouseOver="alert('Button hovered!')"  
    onMouseOut="alert('Mouse left the button!')"  
    >  
    Hover Me  
</button>
```

```
<a href="#"  
    onMouseOver="alert('An onMouseOver event'); ">  
      
</a>
```

# onMouseOver and onMouseOut Events

---

- Practical Example (Changing Button Style on Hover)

```
<button  
    onMouseOver="this.style.backgroundColor='#ff9900';  
    this.style.transform='scale(1.5)';"  
    onMouseOut="this.style.backgroundColor=''; this.style.transform=' ';"  
>  
    Hover Me  
</button>
```

# onBlur, onFocus, and onChange Events

- The onBlur, onFocus, and onChange events are often used in combination with validation of form fields.

Event	Trigger When	Best For
onFocus	User clicks/tabs into the field	Styling (e.g., highlight input)
onChange	User modifies the field's content	Real-time validation (e.g., live feedback)
onBlur	User leaves the field (clicks/tabs away)	Final validation (e.g., before submission)

- Example: The checkEmail() function will be called whenever the user changes the content of the field:

```
<input type="text" size="30" id="email"  
      onChange="checkEmail()">
```

# Example: onChange

---

## Live Password Strength Checker

*html*

```
<input type="password" id="password"
placeholder="Enter password (min 6 chars)"
onChange="checkPasswordStrength(this)" >
<p id="passwordStrength"></p>
```

# Example: onChange

## javaScript

```
function checkPasswordStrength(input) {  
    const password = input.value;  
    const strengthElement =  
document.getElementById("passwordStrength");  
    if (password.length === 0) {  
        strengthElement.textContent = "";  
    } else if (password.length < 6) {  
        strengthElement.textContent = "✗ Too weak (min 6  
            characters);  
        strengthElement.style.color = "red";  
    } else {  
        strengthElement.textContent = "✓ Strong enough!";  
        strengthElement.style.color = "green";  
    }  
}
```

# Example: onblur

---

When the user leave the input field, a function is triggered which transforms the input text to upper case

```
<html>
<head>
<script>
    function upperCase() {
        let x=document.getElementById("fname").value
        document.getElementById("fname").value=x.toUpperCase()
    }
</script>
</head>
<body>
    Enter your name:
    <input type="text" id="fname" onblur="upperCase()">
</body></html>
```

# Example: onFocus

---

When the password field is focused, a hint appears below it.

```
<html> <head>
<script>
    function showHint(message) {
        document.getElementById("hint").textContent = message;
    }
</script></head>
<body>
    <input type="password" id="password"
        onFocus="showHint('Password must be 6+ characters.'"
    >
    <p id="hint" style="font-size: 12px; color: #666;"></p>
</body></html>
```