

Experiment No. 06

Title: Event handling using JavaScript to explore web browser environment.

Batch:B2**Roll No.:1914078****Experiment No.:6****Aim:** Event handling using JavaScript to explore web browser environment.**Resources needed: Notepad++, Web Browser****Theory:**

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

An HTML web page has finished loading

An HTML input field was changed

An HTML button was clicked

Often, when events happen, you may want to do something.

JavaScript lets you execute code when events are detected.

HTML allows event handler attributes, with JavaScript code, to be added to HTML elements.

What can JavaScript Do?

Event handlers can be used to handle, and verify, user input, user actions, and browser actions:

- Things that should be done every time a page loads
- Things that should be done when the page is closed
- Action that should be performed when a user clicks a button □ Content that should be verified when a user inputs data □ And more ...

Many different methods can be used to let JavaScript work with events:

- HTML event attributes can execute JavaScript code directly
- HTML event attributes can call JavaScript functions
- You can assign your own event handler functions to HTML elements □ You can prevent events from being sent or being handled □ And more ... **Syntax**

<element event='some JavaScript'> <element event="some JavaScript">

Example

In the following example, an onclick attribute (with code), is added to a <button> element:

<!DOCTYPE html>

<html>

```

<body>
<button onclick="document.getElementById('demo').innerHTML=Date()">The time
is?</button>

<p id="demo"></p>

</body>
</html>

```

JavaScript HTML DOM Events

HTML DOM allows JavaScript to react to HTML events:

A JavaScript can be executed when an event occurs, like when a user clicks on an HTML element.

To execute code when a user clicks on an element, add JavaScript code to an HTML event attribute:

`onclick=JavaScript`

EXAMPLE

```

<!DOCTYPE html> <html> <body>

<h1 onclick="this.innerHTML='Ooops!'">Click on this text!</h1>

</body> </html>

```

Activity:

Apply following JS events on your web pages

Input Events

- Onblur
- onreset

Mouse Events

- Onmouseover
- Onmousedown

Click Events

- Onclick
- Ondblclick

Apply following DOM events to your webpages

- Onload
- Onchange
- Onmouseover

Results: (Program printout with output)**CODE:**

```
<!DOCTYPE html>
<html>
<head>
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.0-beta2/dist/css/bootstrap.min.css" rel="stylesheet"
integrity="sha384-BmbxuPwQa2lc/FVzBcNJ7UAyJxM6wuqIj61tLrc4wSX0szH/Ev+nYRRuWlolflfl"
crossorigin="anonymous">
<link rel="stylesheet" href="styles.css">

<title>Cafe Paradise</title>
<style>
li a:hover {
    background-color: lightgrey;
}
</style>
</head>
<body bgcolor="black" text="white">
<nav class="navbar navbar-expand-lg navbar-light bg-light">
    
    <button class="navbar-toggler" type="button" data-toggle="collapse" data-
target="#navbarSupportedContent" aria-controls="navbarSupportedContent" aria-expanded="false" aria-
label="Toggle navigation">
        <span class="navbar-toggler-icon"></span>
    </button>

    <div class="collapse navbar-collapse" id="navbarSupportedContent">
        <ul class="navbar-nav mr-auto mt-2 mt-lg-0">
            <li class="nav-item">
                <a class="nav-link" href="Restaurant_html.html">Home</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="Gallery.html">Gallery</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="Reserve.html">Reserve</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="Order_Online.html">Order Online</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#">Location</a>
            </li>
            <li class="nav-item">
                <a class="nav-link" href="#about">Login</a>
            </li>
            <li class="nav-item">
```

```

    <a class="nav-link" href="Signup.html">Sign Up</a>
  </li>

</ul>

</div>
</nav>
<br>
<h1 style="text-align: center; margin-top:20px;"> Reserve Your Table</h1>
<br>
<div style="text-align: center; border:solid 2px rgb(0, 0, 0); margin:10px 350px; padding-bottom:20px;">
  <form id="myForm">
    <br>
      <div style="margin-top: 20px;">
        <label for="fname">First name:</label>
        <input type="text" id="fname" class="textar" name="fname" placeholder="Enter First Name"
autofocus><br>
      </div>
      <div style="margin-top: 10px;">
        <label for="lname">Last name:</label>
        <input type="text" id="lname" class="textar" name="lname" placeholder="Enter Last Name"
autofocus><br><br>
      </div>
      <label>Number of guests:</label>
      <select>
        <option value="2">2</option>
        <option value="3">3</option>
        <option value="4">4</option>
        <option value="5">5</option>
        <option value="6">6</option>
        <option value="7">7</option>
        <option value="8">8</option>
        <option value="9">9</option>
        <option value="10+">10 or more</option>
      </select><br><br>
      <label for="start">Start date:</label>
      <input type="date" id="start" required name="trip-start" value="2020-02-18" min="2020-02-18"
max="2020-06-31">
      <label>Time:</label>
      <input type="time" id="time" name="time" required autocomplete="off"><br><br>
      <label>Phone number:</label>
      <input type="tel" id="phone" name="phone"><br><br>
      <label for="pwd">Enter the OTP sent to your phone number:</label>
      <input type="password" id="pwd" name="pwd"><br><br>
      <h6>Select the type of table you want :</h6><br>
      <a href="Gallery.html" id="showPage" onerror="error()">Click here to Check Some Pictures</a><br>
      <input type="radio" checked="" name="tabletype" value="AC">
      <label for="male">AC</label>
      <input type="radio" onclick="openWindow()" name="tabletype" value="NON-AC">
      <label for="female">NON-AC</label>
      <input type="radio" name="tabletype" value="DELUXE">
      <label for="other">Deluxe(table with a sepeate room)</label><br><br>

```

```

<input type="checkbox" name="accept" value="accept">
<label for="accept"> I accept the terms and conditions of the hotel.</label><br><br>
<input type="submit" class="btn btn-dark" id="submitbtn" value="Book My Table">
<input type="reset" id="resetbtn" value="Reset">
</form>
</div><br>
</body>
<script>
    // onAbort
    function abort(e){
        e.target.style.text="no image found";
    }
    // onBlur
    // onFocus
    var x = document.getElementById("myForm");
    x.addEventListener("focus", myFocusFunction, true);
    x.addEventListener("blur", myBlurFunction, true);

    function myFocusFunction() {
        document.getElementById("phone").style.background = "yellow";
    }

    function myBlurFunction() {
        document.getElementById("phone").style.background = "";
    }

    // onChange
    document.getElementById("fname").onchange = function() {myFunction()};

    function myFunction() {
        var x = document.getElementById("fname");
        x.value = x.value.toUpperCase();
    }

    // onClick
    document.getElementById("submitbtn").onclick = function formSubmit(){
        alert("button was clicked !!");
    };

    // onDblclick
    document.getElementById("resetbtn").ondblclick = function formreset(){
        alert("Form has been reset!!!");
    };

    // onError
    function error(e){
        e.target.style.text="no image found";
    }

    // onKeyPress
    // onKeyDown
    // onKeyUp
    document.getElementById("pwd").onkeydown = function password(e){
        e.target.style.background="red";
    };

```

```

document.getElementById("pwd").onkeyup = function password(e){
  e.target.style.background="";
};

document.getElementById("time").onkeypress = function password(e){
  e.target.style.background="rgb(255, 184, 184)";
};

// onLoad
document.getElementById("myForm").addEventListener("load", alert("Page has been loaded!!!"));

// onMouseDown
// onMouse
document.getElementById("submitbtn").onmousedown = function formSubmit(e){
  e.target.style.background = "rgb(255, 184, 184)";
  e.target.style.border="3px solid maroon";
};
document.getElementById("submitbtn").onmouse = function formSubmit(e){
  e.target.style.removeProperty('border');
  e.target.style.removeProperty('background');
};

//Hyperlink /another page opening(web page loading)
var link = document.getElementById('showPage');
link.onclick = showPage;
function showPage() {
  alert('Look at these amazing photos');
}

// mouseEventhandler
var textar = document.getElementsByClassName("textar");

for(var i=0; i < textar.length; i++){
  textar[i].onmouseover = mouseEventHandler;
  textar[i].onmouseout = mouseEventHandler;
}
function mouseEventHandler(e){
  if(e.type == "mouseover"){
    e.target.style.background = "rgb(255, 184, 184)";
    e.target.style.border="3px solid maroon";
  }
  else {
    e.target.style.removeProperty('border');
    e.target.style.removeProperty('background');
  }
}

// You open different page# with some advertisement(pop up Window)
function openWindow() {
  newWindow = window.open("", null,
  "height=200,width=400,status=yes,toolbar=no,menubar=no,location=no");
  newWindow.document.write("<h1>NON-AC table selected!!!</h1>");
}

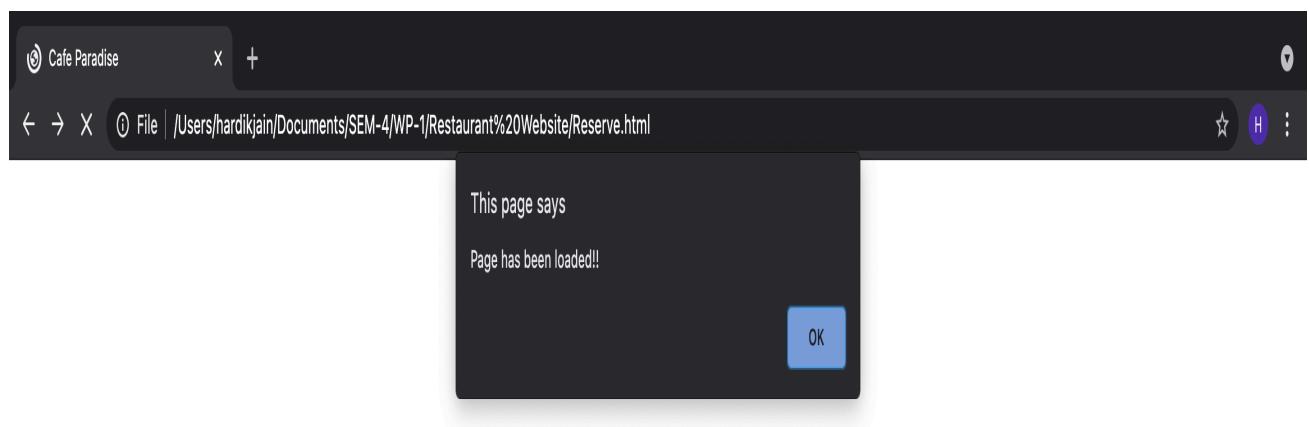
</script>
</html>

```

OUTPUT :

We have implemented this experiment on the ‘Reserve Table’ Page of Our Website

1. When the page loads, we used the onload eventhandler to give this alert:



2. Initial form looks like this:

← → C ⌂ File | /Users/hardikjain/Documents/SEM-4/WP-1/Restaurant%20Website/Reserve.html

PARADISE CAFE Home Gallery Reserve Order Online Location Login Sign Up

Reserve Your Table



First name:

Last name:

Number of guests:

Start date: Time:

Phone number:

Enter the OTP sent to your phone number:

Select the type of table you want :

[Click here to Check Some Pictures](#)



First name:

Last name:

Number of guests:

Start date: Time:

Phone number:

Enter the OTP sent to your phone number:

Select the type of table you want :

[Click here to Check Some Pictures](#)

AC NON-AC Deluxe(table with a seperate room)

I accept the terms and conditions of the hotel.

3. When you hover on the text areas , the background color and border changes :

Reserve Your Table



First name:

Last name:

Number of guests:

Start date: Time:

Phone number:

Enter the OTP sent to your phone number:

Select the type of table you want :

[Click here to Check Some Pictures](#)

AC NON-AC Deluxe(table with a sepearte room)

I accept the terms and conditions of the hotel.

4. Once you have filled the name, it will automatically convert it to uppercase.:

Before:



First name:

Last name:

Number of guests:

After:



First name: **HARDIK**

Last name: **Enter Last Name**

Number of guests: **2**

5. When you focus on the phone number area its background colour changes to yellow and again disappears if you blur.(Focus,Blur)

7

Start date: 18-02-2020 Time: 16:10

Phone number: **90003652460**

Enter the OTP sent to your phone number: 90003652460
25628

Select the type of table you want:

(click here to check some pictures)

AC NON-AC Deluxe(table with a separate room)

After blur :

1

Start date: 18-02-2020 Time: 16:10

Phone number: **90003652460**

Enter the OTP sent to your phone number: **90003652460**

Select the type of table you want:

(click here to check some pictures)

6. When you start filling the OTP ,if you press a key down **its colour will change to red.(onKeyPress, onKeyDown ,onKeyUp):**

Start date: 18/02/2020

Phone number: **8879108369**

Enter the OTP sent to your phone number: **.....**

Select the type of table you want :

[Click here to Check Some Pictures](#)

AC NON-AC Deluxe(table with a seperate room)

7. When you click on **the hyperlink** it gives you an alert and directs to the '**Gallery**' Page.

/hardikjain/Documents/SEM-4/WP-1/Restaurant%20Website/Reserve.html

This page says
Look at these amazing photos

OK



First name:

Last name:

Number of guests:

Start date:

Phone number: **8879108369**

Enter the OTP sent to your phone number: **.....**

Select the type of table you want :

[Click here to Check Some Pictures](#)

AC NON-AC Deluxe(table with a seperate room)

8. If you select any of the table options, **it opens a new window**:

NON-AC table selected!!!

Number of guests: 7

Start date: 18-02-2020 Time: 16:10

Phone number: 90003652460

Enter the OTP sent to your phone number: •

Select the type of table you want :

(click here to check some pictures)

AC NON-AC Deluxe(table with a separate room)

I accept the terms and conditions of the hotel.

9. Lastly, if you double click the reset button then an alert is thrown:

This page says
Form has been reset!!

OK

First name: Enter First Name

Last name: Enter Last Name

Number of guests: 2

Start date: 18/02/2020 Time: --:-- --

Phone number: [REDACTED]

Enter the OTP sent to your phone number: [REDACTED]

Select the type of table you want :

[Click here to Check Some Pictures](#)

AC NON-AC Deluxe(table with a separate room)

I accept the terms and conditions of the hotel.

Questions:

Q1) What are the different types of load events

Ans. These are the top 8 types of JavaScript Event discussed below:

1. User Interface events

These occur as the result of any interaction with the browser window rather than the HTML page. In these events, we attach the event listener to the window object, not the document object. The various UI events are as follows.

Example:load,unload,error,resize,scroll

2. Focus and blur events

These events fire when the HTML elements you can interact with gain/ lose focus. They are most commonly used in forms and especially helpful when you want to do the following tasks:

- To show tips or feedback to users as they interact with an element within a form. The tips are usually shown in the elements other than the one the user is interacting with.
- To trigger form validation as a user moves from one control to the next without waiting to submit the form.

Example:focus,blur,focusin,focusout

3. Mouse events

These events fire when the mouse moves or the user clicks a button. All the elements of the page support these events and use the bubbling approach. These actions work differently on touchscreen devices. Preventing the default behavior of mouse events can cause unexpected results.

Example: click dblclick mousedown mouseup mouseover

4. Keyboard events

These events fire on any kind of device when a user interacts with a keyboard.

Example: input keydown keypress keyup

5. Form events

These events are common while using forms on a webpage. In particular, we see the submit event mostly in form of validation (checking form values). If the users miss any required information or enter incorrect input, validation before sending the data to the server is faster.

Example: submit change input

6. Mutation events and observers

Whenever the structure of the DOM tree changes, it triggers a **mutation event**. The change in the tree may be due to the addition or removal of a DOM node through your script. But these have an alternative that will replace them: **mutation observers**.

Example: DOMNodeInserted, DOMNodeRemoved, DOMNodeSubtreeModified.

7. HTML5 events

These are the page-level events included in the versions of the HTML5 specialization. New events support more recent devices like phones and tablets. They respond to events such as gestures and movements. You will understand them better after you master the above concepts, thus they are not discussed for now. Work with the events below for now and when you are a better developer, you can search for other events available.

Example: DOMContentLoaded, hashchange, beforeunload

8. CSS events

These events trigger when the script encounters a CSS element. As CSS is a crucial part of web development, the developers decided to add these events to js to make working with CSS easier.

Example: transitioned, animationstart, animationend.

Q2) Explain Onkeypress , onkeyup events .

Ans.

onkeyup event

The onkeyup event executes a JavaScript function when the user releases a key. The onkeyup method allows derived classes to handle the event without attaching the delegate. This is the preferred technique for handling the event in derived classes.

onkeypress event

The keypress event only fires for keys that have a printable representation. The onkeypress event is not fired for all keys, for example for ALT, SHIFT, CTRL, ESC and BACKSPACE in all browsers, this event is fired only when the user presses a key.

The difference between onkeyup and onkeypress

- onkeypress is fired when a key on your keyboard is pressed.
- onkeyup is fired when you release the key on your keyboard. If you press any key and do not release it.
- onkeypress works for letters, numbers, and symbols without meta keys on your keyboard.
- onkeyup works for all keys on your keyboard.

Outcomes:

CO3: Apply Javascript for Web Application development

Conclusion: (Conclusion to be based on the outcomes achieved) :

In this experiment, we learned about the various EventHandlers using JavaScript and implemented those on a webpage from our project.

Grade: AA / AB / BB / BC / CC / CD /DD

Signature of faculty in-charge with date

References:**Books/ Journals/ Websites:**

- “Web technologies: Black Book”, Dreamtech Publications
 - <http://www.w3schools.com>
-