|  |
| --- |
| **Question:** What is JavaScript Output method? |
| **Answer:**  Javascript output methods shows the data on the screen. |
| **Question:** How to used JavaScript Output method? |
| **Answer:**  JavaScript can give you output of data in following different ways:   1. Writing into an HTML element, using innerHTML. 2. Writing into the HTML output using document.write(). 3. Writing into an alert box, using window.alert(). 4. Writing into the browser console, using console.log(). |
| **Question:** How to use JavaScript Events to do all examples? |
| **Answer:**  Following is the list of JavaScript listeners’ definitions which have specific types of methods.   * Onload:  When your page loads, it performs accordingly. * Onclick: When a user clicks on a button or inputs it occurs. * Onmouseover: When a user mouses over on the button. * Onfocus:  Certain scenarios when a user keeps the cursor in a form field. * Onblur: If a particular form field leaves within it.   Below are the types of JavaScripts with examples of code syntax. Onclick Events and Syntax We can define it as a mouse event that stimulates as per the code logic you determine in your code. Here is the code snippet we can use.  **Code**:  <!doctype html>  <html>    <head>      <script>        function Greet() {          alert('Hello World!');        }      </script>    </head>    <body>      <button type="button" onclick="Greet()">Please click here! </button>    </body>  </html> Output JavaScript_events_1. Onkeyup Event and Syntax You can use this Javascript event in a scenario where you press a keyboard event and it performs as per your code logic. We can use the below code snippet.  **Code**:  <!DOCTYPE html>  <html>  <body>  Enter your First name: <input type="text" id="firstname" onkeyup="myKeyUpFunction()">  <p>My First name is: <span id="Test"></span></p>  <script>  function myKeyUpFunction() {    var input = document.getElementById("firstname").value;    document.getElementById("Test").innerHTML = input;  }  </script>  </body>  </html> Output JavaScript_events_2 Get the IIT-M Advantage! PCP In Full Stack Development - MERN[EXPLORE COURSE](https://www.simplilearn.com/full-stack-developer-course-and-certification-iitm-pravartak?source=GhPreviewCTABanner)  Get the IIT-M Advantage! Onmouseover Event and Syntax We can use this event for hovering the mouse pointer when we put the cursor and it performs as per the logic of the element which is connected to and its child's elements. We can use the below code snippet.  **Code**:  <!DOCTYPE html>  <html>  <body>  <h1 id="demo">Test Mouse over me</h1>  <script>  document.getElementById("demo").onmouseover = function() {mouseOver()};  function mouseOver() {    document.getElementById("demo").style.color = "Purple";  }  </script>  </body>  </html> Output JavaScript_events_3 Onmouseout Event When you leave the mouse cursor it moves to the element that controls a mouseout event;, a function associated with it is executed. The below code snippet can help you to understand the logic.  **Code**  <!DOCTYPE html>  <html>  <body>  <h1 id="demo">Test Mouse over me</h1>  <script>  document.getElementById("demo").onmouseout = function() {mouseOut()};  function mouseOut() {    document.getElementById("demo").style.color = "Red";  }  </script>  </body>  </html> Output JavaScript_events_4. Onchange Event and Syntax This event identifies the variance in the value of any element listing to this event. The best example of this is when text and dropdown list change events. The below code snippet can help you understand the logic as to how it converts the input name with the upper case when text changes.  **Code**:  <!DOCTYPE html>  <html>  <body>  Please Enter name: <input type="text" id="Firstname">  <script>  document.getElementById("Firstname").onchange = function() {myFunction()};  function myFunction() {    var x = document.getElementById("Firstname");    x.value = x.value.toUpperCase();  }  </script>  </body>  </html> Output JavaScript_events_5.  After changing the event:  JavaScript_events_6. Unleash Your Career as a Full Stack Developer! PCP In Full Stack Development - MERN[EXPLORE COURSE](https://www.simplilearn.com/full-stack-developer-course-and-certification-iitm-pravartak?source=GhPreviewCTABanner)  Unleash Your Career as a Full Stack Developer! Onload Event and Syntax The JavaScript onload event can be utilized when we have a specific requirement to execute a specific function once the page is represented fully. The below code snippet can help you understand the logic.  **Code**:  <!DOCTYPE html>  <html>  <body onload="checkyourCookies()">  <p id="OnloadTest"></p>  <script>  function checkyourCookies() {    var text = "";    if (navigator.cookieEnabled == true) {      text = "your web page Cookies are active.";    } else {      text = "your web page Cookies are not active.";    }    document.getElementById("OnloadTest").innerHTML = text;  }  </script>  </body>  </html> Output JavaScript_events_7. Onfocus Event and Syntax This Javascript function performs when the given instruction receives the focus as per the change or click event. The below code snippet can help you understand the logic.  **Code**:  <!DOCTYPE html>  <html>  <body>  <p>This is the best scenario to uses the addEventListener() function to attach a "focus" event to an input element box.</p>  Enter your First name: <input type="text" id="Firstname">  <script>  document.getElementById("Firstname").addEventListener("focus", myFunction);  function myFunction() {    document.getElementById("Firstname").style.backgroundColor = "DarkBlue";  }  </script>  </body>  </html> Output JavaScript_events_8  Input text focused  JavaScript_events_9. Onblur Event and Syntax This Javascript Onblur event triggers when a certain object loses focus. We can execute the below code to understand how to implement it.  **Code**  <!DOCTYPE html>  <html>  <body>  <p>This code snippet uses the addEventListener() method and performs a "blur" event to an input element.</p>  <p>please write something and see the result (blur).</p>  <input type="text" id="fname">  <script>  document.getElementById("fname").addEventListener("blur", myFunction);  function myFunction() {    alert("your Input element lost focus.");  }  </script>  </body>  </html> Output |