

Basic Javascript

Event Handling & Validation

Yash Lathiya

|  |  |  |  |
| --- | --- | --- | --- |
| Sr No. | Topic | Date | Page No. |
|  |  |  |  |
| 1 | Javascript Introduction | 31/07/2023 | 2 |
| 2 | Javascript Events | 31/07/2023 | 3 |
| 3 | Javascript Validation | 01/08/2023 | 5 |

Table of Contents

JavaScript Introduction

* Javascript is object-oriented scripting language.
* Lightweight programming language.
* Cross-platform language.
* Javascript is used to access & manipulate html & CSS pages in web development.
* It is used to make page user interactive.
* It fires events on basis of user – activity.

Features of Javascript

* All web-browser support.
* Easy to understand Syntax.
* Pure object – oriented programming language.
* Case-sensitive.
* Supports all OS versions & types.
* Client – side validation.

How to add Javascript to HTML page ?

1. Embedding Code

We can add script tag in HTML file, it’s suitable to add them just before ending of body tag.

<body>

// HTML code

<script>

//JS code

</script>

</body>

1. Inline Code

<button onClick=”alert(“button is clicked”);”>Button</button>

1. External file

<script scr=”script.js”></script>

Javascript Events

* Javascript custom code is fired on user’s specific activity, it is called javascript events.
* Following events are majorly used to javascript:
* Mouse Events

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| **click** | **onclick** | **When mouse click on an element** |
| **mouseover** | **onmouseover** | **When the cursor of the mouse comes over the element** |
| **mouseout** | **onmouseout** | **When the cursor of the mouse leaves an element** |
| **mousedown** | **onmousedown** | **When the mouse button is pressed over the element** |
| **mouseup** | **onmouseup** | **When the mouse button is released over the element** |
| **mousemove** | **onmousemove** | **When the mouse movement takes place.** |

* Keyboard Events

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| **Keydown & Keyup** | **onkeydown & onkeyup** | **When the user press and then release the key** |

* Form Events

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| **focus** | **onfocus** | **When the user focuses on an element** |
| **submit** | **onsubmit** | **When the user submits the form** |
| **blur** | **onblur** | **When the focus is away from a form element** |
| **change** | **onchange** | **When the user modifies or changes the value of a form element** |

* Window Events

|  |  |  |
| --- | --- | --- |
| **Event Performed** | **Event Handler** | **Description** |
| **load** | **onload** | **When the browser finishes the loading of the page** |
| **unload** | **onunload** | **When the visitor leaves the current webpage, the browser unloads it** |

* Behaviour of events on nested elements

Javascript is primarily give priority to inner most nested element’s event, After calling it - it will call one by one outer elements. It is also known as Event Bubbling.

If we want reverse execution behaviour, that’s possible by

{ capture: true } property which is known as event capturing.

Javascript Validation

* If we want to structured data inputs from user-side, we can use validation, which will allow form submission only if entered data is validated by programmer’s logic:
* Directly Demo is prepared..