

1. **onabort**: Triggered when a resource (such as an image) is aborted during loading.  

```
image.onabort = function() {  
    console.log("Image loading aborted");  
};
```
2. **onafterprint**: Triggered after the user prints the document.  

```
window.onafterprint = function() {  
    console.log("Document has been printed");  
};
```
3. **onanimationend**: Triggered when a CSS animation completes.  

```
element.onanimationend = function() {  
    console.log("Animation completed");  
};
```
4. **onanimationiteration**: Triggered when a CSS animation completes one iteration.  

```
element.onanimationiteration = function() {  
    console.log("Animation iteration completed");  
};
```
5. **onanimationstart**: Triggered when a CSS animation starts.  

```
element.onanimationstart = function() {  
    console.log("Animation started");  
};
```
6. **onappinstalled**: Triggered after the user installs the application.  

```
window.onappinstalled = function() {  
    console.log("App installed");  
};
```
7. **onauxclick**: Triggered when a non-primary mouse button is clicked.  

```
element.onauxclick = function() {  
    console.log("Auxiliary mouse button clicked");  
};
```
8. **onbeforeprint**: Triggered before the user prints the document.  

```
window.onbeforeprint = function() {  
    console.log("Before printing");  
};
```

9. **onbeforeunload**: Triggered before the document is about to be unloaded (e.g., when the user navigates away).  

```
window.onbeforeunload = function() {  
    return "Are you sure you want to leave?";  
};
```
10. **onblur**: Triggered when an element loses focus.  

```
element.onblur = function() {  
    console.log("Element lost focus");  
};
```
11. **oncancel**: Triggered when a user cancels an interaction (e.g., file upload or printing).  

```
request.oncancel = function() {  
    console.log("Request canceled");  
};
```
12. **oncanplay**: Triggered when media can start playing (but not necessarily when it has started).  

```
audio.oncanplay = function() {  
    console.log("Audio can start playing");  
};
```
13. **oncanplaythrough**: Triggered when media can be played through to the end without interruption.  

```
video.oncanplaythrough = function() {  
    console.log("Video can be played through");  
};
```
14. **onchange**: Triggered when the value of a form element changes.  

```
input.onchange = function() {  
    console.log("Input value changed");  
};
```
15. **onclick**: Triggered when an element is clicked.  

```
button.onclick = function() {  
    console.log("Button clicked");  
};
```
16. **onclose**: Triggered when a WebSocket connection is closed.

```
socket.onclose = function() {  
    console.log("WebSocket connection closed");  
};
```

17. **oncontextmenu**: Triggered when the right mouse button is clicked.

```
element.oncontextmenu = function() {  
    console.log("Context menu opened");  
};
```

18. **oncopy**: Triggered when the user copies content.

```
document.oncopy = function() {  
    console.log("Content copied");  
};
```

19. **oncuechange**: Triggered when the cue changes in a `<track>` element.

```
track.onscuechange = function() {  
    console.log("Cue changed");  
};
```

20. **oncut**: Triggered when the user cuts content.

```
document.oncut = function() {  
    console.log("Content cut");  
};
```

21. **ondblclick**: Triggered when an element is double-clicked.

```
element.ondblclick = function() {  
    console.log("Element double-clicked");  
};
```

22. **ondrag**: Triggered when an element or text selection is being dragged.

```
element.ondrag = function() {  
    console.log("Element dragged");  
};
```

23. **ondragend**: Triggered when a drag operation is completed.

```
element.ondragend = function() {  
    console.log("Drag operation completed");  
};
```

24. **ondragenter**: Triggered when a dragged element enters a valid drop target.

```
dropTarget.ondragenter = function() {
    console.log("Dragged element entered drop target");
};
```

25. **ondragleave**: Triggered when a dragged element leaves a valid drop target.

```
dropTarget.ondragleave = function() {
    console.log("Dragged element left drop target");
};
```

26. **ondragover**: Triggered when a dragged element is being dragged over a valid drop target.

```
dropTarget.ondragover = function() {
    console.log("Element being dragged over drop target");
};
```

27. **ondragstart**: Triggered when the user starts to drag an element.

```
element.ondragstart = function() {
    console.log("Drag operation started");
};
```

28. **ondrop**: Triggered when a dragged element is dropped on a valid drop target.

```
dropTarget.ondrop = function() {
    console.log("Element dropped on drop target");
};
```

29. **ondurationchange**: Triggered when the duration of the media changes.

```
video.ondurationchange = function() {
    console.log("Media duration changed");
};
```

30. **onemptied**: Triggered when the media has become empty.

```
audio.onemptied = function() {
    console.log("Media has become empty");
};
```

31. **onended**: Triggered when the media has reached the end.

```
audio.onended = function() {
    console.log("Media has ended");
};
```

32. **onerror**: Triggered when an error occurs during the loading of an external resource.

```
image.onerror = function() {  
    console.log("Error loading image");  
};
```

33. **onfocus**: Triggered when an element receives focus.

```
input.onfocus = function() {  
    console.log("Input element focused");  
};
```

34. **ongotpointercapture**: Triggered when an element receives pointer capture.

```
element.ongotpointercapture = function() {  
    console.log("Element received pointer capture");  
};
```

35. **oninput**: Triggered when the value of an input element changes.

```
input.oninput = function() {  
    console.log("Input value changed");  
};
```

36. **oninvalid**: Triggered when a form element is invalid.

```
input.oninvalid = function() {  
    console.log("Input is invalid");  
};
```

37. **onkeydown**: Triggered when a key is pressed down.

```
document.onkeydown = function() {  
    console.log("Key pressed down");  
};
```

38. **onkeypress**: Triggered when a key is pressed.

```
document.onkeypress = function() {  
    console.log("Key pressed");  
};
```

39. **onkeyup**: Triggered when a key is released.

```
document.onkeyup = function() {  
    console.log("Key released");  
};
```

40. **onload**: Triggered when a resource and its dependent resources have finished loading.
- ```
window.onload = function() {  
    console.log("Page loaded");  
};
```
41. **onloadeddata**: Triggered when the first frame of the media has finished loading.
- ```
video.onloadeddata = function() {  
    console.log("First frame of video loaded");  
};
```
42. **onloadedmetadata**: Triggered when metadata for the media has been loaded.
- ```
video.onloadedmetadata = function() {  
    console.log("Metadata for video loaded");  
};
```
43. **onloadstart**: Triggered when loading of the media begins.
- ```
video.onloadstart = function() {  
    console.log("Loading of video started");  
};
```
44. **onlostpointercapture**: Triggered when an element loses pointer capture.
- ```
element.onlostpointercapture = function() {  
    console.log("Element lost pointer capture");  
};
```
45. **onmousedown**: Triggered when a mouse button is pressed down.
- ```
element.onmousedown = function() {  
    console.log("Mouse button pressed down");  
};
```
46. **onmouseenter**: Triggered when the mouse pointer enters an element.
- ```
element.onmouseenter = function() {  
    console.log("Mouse pointer entered element");  
};
```
47. **onmouseleave**: Triggered when the mouse pointer leaves an element.
- ```
element.onmouseleave = function() {  
    console.log("Mouse pointer left element");  
};
```

48. **onmousemove**: Triggered when the mouse pointer is moved.
- ```
element.onmousemove = function() {  
    console.log("Mouse moved");  
};
```
49. **onmouseout**: Triggered when the mouse pointer leaves an element.
- ```
element.onmouseout = function() {  
    console.log("Mouse pointer left element");  
};
```
50. **onmouseover**: Triggered when the mouse pointer enters an element.
- ```
element.onmouseover = function() {  
    console.log("Mouse pointer entered element");  
};
```
51. **onmouseup**: Triggered when a mouse button is released.
- ```
element.onmouseup = function() {  
    console.log("Mouse button released");  
};
```
52. **onpaste**: Triggered when the user pastes content.
- ```
document.onpaste = function() {  
    console.log("Content pasted");  
};
```
53. **onpause**: Triggered when the media is paused.
- ```
video.onpause = function() {  
    console.log("Media paused");  
};
```
54. **onplay**: Triggered when the media begins to play.
- ```
video.onplay = function() {  
    console.log("Media started playing");  
};
```
55. **onplaying**: Triggered when the media has started playing.
- ```
video.onplaying = function() {  
    console.log("Media playing");  
};
```
56. **onpointercancel**: Triggered when a pointer event is canceled.

```
element.onpointercancel = function() {  
    console.log("Pointer event canceled");  
};
```

57. **onpointerdown**: Triggered when a pointer is pressed down.

```
element.onpointerdown = function() {  
    console.log("Pointer pressed down");  
};
```

58. **onpointerenter**: Triggered when a pointer enters an element.

```
element.onpointerenter = function() {  
    console.log("Pointer entered element");  
};
```

59. **onpointerleave**: Triggered when a pointer leaves an element.

```
element.onpointerleave = function() {  
    console.log("Pointer left element");  
};
```

60. **onpointermove**: Triggered when a pointer is moved.

```
element.onpointermove = function() {  
    console.log("Pointer moved");  
};
```

61. **onpointerout**: Triggered when a pointer leaves an element.

```
element.onpointerout = function() {  
    console.log("Pointer left element");  
};
```

62. **onpointerover**: Triggered when a pointer enters an element.

```
element.onpointerover = function() {  
    console.log("Pointer entered element");  
};
```

63. **onpointerup**: Triggered when a pointer is released.

```
element.onpointerup = function() {  
    console.log("Pointer released");  
};
```

64. **onprogress**: Triggered periodically as the resource is being downloaded.



```
xhr.onprogress = function(event) {  
    console.log('Downloaded ${event.loaded} bytes');  
};
```

65. **onratechange**: Triggered when the playback rate of the media changes.

```
video.onratechange = function() {  
    console.log("Playback rate changed");  
};
```

66. **onreset**: Triggered when a form is reset.

```
form.onreset = function() {  
    console.log("Form reset");  
};
```

67. **onresize**: Triggered when the browser window is resized.

```
window.onresize = function() {  
    console.log("Window resized");  
};
```

68. **onscroll**: Triggered when the document view or an element is scrolled.

```
window.onscroll = function() {  
    console.log("Document or element scrolled");  
};
```

69. **onsearch**: Triggered when the user submits a search within a search box.

```
searchInput.onsearch = function() {  
    console.log("Search submitted");  
};
```

70. **onseeked**: Triggered when the seeking operation has completed.

```
video.onseeked = function() {  
    console.log("Seeking operation completed");  
};
```

71. **onseeking**: Triggered when the seeking operation is in progress.

```
video.onseeking = function() {  
    console.log("Seeking operation in progress");  
};
```

72. **onselect**: Triggered when text within an element is selected.

```
input.onsselect = function() {  
    console.log("Text selected");  
};
```

73. **onselectstart**: Triggered when the user starts to select content.

```
element.onselectstart = function() {  
    console.log("Selection started");  
};
```

74. **onstalled**: Triggered when the media playback is interrupted unexpectedly.

```
video.onstalled = function() {  
    console.log("Media playback stalled");  
};
```

75. **onsubmit**: Triggered when a form is submitted.

```
form.onsubmit = function() {  
    console.log("Form submitted");  
};
```

76. **onsuspend**: Triggered when the media is intentionally suspended.

```
video.onsuspend = function() {  
    console.log("Media suspended");  
};
```

77. **ontimeupdate**: Triggered when the playback position of the media changes.

```
video.ontimeupdate = function() {  
    console.log("Playback position changed");  
};
```

78. **ontoggle**: Triggered when the user toggles the state of an element.

```
checkbox.ontoggle = function() {  
    console.log("Checkbox toggled");  
};
```

79. **ontouchcancel**: Triggered when a touch event is canceled.

```
element.ontouchcancel = function() {  
    console.log("Touch event canceled");  
};
```

80. **ontouchend**: Triggered when a touch event ends.

```
element.ontouchend = function() {  
    console.log("Touch event ended");  
};
```

81. **ontouchmove**: Triggered when a touch point is moved along the touch surface.

```
element.ontouchmove = function() {  
    console.log("Touch point moved");  
};
```

82. **ontouchstart**: Triggered when a touch event starts.

```
element.ontouchstart = function() {  
    console.log("Touch event started");  
};
```

83. **ontransitionend**: Triggered when a CSS transition completes.

```
element.ontransitionend = function() {  
    console.log("Transition completed");  
};
```

84. **onunload**: Triggered when a page is being unloaded.

```
window.onunload = function() {  
    console.log("Page is being unloaded");  
};
```

85. **onvolumechange**: Triggered when the volume of the media changes.

```
video.onvolumechange = function() {  
    console.log("Volume changed");  
};
```

86. **onwaiting**: Triggered when the media playback is waiting for data.

```
video.onwaiting = function() {  
    console.log("Media playback waiting for data");  
};
```