Events & Event Object in Javascript

→ Events are actions that happen in the webpage like cliking a button or submit the form. If the user selects a button on a webpage, we might want to respond to that action by displaying an alert on the webpage. Each available event has an event handler, which

is a block of code that runs when the event occurred. Event handlers are sometimes called event listeners. The listener listens out for the event happening, and the handler is the code that is run in response to it happening. There are many ways to assign a handler:

→ HTML - attribute:
We can set a handler in HTML with an attribute named on<event>. For ex: to assign a click handler for an input, we can use onclick, e.g:

```
<INPUT VALUE="CLICK HERE" ONCLICK="ALERT('CLICK HERE!')" TYPE="BUTTON">
```

→ The code inside onclick runs on mouse click. But note that the HTML - attribute is not a convenient place to write a lot of code, so it will better create a Javascript function and call it there.

→ ADDEVENTLISTENER():
The Javascript ADDEVENTLISTENER() method allows the programmer to set up functions to be called when a specified event happens, such as when a user clicks a button. The purpose of using the ADDEVENTLISTENER()

method is to attach an event handler to the specified element. While using the ADDEVENTLISTENER() method, remember that it attaches an event handler to an element without overwriting existing event handlers. We can add many event handlers to one element. Removing event is as simple as adding an event ~~that~~ handlers. This task can easily be done by using the REMOVE EVENTLISTENER() method.

SYNTAX:

```
ELEMENT.ADDEVENTLISTENER(EVENT, FUNCTION, USECAPTURE);
```

→ The first parameter is the type of event like "click" or "mousedown".
→ The second parameter is the function we want to call when the event occurs.
→ The third parameter is optional. It is a boolean value specifying whether to use event bubbling or event capturing.

Ex:
Alert "EVENT OCCURED" when the user clicks on an element.

```
DOCUMENT.ADDEVENTLISTENER("CLICK", FUNCTION(){
    ALERT("EVENT OCCURED");
});
```

Page No.
Date

→ Passing Event as a Parameter:
→ Sometimes we may want to know more information about the event, such as what element was clicked. When an event happens, the browser creates an event object, puts details into it and passes it as an argument to the handler.
Eg:

```
DOCUMENT.ADDEVENTLISTENER("CLICK", FUNCTIO
N() {
    ALERT(EVENT.TYPE);
});
```

→ Event Properties and Methods:

PROPERTY/METHOD	DESCRIPTION
→ ALTKEY	→ It will return whether the "ALT" key was pressed when the mouse event was triggered.
→ BUTTON	→ It will return which mouse button was pressed when the mouse event was triggered.
→ CHARCODE	→ It will return the Unicode character code

	of the key that triggered the onkeypress event.
→ CLIENTX	→ It will return the horizontal coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered
→ CLIENTY	→ It will return the vertical coordinate of the mouse pointer, relative to the current window, when the mouse event was triggered.
→ CODE	→ It will return the code of the key that triggered the event.
→ DELTAX	→ It will return the horizontal scroll amount of a mouse wheel (x-axis).
→ DELTAY	→ It will return the vertical scroll amount of a mouse wheel (y-axis).
→ DELTAZ	→ It will return the scroll amount of a mouse wheel for the (z-axis).
→ DETAIL	→ It will return a number that indicates how many times the mouse was clicked.

→ KEYCODE	→ It will returns the UNICODE character code of the key that triggered the onkeypress event, or the Unicode key code of the key that triggered the onkeydown or onkeyup event.
→ LOCATION	→ It will returns the location of a key on the keyboard or device.
→ PAGEX	→ It will return the horizontal coordinate of the mouse pointer, relative to the document, when the mouse event was triggered.
→ PAGEY	→ It will return the vertical coordinate of the mouse pointer, relative to the document, when the mouse event was triggered.
→ SCREENX	→ It will returns the horizontal coordinate of the mouse pointer, relative to the screen, when an event was triggered.
→ SCREENY	→ It will returns the vertical coordinate of the mouse pointer, relative

Page No.
Date

to the screen, when an event was triggered.

→ SHIFTKEY → It will return whether the "SHIFT" key was pressed when ~~the~~ an event was triggered.

→ TYPE → It will return the name of the event.

→ WHICH → It will return which mouse button was pressed when the mouse event was triggered.

→ VIEW → It will return a refrence to the Window object where the event occured.