indiXpert

Javascript Events

Javascript Events

An event is a signal that something has happened. For example:

- A mouse click
- A keypress
- A page load
- A form submission
- A DOM element gaining focus

When such events happen, JavaScript can run functions, known as **event handlers**, to respond.

Event	Description
click	Triggered when an element is clicked
mouseover/mouseout	Triggered when mouse enters/leaves the element
Keyup/Keydown	Triggered when a key is pressed up/down
Load	Triggered when the page finishes loading
submit	Triggered when a form is submitted
change	Triggered when the value of an input element changes
focus/blur	Triggered when an element gains/looses focus

Javascript Events

Ways to Handle Events in JavaScript

- **Inline Event Handlers (Not Recommended)**

```
<button onclick="alert('Button clicked!')">Click Me</button>
```

- **Using HTML DOM Property**

```
const btn = document.getElementById("myBtn");
btn.onclick = function() {
  alert("Button clicked!");
};
```

- **Using addEventListener() (Best Practice)**

```
const btn = document.getElementById("myBtn");
btn.addEventListener("click", function() {
  alert("Button clicked!");
});
```

Javascript Events

Event Object

When an event occurs, an event object is automatically passed to the event handler, containing useful information.

```
element.addEventListener("click", function(event) {
  console.log(event.type); // "click"
  console.log(event.target); // The clicked element
});
```

Event Propagation: Bubbling vs Capturing

JavaScript events propagate in two phases:

- **Capturing:** From root to target element
- **Bubbling:** From target element to root.

```
parentDiv.addEventListener("click", handler, true); // Capturing
parentDiv.addEventListener("click", handler, false); // Bubbling (default)
```

Javascript Events

Preventing Default Behavior

Sometimes, the default behavior (like submitting a form or following a link) needs to be prevented.

```
form.addEventListener("submit", function(event) {
  event.preventDefault(); // Stops the form from submitting
  console.log("Form submission prevented.");
});
```

Removing Event Listeners

To remove an event listener:

```
function greet() {
  alert("Hello!");
}
button.addEventListener("click", greet);
// Later...
button.removeEventListener("click", greet);
```

Javascript Events

Event Delegation

Instead of adding event listeners to multiple child elements, you can add one to a common parent.

```
<div class="card-action">
  <button class="addtocart">Add to cart</button>
  <button class="addtowishlist">Add to wishlist</button>
</div>
```

```
document.querySelector(".card-action").addEventListener("click", function(e) {
      if (e.target.classList.contains('addtocart') ) {
        console.log('Add to cart');
      }
      if (e.target.classList.contains('addtowishlist') ) {
        console.log('Add to wishlist');
      }
});
```

indixpert

THANK YOU

PHONE NUMBER

(+91) 778 899 2897

WEBSITE

www.indixpert.com