Event Listeners and More Events | Cheat Sheet

1. Event Listeners

JavaScript offers three ways to add an Event Listener to a DOM element.

- Inline event listeners
- onevent listeners
- addEventListener()

1.1 Inline event listeners

HTML

```
<button onclick="greeting()">Greet</button>
```

JAVASCRIPT

```
function greeting() {
  console.log("Hi Rahul");
}
```

1.2 onevent listeners

HTML

```
<button id="greetBtn">Greet</button>
```

JAVASCRIPT

```
let greetBtnEl = document.getElementById("greetBtn");

greetBtnEl.onclick = function() {
  console.log("Hi Rahul");
};
```

1.3 addEventListener()

It is a modern approach to add an event listener.

Syntax:

element.addEventListener(event, function);

element - HTML element

event - event name

function - Callback function

HTML

```html
<button id="greetBtn">Greet</button>
```

JAVASCRIPT

```javascript
let greetBtn = document.getElementById("greetBtn");

greetBtn.addEventListener("click", function() {
  console.log("Hi Rahul");
});
```

2. Operators

2.1 Comparison Operators

Operator	Usage	Description
Equal (==)	a == b	returns true if both *a* and *b* values are equal.
Not equal (!=)	a != b	returns true if both *a* and *b* values are not equal.
Strict equal (===)	a === b	returns true if both *a* and *b* values are equal and of the same type.
Strict not equal (!==)	a !== b	returns true if either *a* and *b* values are not equal or of the different type.
Greater than (>)	a > b	returns true if *a* value is greater than *b* value.
Greater than or equal (>=)	a >= b	returns true if *a* value is greater than or equal to *b* value.
Less than (<)	a < b	returns true if *a* value is less than *b* value.
Less than or equal (<=)	a <= b	returns true if *a* value is less than or equal to *b* value.

2.2 Logical Operators

Operator	Usage	Description
AND (&&)	a && b	returns true if both *a* and *b* values are true.
OR (||)	a || b	returns true if either *a* or *b* value is true.
NOT (!)	!a	returns true if *a* value is not true.

3. More Events

Events are the actions by which the user or browser interact with HTML elements.

There are different types of events.

- **Keyboard Events**
- Mouse Events
- Touch Events, and many more.

3.1 Keyboard Events

Keyboard Event is the user interaction with the keyboard.

The keyboard events are

- keydown
- keyup

3.1.1 Keydown event

The

`keydown` event occurs when a key on the keyboard is pressed.

Syntax:

`element.addEventListener("keydown", function);`

JAVASCRIPT

```
let inputEl = document.createElement("input");

function printKeydown() {
  console.log("key pressed");
}

inputEl.addEventListener("keydown", printKeydown);
document.body.appendChild(inputEl);
```

3.1.2 Keyup event

The

`keyup` event occurs when a key on the keyboard is released.

Syntax:

`element.addEventListener("keyup", function);`

3.2 Event Object

Whenever an event happens, the browser creates an event object.

It consists of information about the event that has happened.

It consists of many properties and methods.

- type
- target
- key
- timeStamp
- stopPropagation , and many more.

3.2.1 Properties & Methods

For any event, event-specific properties and methods will be present.

For Example,

The `keydown` event has `key` property, whereas the `onclick` event doesn't have it.

event.type

The `event.type` property contains the type of event occurred like `click` , `keydown` , etc.

JAVASCRIPT

```
let inputEl = document.createElement("input");

function printKeydown(event) {
  console.log(event.type);  // keydown
}

inputEl.addEventListener("keydown", printKeydown);
document.body.appendChild(inputEl);
```

event.target

The `event.target` property contains the HTML element that triggered the event.

JAVASCRIPT

```
let inputElement = document.createElement("input");

function printKeydown(event) {
  console.log(event.target);  // <input></input>
}

inputElement.addEventListener("keydown", printKeydown);
document.body.appendChild(inputElement);
```

event.key

The `event.key` property contains the value of the key pressed by the user.

JAVASCRIPT

```
let inputElement = document.createElement("input");

function printKeydown(event) {
  console.log(event.key);
}

inputElement.addEventListener("keydown", printKeydown);
document.body.appendChild(inputElement);
```

Keyboard key	event.key value
Enter	Enter
a	a
A	A
1	1
*	*
<	<

Try out the keyboard events and the event object in the below Code Playground.

HTML CSS JAVASCRIPT Inspect

```
<!DOCTYPE html>
<html>
  <head>
    <link rel="stylesheet" href="https://stackpa
    <script src="https://code.jquery.com/jquery-
    <script src="https://cdn.jsdelivr.net/npm/pc
    <script src="https://stackpath.bootstrapcdn.
  </head>
  <body>
    <div class="timer-container">
      <input type="text" class="user-input" id="
      <br />
      <img src="https://d1tgh8fmlzexmh.cloudfror
      <p class="timer-display" id="timer">10</p>
    </div>
```

Submit Feedback