

# What is Event

The actions to which JavaScript can respond are called Events.



Click

- Clicking an element
- Submitting a form
- Scrolling page
- Hovering an element

# Event Handling

Handling events with React elements is very similar to handling events on DOM elements. There are some syntactic differences:

- React events are named using camelCase, rather than lowercase.
- With JSX you pass a function as the event handler, rather than a string.

## In HTML

```
<button onclick="handleClick()">Click Me</button>
```

## In React

```
<button onClick={handleClick}>Click Me</button>    // Function Component
```

```
<button onClick={this.handleClick}>Click Me</button> // Class Component
```

# Event Handling

You cannot return *false* to prevent default behavior in React. You must call *preventDefault* explicitly.

## In HTML

```
<a href="#" onclick="console.log('Clicked.');" return false"> Click me </a>
```

## In React

```
function handleClick(e) {  
  e.preventDefault();  
  console.log('Clicked.');
```

```
}
```

```
<a href="#" onClick={handleClick}> Click me </a>
```

# Event Handling

When using React you should generally not need to call `addEventListener` to add listeners to a DOM element after it is created. Instead, just provide a listener when the element is initially rendered.

```
<a href="#" onClick={handleClick}> Click me </a>
```

# Mouse Events

onClick

onContextMenu

onDoubleClick

onDrag

onDragEnd

onDragEnter

onDragExit

onDragLeave

onDragOver

onDragStart

onDrop

onMouseDown

onMouseEnter

onMouseLeave

onMouseMove

onMouseOut

onMouseOver

onMouseUp