# 📘 React Event Handling Concepts

### 🔹 React Events

* React events are objects that represent user interactions like clicks, typing, submitting forms, etc.
* They work similarly to DOM events but follow React’s own system for consistency across browsers.
* Events are triggered through JSX attributes (e.g. onClick, onChange).

function handleClick() {

alert("Button clicked!");

}

<button onClick={handleClick}>Click Me</button>

### 🔹 Event Handlers in React

* An event handler is a **function** that gets called when an event occurs.
* It’s usually defined as a method or arrow function.
* React handlers receive a SyntheticEvent as their argument.

const handleInput = (event) => {

console.log("Typed:", event.target.value);

};

<input type="text" onChange={handleInput} />

### 🔹 SyntheticEvent in React

* SyntheticEvent is a **wrapper around native browser events**.
* It provides cross-browser consistency and performance improvements.
* Contains all standard properties/methods of native events.

function handleSubmit(event) {

event.preventDefault(); // prevent native form submission

console.log("Form submitted.");

}

* Example properties: event.type, event.target, event.currentTarget

### 🔹 React Event Naming Convention

| **JSX Attribute** | **Corresponding DOM Event** |
| --- | --- |
| onClick | click |
| onChange | change |
| onSubmit | submit |
| onMouseOver | mouseover |
| onKeyDown | keydown |
| onFocus | focus |

* All React event names are **camelCase** (e.g. onClick, not onclick).
* Event handlers are passed as **functions**, not strings.

<button onclick="alert('Hi')" /> // ❌ HTML style

<button onClick={() => alert('Hi')} /> // ✅ React style