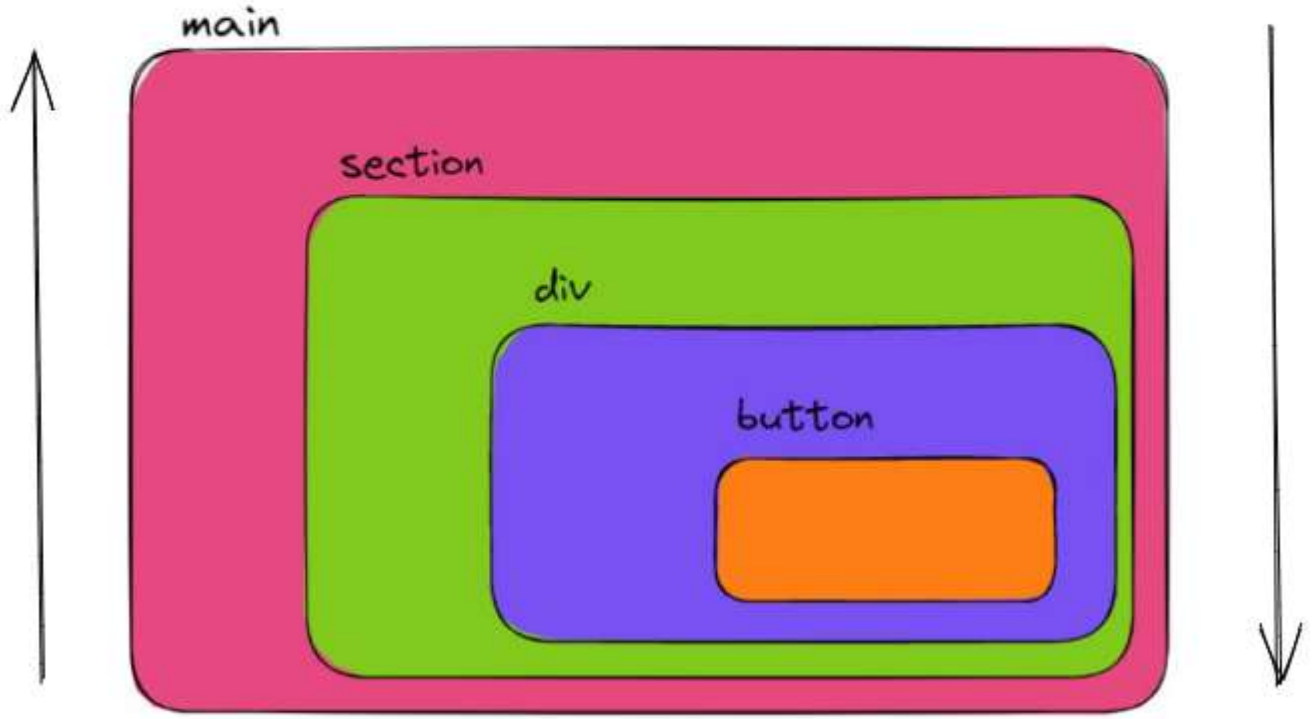
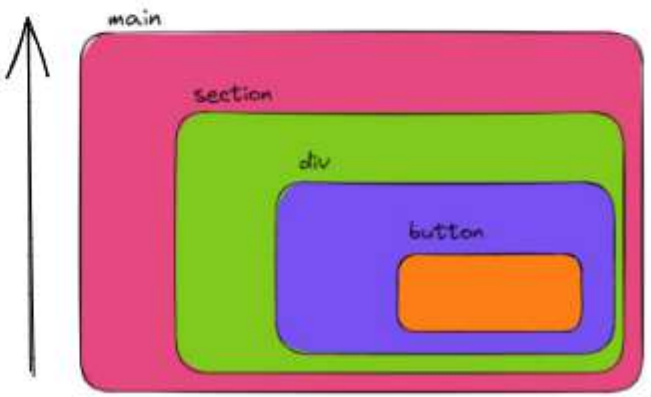
Event Bubbling & Capturing



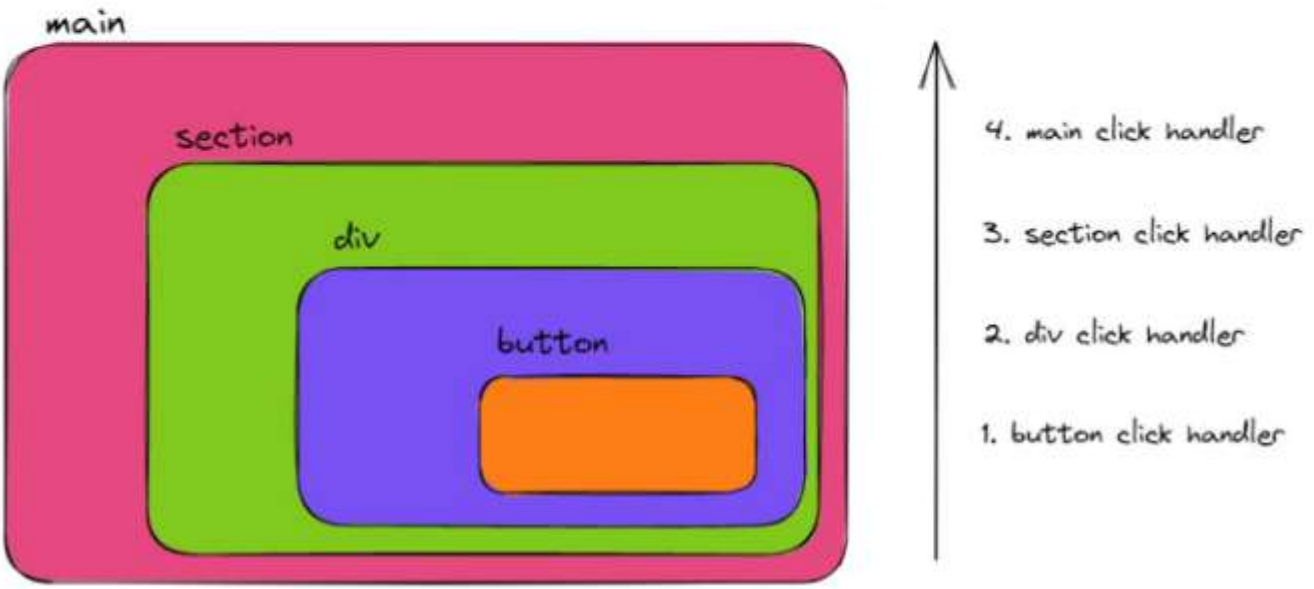
Event Bubbling

Event bubbling refers to the process where an event is first captured by the innermost element and then propagated up the DOM tree to the outermost element.



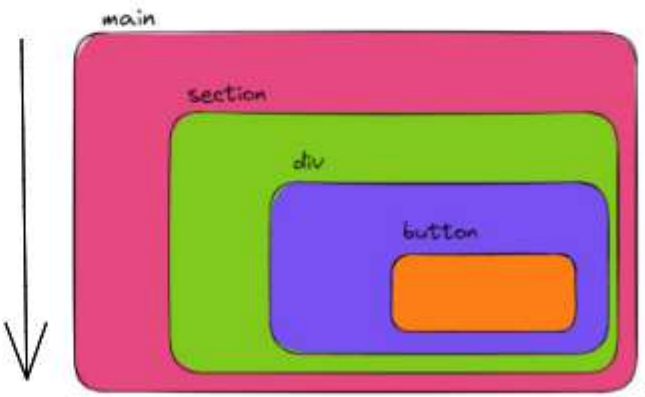
Event Bubbling

For example, let's say you have a button < div < section < main. If you click the button, the event will first be handled by the button, then the div, then the section, and finally the main tag.



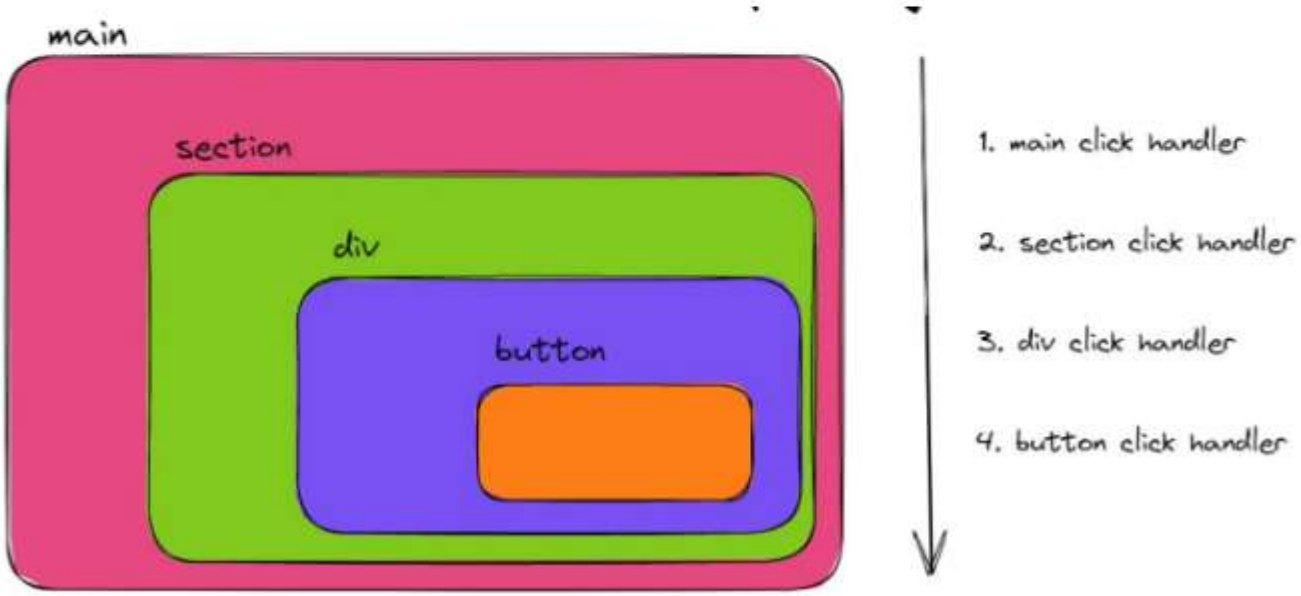
Event Capturing

Event capturing, on the other hand, works the opposite way. The event is first captured by the outermost element and then propagated down the DOM tree to the innermost element.



Event Capturing

In the same example, if you were to use event capturing instead of bubbling, the event would first be handled by the main tag, then the section, then the div, and finally the button.



Event Bubbling & Capturing

You can specify whether you want to use event bubbling or capturing when you add an event listener to an element using the third argument of the addEventListener method.

```
const mainElement = document.querySelector('main');

// Add an event listener for click events using event capturing
mainElement.addEventListener('click', function(event) {
  console.log('Event captured by main element');
}, true);

// Add an event listener for click events using event bubbling (default behavior)
mainElement.addEventListener('click', function(event) {
  console.log('Event bubbled up to main element');
});
```

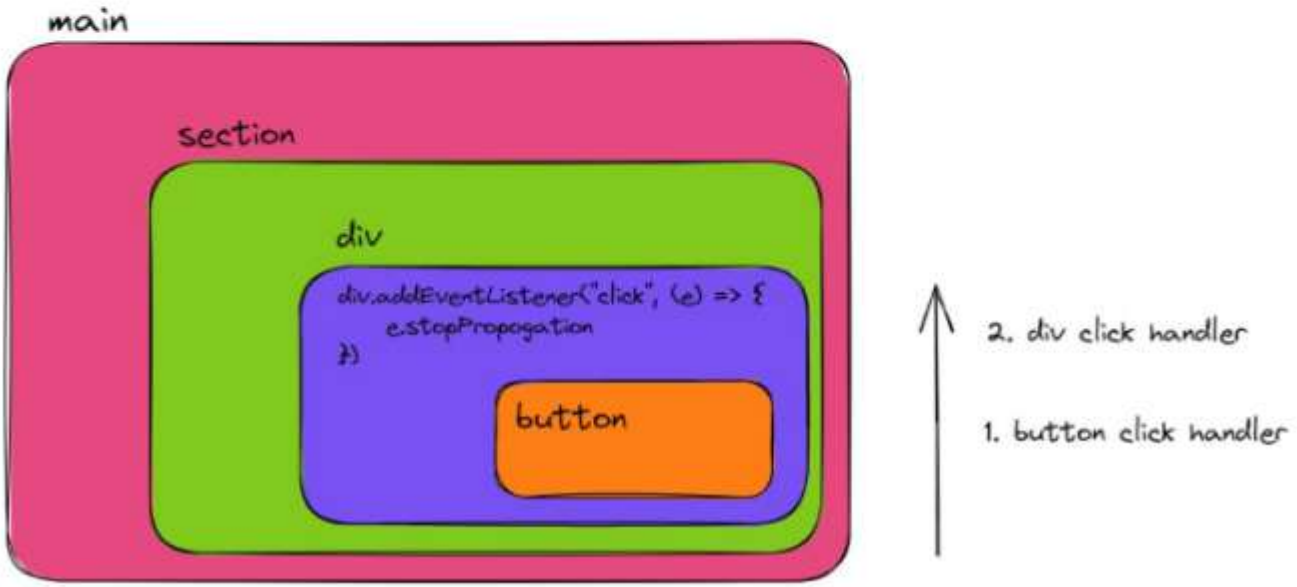
Event stopPropagation

But what if you want to stop an event from propagating further up or down the DOM tree? That's where the stopPropagation method comes in.

When you call stopPropagation on an event object inside an event listener, it prevents the event from propagating any further up or down the DOM tree.

Event stopPropagation

In the same example, If you click the div and call stopPropagation on the event object inside the div's event listener, the event will not propagate up to the section, or main tag event listeners.



That' all folks 👏

That's it for today's post on event bubbling & capturing in JavaScript! I hope this has been helpful in understanding these mechanisms for handling events in the DOM 🙌