

## Events are not only used with JavaScript

Although this course puts a lot of focus on using web events with JavaScript, they are not unique to the web browser only.

What do I mean?

Well, most programming languages have some kind of event model, and they all work in slightly different ways (i.e. in a way that differs from the way JavaScript works in a browser).



For example, Node.js is a very popular JavaScript runtime that enables developers to write JavaScript in the backend. In other words, Node.js allows you to write server-side applications with JavaScript. Important to us, is that the Node.js event model relies on listeners to listen for events and emitters to emit events periodically.

As you can already tell, it's kinda similar. However, the code can be quite different. Node.js makes use of functions like `on()` to register an event listener whereas in this course we're using methods like `addEventListener()`.

## WEB EXTENSIONS

Another example of using JavaScript with events is with a technology called WebExtensions. Again, it's kinda similar, but also different. The event model here uses camel-case to attach a listener (e.g. we write `onMessage` rather than `onmessage`). You also need to use an `addListener()` function.

Whats my point?

My point is that although we're focusing on a particular aspect of event listeners (i.e. web events), what you learn in this course can be applied to other technology stacks. Although the way events work is slightly different across different technologies, the concepts you learn here will help you learn and master other environments faster.

So, keep motivated and keep going. It'll pay off.

See you in the next lecture.