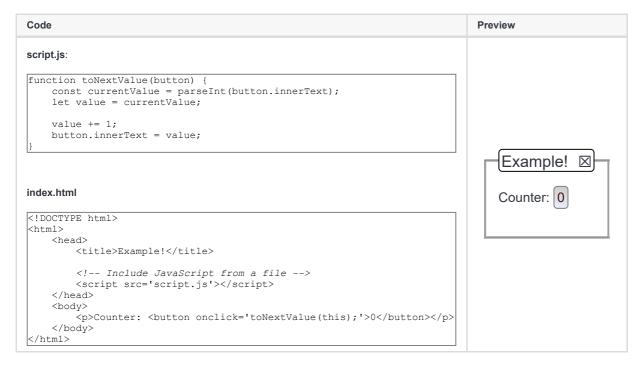
JavaScript "Cheat Sheet"



How do I...

Define a function?

```
function foo(arg1, arg2) {
   console.log("Hello, world."); // Logged to the F12 developer tools.
}
```

or equivalently

```
const foo = (arg1, arg2) => {
    console.log("Hello, world."); // Logged to the F12 developer tools.
};
```

Define a class?

```
class Foo {
    constructor(arg1) {
        // The '#' marks #thing1 as private.
        this.#thing1 = arg1;
    }
    getThing() {
        return this.#thing1;
    }
}
const foo = new Foo(7);
console.log(foo.getThing());
```

Get an HTML element

```
const elem1 = document.getElementById('elementIdHere');
const elem2 = document.querySelector('#elementIdHere');
// At this point, elem1 === elem2
// (=== is 'more equal' than ==. E.g. 0 == '0', but 0 !== '0')
```

Create an HTML element

```
const tagName = 'button';
const button = document.createElement(tagName);
```

Use the just-created element?

```
button.innerHTML = 'This text goes in the button.';
button.onclick = () => {
    // This function gets called when the button is clicked.
};
document.body.appendChild(button);
```

Do something 5 seconds from now?

```
setTimeout(() => {
    doSomething();
}, 5000); // 5000 ms = 5 s
```

Fetch data from a server

```
// async: Lets us `await` for tasks that return a Promise.
async function fetchData() {
    // Fetch data.txt from this server
    const response = await fetch('./data.txt');
    if (response.ok) {
        const text = await response.text();
        return text;
    }
    return null;
}
```

 $\textbf{See} \ \underline{\textbf{https://developer.mozilla.org/en-US/docs/Web/API/Fetch} \ \ \textbf{API/Using} \ \ \textbf{Fetch}}$