Cheat Sheet – Modules & Classes

Modules

ES6 finally, officially, adds Modules to JavaScript. This means, that you may split up your code over multiple files, which of course is a good practice. This is common in ES6 already, however you always require a module loader for that.

Once ES6 is broadly implemented into browsers, this will no longer be necessary.

To split up your code, you basically export variables, functions, objects, … in one file and import it in another:

```
// export.js
export let myExportedVar = 42;


// import.js
import { myExportedVar } from './export.js';
```

More info on Modules (export):
https://developer.mozilla.org/en/docs/web/javascript/reference/statements/export
More info on Modules (import):
https://developer.mozilla.org/en/docs/web/javascript/reference/statements/import
More info on Class: https://developer.mozilla.org/en-US/docs/Web/JavaScript/Reference/Classes

Classes

Classes are now also available via the class keyword. You may of course continue using other ways to create objects, but here's the class-way:

```
class Person {
    greet() {
        this.name = 'Max'; // this is how you set up properties!
        console.log('Hello!');
    }
}
```

```
let person = new Person();
person.greet(); // prints 'Hello!'
```

You may also use inheritance with ES6 classes:

```
class Max extends Person {
    constructor(name) {
        this.name = name;
    }
}
```

Static methods are also possible:

```
class Helper {
    static log(message) {
        console.log(message);
    }
}
```