WEB ADVANCED

JS & JQUERY

WHAT IS JQUERY?

- JavaScript library.

- It makes things like HTML document traversal and manipulation, event handling, animation, and Ajax much simpler

- It uses an easy-to-use API that works across a multitude of browsers.

https://jquery.com/

OTHER RELATED TERMS

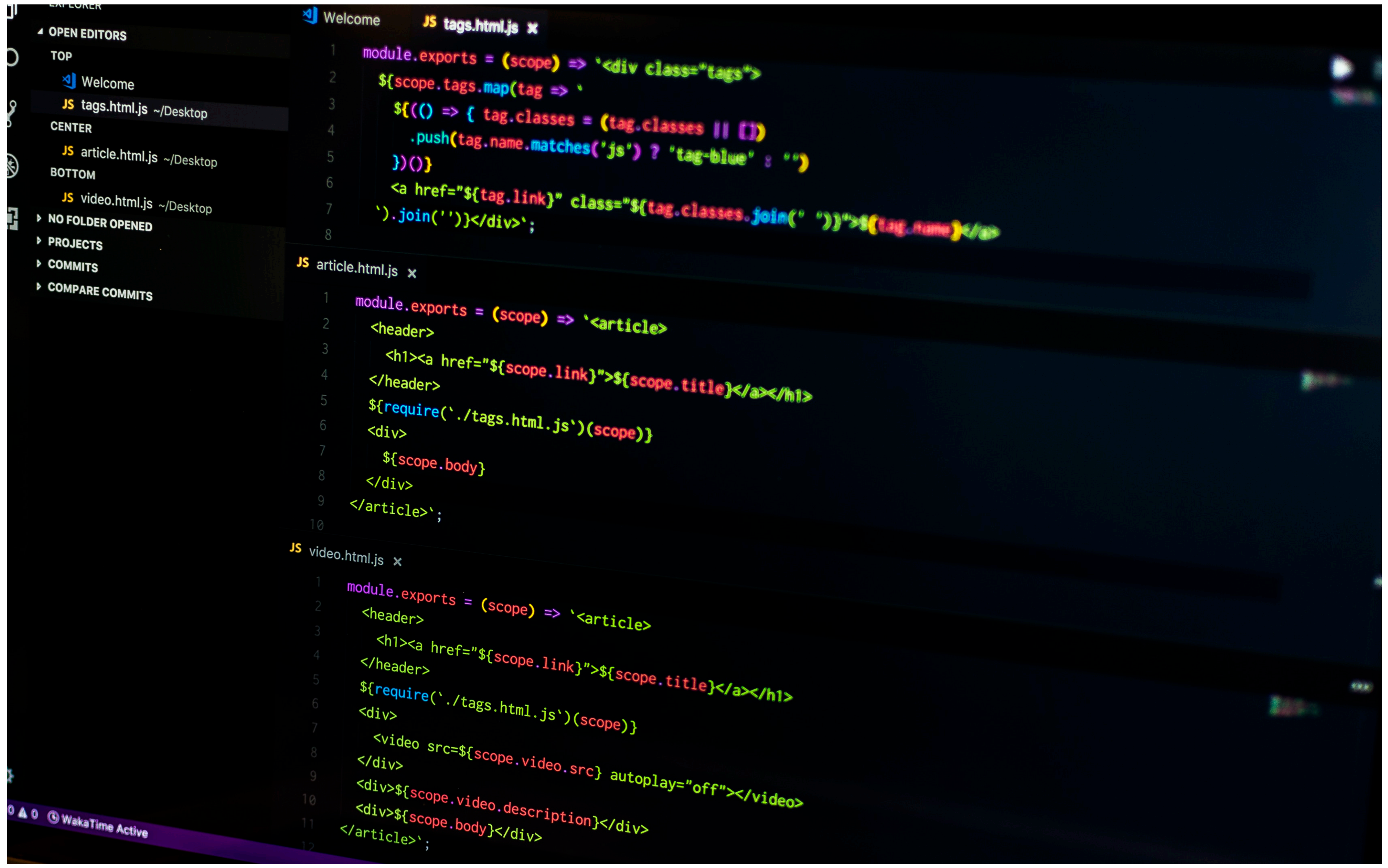
Ajax: Method of exchanging data with a server, and updating parts of a web page, without reloading the entire page.

API : (Application Program Interface) An API specifies how software components should interact. APIs are used when programming graphical user interface (GUI) components.

www.seguetech.com/ajax-technology/
http://www.webopedia.com/TERM/A/API.html

WHY USE JQUERY?

- It helps JavaScript developers when writing code.
- Shorter syntax.



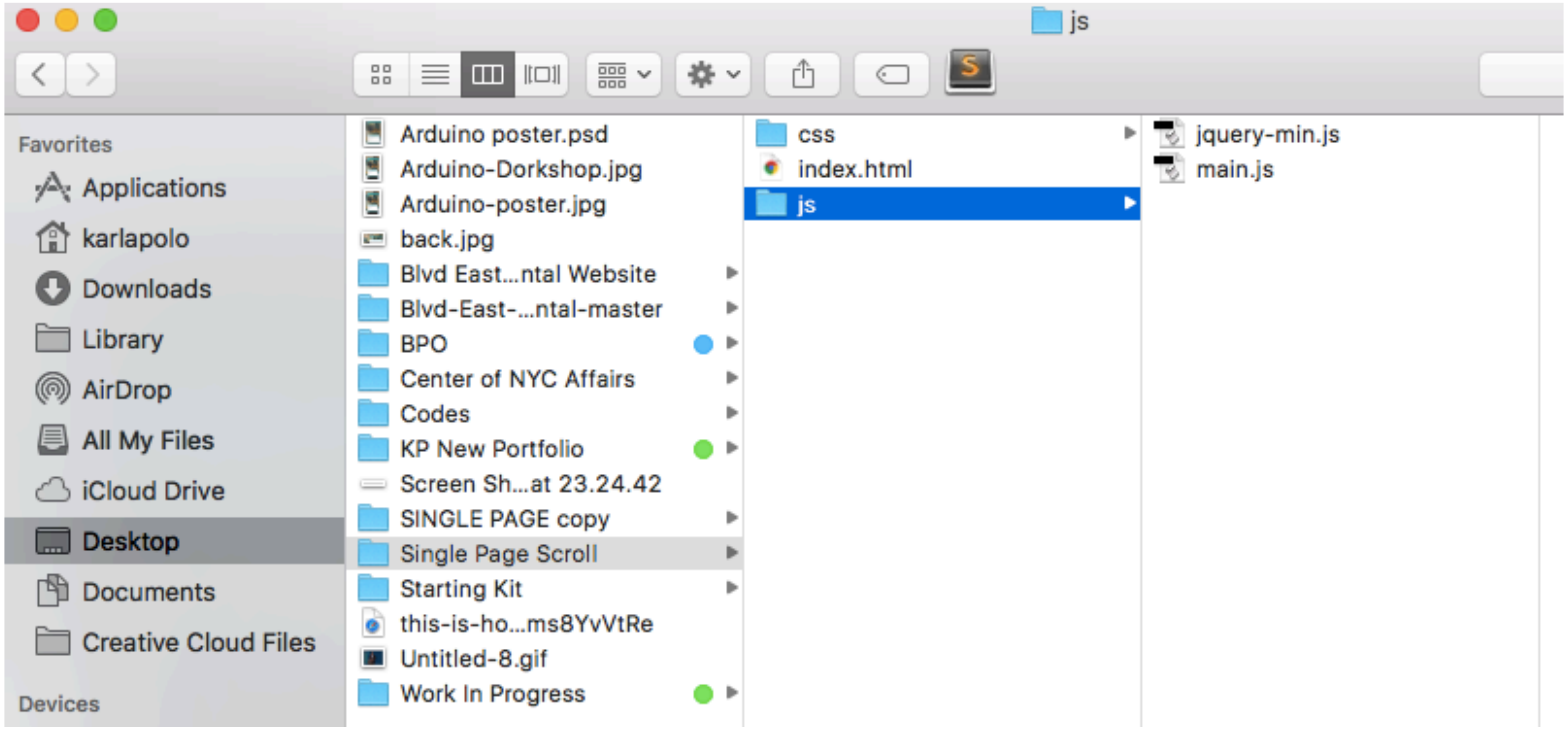
```
module.exports = (scope) => `<div class="tags">
  ${scope.tags.map(tag => `
    ${(() => { tag.classes = (tag.classes || [])
      .push(tag.name.matches('js') ? 'tag-blue' : '')
    })()}
    <a href="${tag.link}" class="${tag.classes.join(' ')}">${tag.name}</a>
  `).join('')}</div>`;
```

```
module.exports = (scope) => `<article>
  <header>
    <h1><a href="${scope.link}">${scope.title}</a></h1>
  </header>
  ${require(`./tags.html.js`)(scope)}
  <div>
    ${scope.body}
  </div>
</article>`;
```

```
module.exports = (scope) => `<article>
  <header>
    <h1><a href="${scope.link}">${scope.title}</a></h1>
  </header>
  ${require(`./tags.html.js`)(scope)}
  <div>
    <video src=${scope.video.src} autoplay="off"></video>
  </div>
  <div>${scope.video.description}</div>
  <div>${scope.body}</div>
</article>`;
```

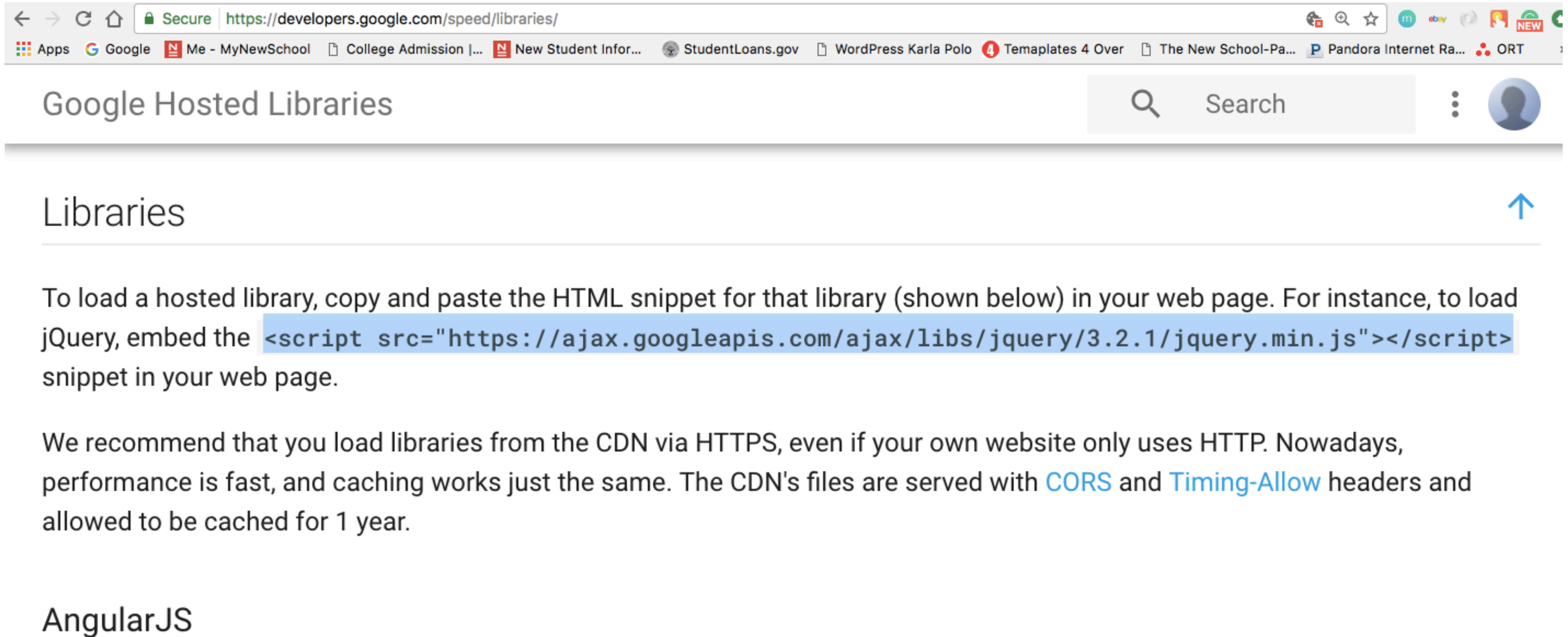
HOW TO INCLUDE JQUERY?

```
<script type="text/javascript" src="javascript/jquery-3.5.1.min.js">
```



HOW TO INCLUDE JQUERY?

```
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.5.1/jquery.min.js"></script>
```



Libraries

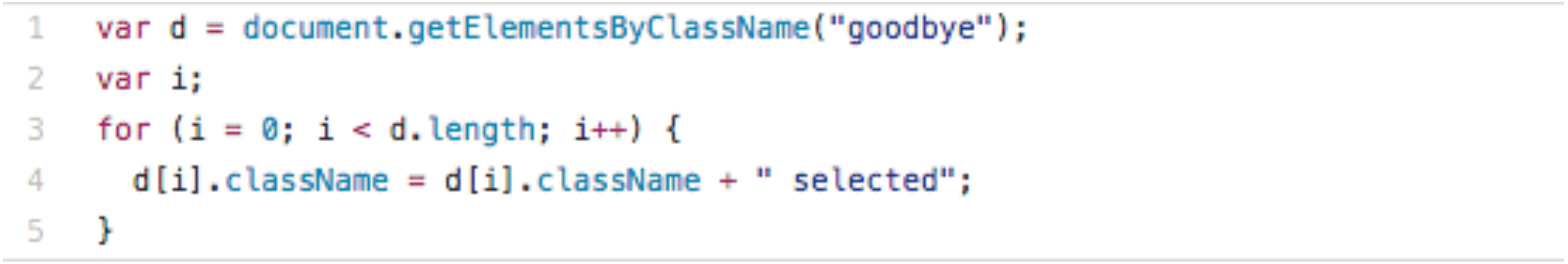
To load a hosted library, copy and paste the HTML snippet for that library (shown below) in your web page. For instance, to load jQuery, embed the `<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.2.1/jquery.min.js"></script>` snippet in your web page.

We recommend that you load libraries from the CDN via HTTPS, even if your own website only uses HTTP. Nowadays, performance is fast, and caching works just the same. The CDN's files are served with CORS and Timing-Allow headers and allowed to be cached for 1 year.

AngularJS

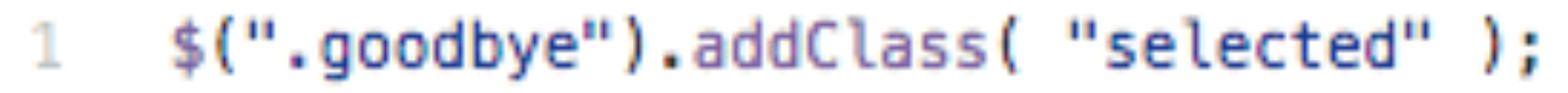
MAIN DIFFERENCES

PURE JS (VANILLA JAVASCRIPT)



```
var d = document.getElementsByClassName("goodbye");
var i;
for (i = 0; i < d.length; i++) {
  d[i].className = d[i].className + " selected";
}
```

JQUERY



```
$(".goodbye").addClass( "selected" );
```

LET'S CODE

WEB ADVANCED

JSON

JS OBJECTS

In JavaScript, almost "everything" is an object.

- Booleans can be objects (if defined with the new keyword)
- Numbers can be objects (if defined with the new keyword)
- Strings can be objects (if defined with the new keyword)
- Dates are always objects
- Maths are always objects
- Regular expressions are always objects
- Arrays are always objects
- Functions are always objects
- Objects are always objects

All JavaScript values, except primitives, are objects.

https://www.w3schools.com/js/js_object_definition.asp

JavaScript defines 5 types of primitive data types:

- string
- number
- boolean
- null
- undefined

JS OBJECT

Name: Bart
Last Name: Simpson
Age: 10

https://www.w3schools.com/js/js_objects.asp

JS OBJECT

Name: Bart
Last Name: Simpson
Age: 10

```
let person  = {
    “name”:“Bart”,
    “lastName”:“Simpson”,
    “age”: 10
}
```

https://www.w3schools.com/js/js_objects.asp

JS OBJECTS

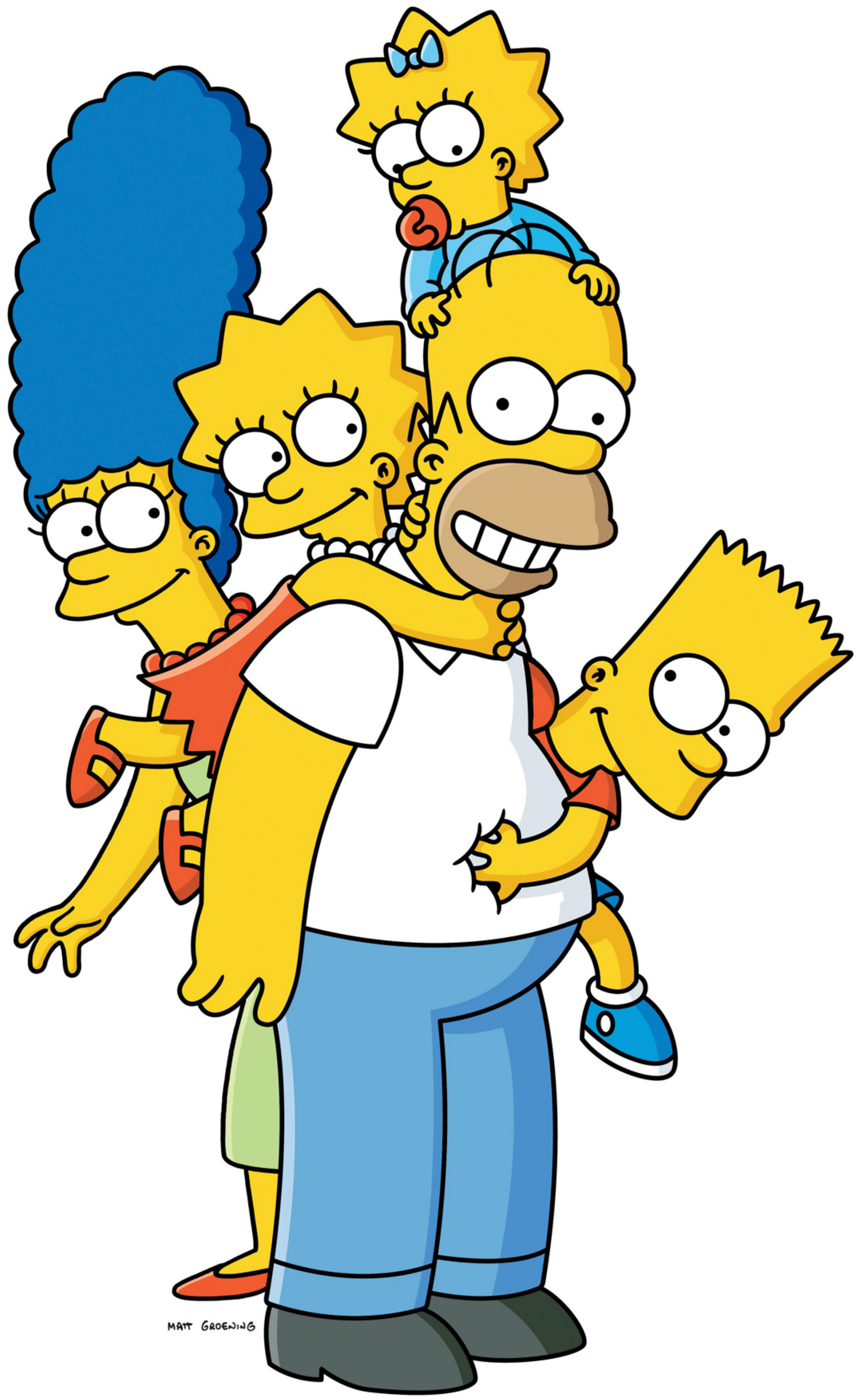
Name: Homer
Last Name: Simpson
Age: 39

Name: Marge
Last Name: Simpson
Age: 36

Name: Bart
Last Name: Simpson
Age: 10

Name: Lisa
Last Name: Simpson
Age: 8

Name: Maggie
Last Name: Simpson
Age: 1



https://www.w3schools.com/js/js_objects.asp

JS OBJECTS

Name: Homer
Last Name: Simpson
Age: 39

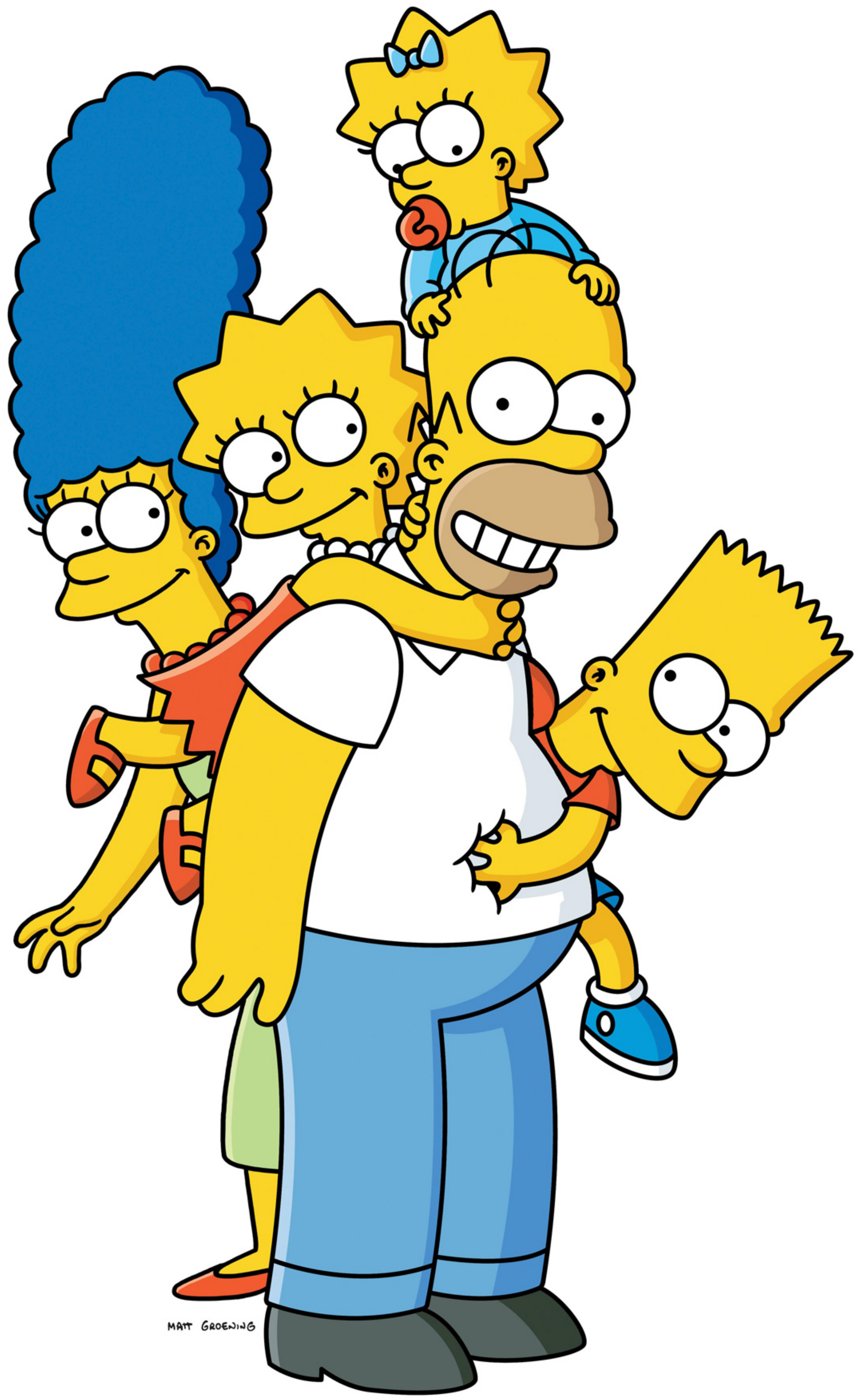
Name: Marge
Last Name: Simpson
Age: 36

Name: Bart
Last Name: Simpson
Age: 10

Name: Lisa
Last Name: Simpson
Age: 8

Name: Maggie
Last Name: Simpson
Age: 1

```
let family  = {“simpsons”:
[ {
        “name”:“Homer”,
        “lastName”:“Simpson”,
        “age”: 39
},
{
        “name”:“Marge”,
        “lastName”:“Simpson”,
        “age”: 36
},
{
        “name”:“Bart”,
        “lastName”:“Simpson”,
        “age”: 10
},
{
        “name”:“Lisa”,
        “lastName”:“Simpson”,
        “age”: 8
},
{
        “name”:“Maggie”,
        “lastName”:“Simpson”,
        “age”: 1
}]}
```



https://www.w3schools.com/js/js_objects.asp

JSON

JSON

JavaScript Object Notation

JSON is a lightweight data-interchange format. It is easy for humans to read and write.

* The JSON syntax is derived from JavaScript object notation syntax, but the JSON format is text only. Code for reading and generating JSON data can be written in any programming language.

https://www.json.org/

HANDLEBARS.JS

HANDLEBARS.JS

Written in JavaScript, Handlebars.js is a compiler that takes any HTML and Handlebars expression and compiles them to a JavaScript function. This derived JavaScript function then takes one parameter, an object—your data—and it returns a string with the HTML and the object properties' values inserted into the HTML.

https://handlebarsjs.com/

HANDLEBARS.JS

Written in JavaScript, Handlebars.js is a compiler that takes any HTML and Handlebars expression and compiles them to a JavaScript function. This derived JavaScript function then takes one parameter, an object—your data—and it returns a string with the HTML and the object properties' values inserted into the HTML.

Allows you to build semantic templates easier

https://handlebarsjs.com/

HANDLEBARS.JS

Installation

DOWNLOAD

Or

<script src="https://cdnjs.cloudflare.com/ajax/libs/handlebars.js/4.7.6/handlebars.min.js"></script>

https://handlebarsjs.com/

HANDLEBARS.JS

HTML

```
<script id="entry-template" type="text/x-handlebars-template">
Your template goes here!
</script>
```

https://handlebarsjs.com/

HANDLEBARS.JS

JS: Compile Template

```
var source   = document.getElementById("entry-template").innerHTML;
var template = Handlebars.compile(source);
```

JS: Execute the template

```
var context = {title: "My New Post", body: "This is my first post!"};
var html    = template(context);
```

https://handlebarsjs.com/

LET'S CODE