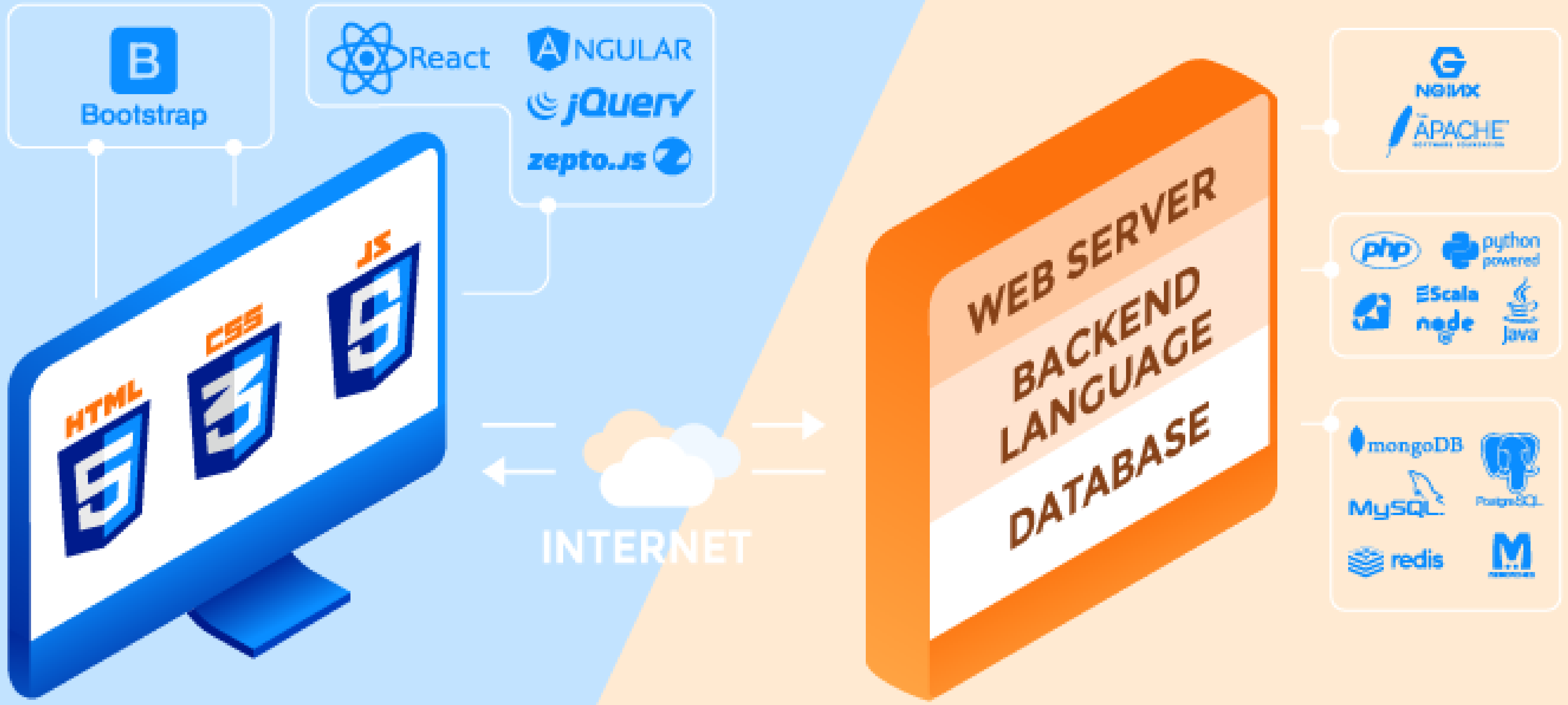
JAVASCRIPT

Feb 5, 2025

AGENDA

- Document Object Model
- Examples
- JavaScript Syntax
- Using Variables
- Using Functions – Type of functions
- Events
- External JavaScript
  - Where should you include the libraries?



CLIENT SIDE

SERVER SIDE

ONE LANGUAGE; TWO PLATFORMS

- Client side – embedded within HTML, executed by browser
- Server side – embedded within server-side scripts, executed by the web server

- We begin with Client-Side scripting . . .

THE HTML DOM (DOCUMENT OBJECT MODEL)

For an HTML page, the browser creates a DOM when the page is loaded

The DOM provides a “glossary” of objects on the page

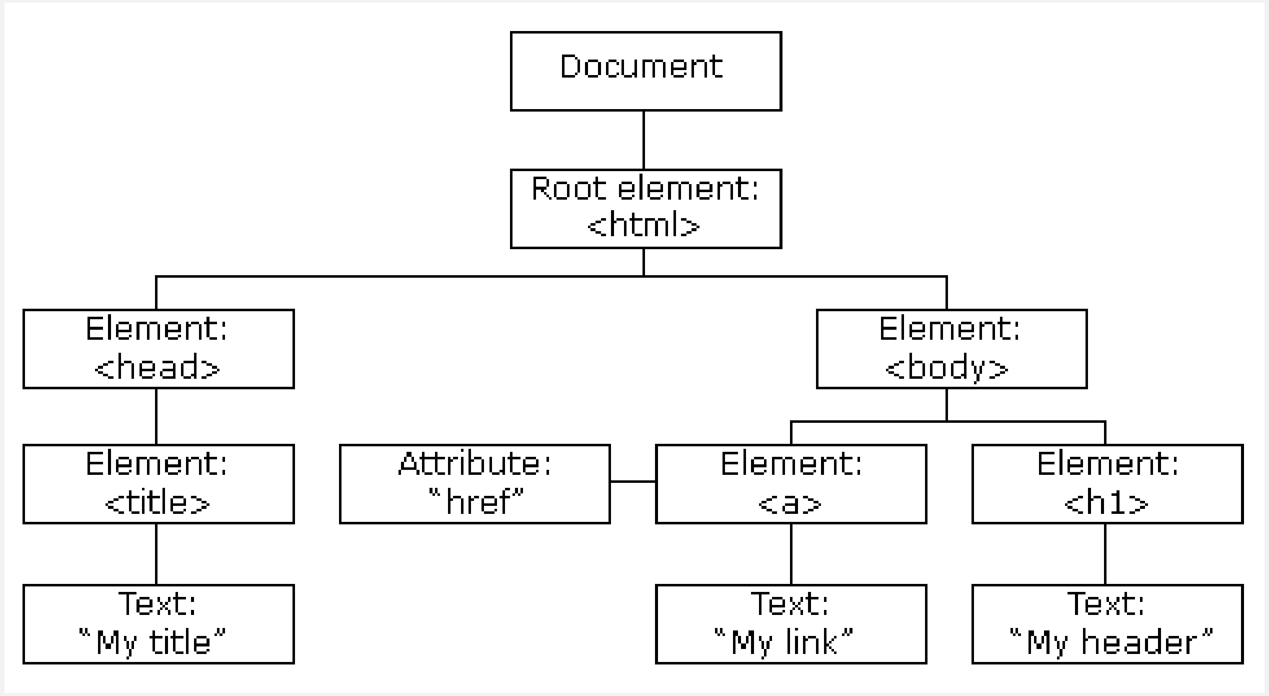
Each <tag> on the page is defined in the DOM

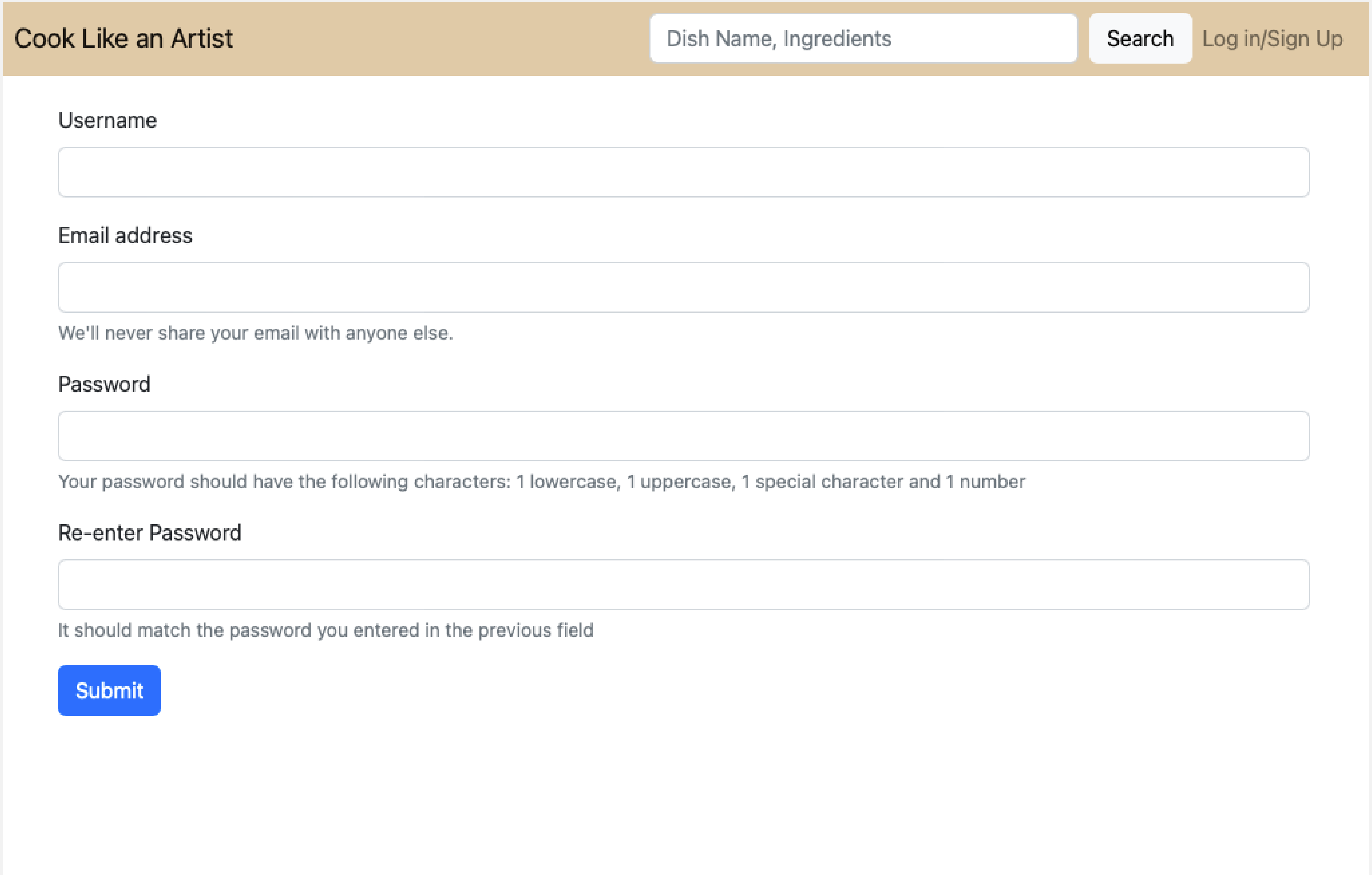
The DOM contains a hierarchy of objects on the page

JavaScript can manipulate objects “real time” as the page is being rendered by the browser.

HTML DOM TREE OF OBJECTS

```
<!DOCTYPE html>
<html>
    <head>
       <title>My Title</title>
    </head>
    <body>
        <a href="">My Link</a>
        <h1>My Header</h1>
    </body>
</html>
```




Cook Like an Artist

Username

Email address

We'll never share your email with anyone else.

Password

Your password should have the following characters: 1 lowercase, 1 uppercase, 1 special character and 1 number

Re-enter Password

It should match the password you entered in the previous field

Practice exercise: Try drawing the DOM for this HTML page on the next slide

THE HTML DOM

- With the object model, JavaScript gets all the power it needs to create dynamic HTML:
  - JavaScript can change all the HTML elements in the page
  - JavaScript can change all the HTML attributes in the page
  - JavaScript can change all the CSS styles in the page
  - JavaScript can remove existing HTML elements and attributes
  - JavaScript can add new HTML elements and attributes
  - JavaScript can react to all existing HTML events in the page
  - JavaScript can create new HTML events in the page

EXAMPLE

Method

Property

```
<html>
  <body>
    <p id="demo"></p>
    <script>
     document.getElementById("demo").innerHTML="My paragraph of
text.";
    </script>
  </body>
</html>
```

https://jsfiddle.net/j3r6gcam/

EXAMPLE 2

```
<html>
  <body>
    <h2>JavaScript Statements</h2>
    <p>A JavaScript program can manipulate the window alert object.</p>
    <script>
      alert("Hello World");
    </script>
  </body>
</html>
```

https://jsfiddle.net/w3hyquzk/

EXAMPLE 3

```
<!DOCTYPE html>
<html>
  <body>
    <h2>What can Javascript Do?</h2>
    <p id="style_demo">Javascript can change the style attribute.</p>
    <button
onclick="document.getElementById('style_demo').style.fontSize='25px';">
         Click to change font size
    </button>
  </body>
</html>
```

https://jsfiddle.net/L3yx9cbd/1/

JAVASCRIPT SYNTAX

- JavaScript code is embedded between a `<script>..</script>` tag set within your HTML file.
- Can be within `<head>` section or the `<body>` section, or both.
- { } encloses a block of code
- ; marks the end of each java script statement
- Functions are identified by ( ) following the function name
  - Pre-defined
  - User-defined

JAVASCRIPT SYNTAX

- You can use single or double quotes
  - “This isn’t fair”
  - ‘He said “This isn\’t fair ” ’
- Variables – Strings or Numeric or Boolean
  - Defined by
    - ES5 - var
    - ES6 -
      - const – immutable variable
      - let – mutable variable
  - Variable name begins with a letter, $, or _
  - Case sensitive
    - “Value” does not equal “value”
  - JS programmers usually use camelCaps

https://www.freecodecamp.org/news/write-less-do-more-with-javascript-es6-5fd4a8e50ee2/

SETTING VARIABLES

```
var firstName = 'Sreesha';
var lastName = 'Nath';
var fullName = firstName + ' ' + lastName
var isInstructor = true;
var score = 0;
score = 50;
score = score + 10;
score += 10; (adds value on right to variable on left)
var name = 'Sreesha';
var message = 'Hey';
message = message + ' ' + name;
message += ' ' + name;
```

CREATING ARRAYS

```
var days = [‘Mon’, ‘Tue’, ‘Wed’, ‘Thu’, ‘Fri’];
alert (days[0]);

document.write(’<p>’);
document.write(days[2]);
document.write(‘</p>’);

var i = 0;
while (i < 5) {
      document.write(’<p>’);
      document.write(days[i]);
      document.write(‘</p>’);
      i += 1;
}
```

FUNCTIONS IN JAVASCRIPT

In-built functions:

```
function random(number) {
  return Math.floor(Math.random()*number);
}
```

User defined function:

```
function myFunction() {
  alert('hello’);
}
```

USING FUNCTIONS

- Step One: Create the Function (often done in the <head> section)

```
function printToday() {
  var today = new Date();
  document.write(today.toDateString());
}
```

USING FUNCTIONS

- Step Two: Call the Function. (in the <body> section)

```
<h1>Using Functions</h1>
<p>Today is
  <script>
    printToday();
  </script>
</p>
```

USING FUNCTIONS

- Step One: add TIME to the Function

```
function printToday() {
  var today = new Date();
  var hours = today.getHours();
  var minutes = today.getMinutes();
  var seconds = today.getSeconds();
  document.write(today.toDateString());
  document.write("   ");
  document.write(hours + ":" + minutes + ":" + seconds);
}
```

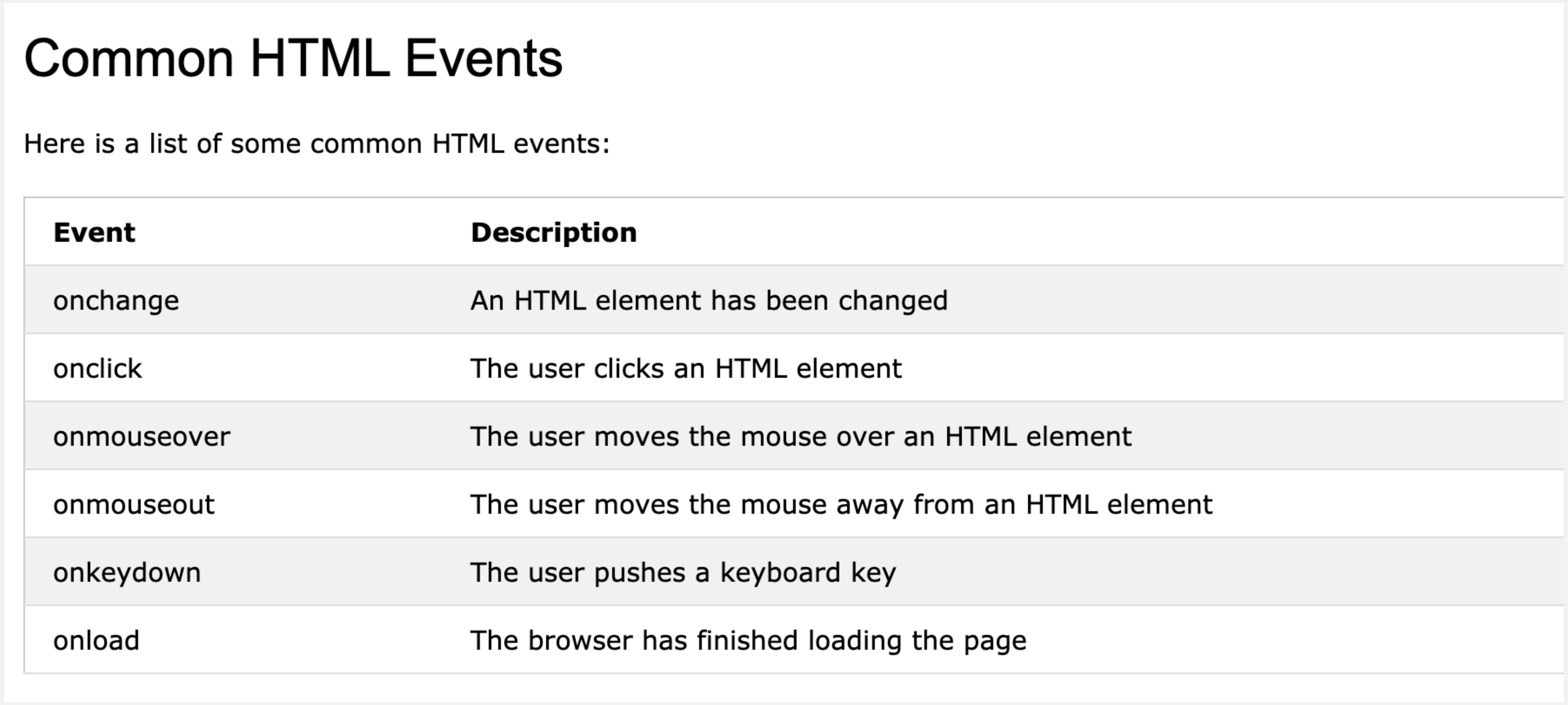
https://jsfiddle.net/xq8rhsj4/

USING FUNCTIONS

- Step Two: Call the function.

```
<h1>Using Functions</h1>
<p>Today is
  <script>
    printToday();
  </script>
</p>
```

JAVASCRIPT CLIENT-SIDE EVENTS



Common HTML Events

Here is a list of some common HTML events:

Event	Description
onchange	An HTML element has been changed
onclick	The user clicks an HTML element
onmouseover	The user moves the mouse over an HTML element
onmouseout	The user moves the mouse away from an HTML element
onkeydown	The user pushes a keyboard key
onload	The browser has finished loading the page

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

EVENTS

- Onclick

```
<html>
  <body>
    <h1 onclick="alert('You clicked this heading');">
          Click Here!
    </h1>
  </body>
</html>
```

https://jsfiddle.net/4scn5yq3/

EVENTS

```
<html>
  <body>
        <div onmouseover="mOver(this);”
    onmouseout="mOut(this);" style="background-
    color:yellow;  width:120px; height:20px;
    padding:40px;">
        Mouse Over the Box
        </div>
        <script>
          function mOver(obj){
             obj.innerHTML = "Moused Over";
          }
          function mOut(obj){
             obj.innerHTML = "Mouse Over This
        Box";
          }
        </script>
  </body>
</html>
```

https://jsfiddle.net/Lf61zt87/

EXTERNAL JAVASCRIPT

- Why?
  - Same JavaScript on several pages in a web site
  - Avoid writing the same script repeatedly.
- Include the external script file exactly where you would have written the script.
- Save the script file with a .js extension, and then refer to it using the src attribute in the <script> tag.
  - `<script src=”URL"></script>`
- URL can be:
  - The URL of the external script file.
  - Possible values:
    - An absolute URL - points to another web site (like `src="http://www.example.com/example.js"`)
    - A relative URL - points to a file within a web site (like `src="/scripts/example.js"`)

JAVASCRIPT – HEAD VS BODY

- `<script>` can be included in both `<head>` and `<body>`
- If you're including JavaScript libraries from external resources, consider including them right before `</body>` within you web page
- Why you should you consider doing so?
  - Prevent slow loading of the remaining resources
  - Your DOM will be completely loaded when your js scripts are loaded
- Potential problems in doing so?
  - External library (like jQuery) dependent page building components will be delayed

GO PLAY!

- http://www.w3schools.com/js/default.asp

QUESTIONS?