ELSEWEDY UNIVERSITY OF TECHNOLOGY
- POLYTECHNIC OF EGYPT -

Web Programming

Lec 5: JavaScript

References

- PHP 8 Basics, 2020, Springer
  - https://link.springer.com/chapter/10.1007/978-1-4842-8082-9_2
- The Absolute Beginner's Guide to HTML and CSS, 2023, Springer
  - https://link.springer.com/chapter/10.1007/978-1-4842-9250-1_7
- W3C Tutorial
  - https://www.w3schools.com/php
  - https://www.w3schools.com/html
  - https://www.w3schools.com/js
- Additional Topics
  - JQuery: https://www.w3schools.com/jquery
  - Bootstrap 5.0: https://www.w3schools.com/bootstrap5
  - Laravel/Blade Framework 11.0: https://www.w3schools.in/laravel

JavaScript

Why Study JavaScript?

JavaScript is one of the **3 languages** all web developers **must** learn:

1. **HTML** to define the content of web pages
2. **CSS** to specify the layout of web pages
3. **JavaScript** to program the behavior of web pages

https://www.youtube.com/playlist?list=PL8q8h6vqfkSVRNnlbUk-O9JJ0c9B7mqCp

JavaScript

JavaScript Objects

Real Life Objects

In real life, **objects** are things like: houses, cars, people, animals, or any other subjects.

Here is a **car object** example:

Car Object	Properties	Methods
	car.name = Fiat	car.start()
	car.model = 500	car.drive()
	car.weight = 850kg	car.brake()
	car.color = white	car.stop()

Important

```
<!DOCTYPE html>
<html>
<body>
<h1>Creating JavaScript Objects</h1>
<h2>Using an Object Literal</h2>

<p id="demo"></p>

<script>
// Create an Object:
const person = {
  firstName: "John",
  lastName: "Doe",
  age: 50,
  eyeColor: "blue"
};

// Display Data from the Object:
document.getElementById("demo").innerHTML =
person.firstName + " is " + person.age + " years old.";
</script>

</body>
</html>
```

Creating JavaScript Objects

Using an Object Literal

John is 50 years old.

```
<!DOCTYPE html>
<html>
<body>
<h1>Creating JavaScript Objects</h1>
<h2>Using the new Keyword</h2>

<p id="demo"></p>

<script>
// Create an Object
const person = new Object();
person.firstName = "John";
person.lastName = "Doe";
person.age = 50;
person.eyeColor = "blue";

// Diplay Object Content
document.getElementById("demo").innerHTML =
person.firstName + " is " + person.age + " years old.";
</script>

</body>
</html>
```

Creating JavaScript Objects

Using the new Keyword

John is 50 years old.

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Objects</h1>
<h2>Object Methods</h2>
<p>A method is a function definition stored as a property value.</p>

<p id="demo"></p>

<script>
const person = {
  firstName: "John",
  lastName: "Doe",
  id: 5566,
  fullName: function() {
    return this.firstName + " " + this.lastName;
  }
};

document.getElementById("demo").innerHTML = person.fullName();
</script>
</body>
</html>
```

JavaScript Objects

Object Methods

A method is a function definition stored as a property value.

John Doe

```html
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Object Properties</h1>
<h2>Deleting a Property</h2>

<p id="demo"></p>

<script>
const person = {
  firstname: "John",
  lastname: "Doe",
  age: 50,
  eyecolor: "blue"
};

delete person.age;

document.getElementById("demo").innerHTML =
person.firstname + " is " + person.age + " years old.";
</script>

</body>
</html>
```

JavaScript Object Properties

Deleting a Property

John is undefined years old.

```html
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Objects</h1>
<h2>Accessing Nested Objects</h2>

<p id="demo"></p>

<script>
// Create nested Objects
const myObj = {
  name: "John",
  age: 30,
  myCars: {
    car1: "Ford",
    car2: "BMW",
    car3: "Fiat"
  }
}

let p1 = "myCars";
let p2 = "car2";
document.getElementById("demo").innerHTML = myObj[p1][p2];
</script>

</body>
</html>
```

Nested Objects

```html
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Objects</h1>
<h2>Adding a Method</h2>

<p id="demo"></p>

<script>
// Create an Object
const person = {
  firstName: "John",
  lastName: "Doe",
  id: 5566,
};

// Add a Method
person.name = function() {
  return this.firstName + " " + this.lastName;
};

// Display Object Data
document.getElementById("demo").innerHTML =
"My father is " + person.name();
</script>

</body>
</html>
```

JavaScript Objects

Adding a Method

My father is John Doe

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Objects</h1>
<h2>The Object.values() Method</h2>
<p>Object.values() returns an array of values from an object:</p>

<p id="demo"></p>

<script>
// Create an Object
const person = {
  name: "John",
  age: 30,
  city: "New York"
};

// Create an Array
const myArray = Object.values(person);

// Display the Array
document.getElementById("demo").innerHTML = myArray;
</script>

</body>
</html>
```

Important

JavaScript Objects

The Object.values() Method

Object.values() returns an array of values from an object:

John,30,New York

```html
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Object Constructors</h1>

<p id="demo"></p>

<script>

// Constructor function for Person objects
function Person(first, last, age, eye) {
  this.firstName = first;
  this.lastName = last;
  this.age = age;
  this.eyeColor = eye;
}

// Create two Person objects
const myFather = new Person("John", "Doe", 50, "blue");
const myMother = new Person("Sally", "Rally", 48, "green");

// Display age
document.getElementById("demo").innerHTML =
"My father is " + myFather.age + ". My mother is " + myMother.age + ".";

</script>

</body>
</html>
```

JavaScript

JavaScript Events

HTML Events

An HTML event can be something the browser does, or something a user does.

Here are some examples of HTML events:

•An HTML web page has finished loading.
•An HTML input field was changed.
•An HTML button was clicked.

JavaScript

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript HTML Events</h1>
<h2>The onclick Attribute</h2>

<p>Click the button to display the date.</p>
<button onclick="displayDate()">The time is?</button>

<script>
function displayDate() {
  document.getElementById("demo").innerHTML = Date();
}
</script>

<p id="demo"></p>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript if .. else</h2>

<p>A time-based greeting:</p>

<p id="demo"></p>

<script>
const time = new Date().getHours();
let greeting;
if (time < 10) {
  greeting = "Good morning";
} else if (time < 20) {
  greeting = "Good day";
} else {
  greeting = "Good evening";
}
document.getElementById("demo").innerHTML = greeting;
</script>

</body>
</html>
```

JavaScript if .. else

A time-based greeting:

Good evening

```html
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript switch</h2>

<p id="demo"></p>

<script>
let text;
switch (new Date().getDay()) {
  case 6:
    text = "Today is Saturday";
    break;
  case 0:
    text = "Today is Sunday";
    break;
  default:
    text = "Looking forward to the Weekend";
}
document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

JavaScript switch

Today is Saturday

```html
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript For Loop</h2>

<p id="demo"></p>

<script>
let text = "";

for (let i = 0; i < 5; i++) {
  text += "The number is " + i + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

Important

JavaScr

The For In Loop

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript For In Loop</h2>
<p>The for in statement loops through the properties of an object:</p>

<p id="demo"></p>

<script>
const person = {fname:"John", lname:"Doe", age:25};

let txt = "";
for (let x in person) {
  txt += person[x] + " ";
}

document.getElementById("demo").innerHTML = txt;
</script>

</body>
</html>
```

JavaScript For In Loop

The for in statement loops through the properties of an object:

John Doe 25

- The **for in** loop iterates over a **person** object
- Each iteration returns a **key** (x)
- The key is used to access the **value** of the key
- The value of the key is **person[x]**

Important

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Arrays</h1>
<h2>The forEach() Method</h2>

<p>Call a function once for each array element:</p>

<p id="demo"></p>

<script>
const numbers = [45, 4, 9, 16, 25];

let txt = "";
numbers.forEach(myFunction);
document.getElementById("demo").innerHTML = txt;

function myFunction(value, index, array) {
  txt += value + "<br>";
}
</script>

</body>
</html>
```

JavaScript Arrays

The forEach() Method

Call a function once for each array element

45
4
9
16
25

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Arrays</h1>
<h2>The forEach() Method</h2>

<p>Call a function once for each array element:</p>

<p id="demo"></p>

<script>
const numbers = [45, 4, 9, 16, 25];

let txt = "";
numbers.forEach(myFunction);
document.getElementById("demo").innerHTML = txt;

function myFunction(value) {
  txt += value + "<br>";
}
</script>

</body>
</html>
```

JavaScript

The Do While Loop

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Do While Loop</h2>

<p id="demo"></p>

<script>
let text = ""
let i = 0;

do {
  text += "<br>The number is " + i;
  i++;
}
while (i < 10);

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

JavaScript Do While Loop

The number is 0
The number is 1
The number is 2
The number is 3
The number is 4
The number is 5
The number is 6
The number is 7
The number is 8
The number is 9

Comparing For and While

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
const cars = ["BMW", "Volvo", "Saab", "Ford"];

let i = 0;
let text = "";
for (;cars[i];) {
  text += cars[i] + "<br>";
  i++;
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

```
<!DOCTYPE html>
<html>
<body>

<p id="demo"></p>

<script>
const cars = ["BMW", "Volvo", "Saab", "Ford"];

let i = 0;
let text = "";
while (cars[i]) {
  text += cars[i] + "<br>";
  i++;
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

```html
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Loops</h2>

<p>A loop with a <b>break</b> statement.</p>

<p id="demo"></p>

<script>
let text = "";
for (let i = 0; i < 10; i++) {
  if (i === 3) { break; }
  text += "The number is " + i + "<br>";
}

document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

JavaScript Loops

A loop with a **break** statement.

The number is 0
The number is 1
The number is 2

```html
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript Loops</h2>

<p>A loop with a <b>continue</b> statement.</p>

<p>A loop which will skip the step where i = 3.</p>

<p id="demo"></p>

<script>
let text = "";
for (let i = 0; i < 10; i++) {
  if (i === 3) { continue; }
  text += "The number is " + i + "<br>";
}
document.getElementById("demo").innerHTML = text;
</script>

</body>
</html>
```

Important

JavaScript

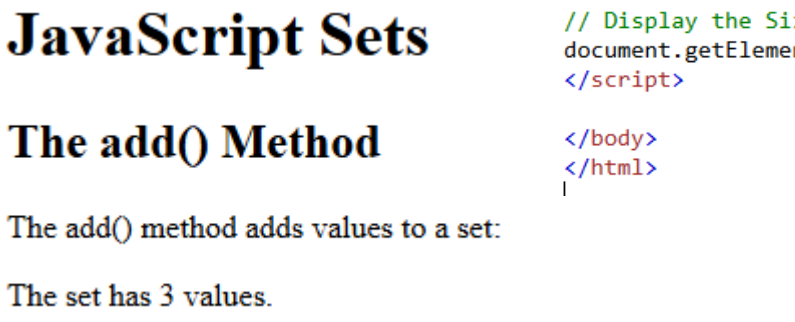
JavaScript Sets

How to Create a Set

You can create a JavaScript Set by:

- Passing an array to `new Set()`
- Create an empty set and use `add()` to add values

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Sets</h1>
<h2>The add() Method</h2>

<p>The add() method adds values to a set:</p>

<p id="demo"></p>

<script>
// Create a Set
const letters = new Set();

// Add Values to the Set
letters.add("a");
letters.add("b");
letters.add("c");

// Display the Size
document.getElementById("demo").innerHTML = "The set has " + letters.size + " values.";
</script>

</body>
</html>
```



JavaScript Classes

Important

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Class Methods</h1>
<p>How to define and use a Class method.</p>

<p id="demo"></p>

<script>
class Car {
  constructor(name, year) {
    this.name = name;
    this.year = year;

  }
  age() {
    const date = new Date();
    return date.getFullYear() - this.year;
  }
}

const myCar = new Car("Ford", 2014);
document.getElementById("demo").innerHTML =
"My car is " + myCar.age() + " years old.";
</script>

</body>
</html>
```

JavaScript Class Methods

How to define and use a Class method.

My car is 10 years old.

Important

```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Class Inheritance</h1>

<p>Use the "extends" keyword to inherit all methods from another class.</p>
<p>Use the "super" method to call the parent's constructor function.</p>

<p id="demo"></p>

<script>
class Car {
  constructor(brand) {
    this.carname = brand;
  }
  present() {
    return 'I have a ' + this.carname;
  }
}

class Model extends Car {
  constructor(brand, mod) {
    super(brand);
    this.model = mod;
  }
  show() {
    return this.present() + ', it is a ' + this.model;
  }
}

const myCar = new Model("Ford", "Mustang");
document.getElementById("demo").innerHTML = myCar.show();
</script>

</body>
</html>
```

JavaScript Class Inheritance

Use the "extends" keyword to inherit all methods from another class.

Use the "super" method to call the parent's constructor function.

I have a Ford, it is a Mustang

```html
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript Class Static Methods</h1>
<p>A static method is created with the "static" keyword,
and you can only call the method on the class itself.</p>

<p id="demo"></p>

<script>
class Car {
  constructor(name) {
    this.name = name;
  }
  static hello() {
    return "Hello!!";
  }
}

const myCar = new Car("Ford");

//You can call 'hello()' on the Car Class:
document.getElementById("demo").innerHTML = Car.hello();

// But NOT on  a Car Object:
// document.getElementById("demo").innerHTML = myCar.hello();
// this will raise an error.
</script>

</body>
</html>
```

Important

JavaScript Class Static Methods

A static method is created with the "static" keyword, and you can only call the method on the class itself.

Hello!!

JavaScript

JS HTML DOM

The HTML DOM (Document Object Model)

The HTML DOM is a standard for how to get, change, add, or delete HTML elements

Finding HTML Element by Id

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>

<p id="intro">Finding HTML Elements by Id</p>
<p>This example demonstrates the <b>getElementsById</b> method.</p>

<p id="demo"></p>

<script>
const element = document.getElementById("intro");

document.getElementById("demo").innerHTML =
"The text from the intro paragraph is: " + element.inn

</script>

</body>
</html>
```

JavaScript HTML DOM

Finding HTML Elements by Id

This example demonstrates the **getElementsById** method.

The text from the intro paragraph is: Finding HTML Elements by Id

JavaScript

Finding HTML Elements by Tag Name

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>

<p>Finding HTML Elements by Tag Name.</p>
<p>This example demonstrates the <b>getElementsByTagName</b> method.</p>

<p id="demo"></p>

<script>
const element = document.getElementsByTagName("p");

document.getElementById("demo").innerHTML = 'The text in first paragraph (index 0) is: '

+ element[0].innerHTML;

</script>

</body>
</html>
```

JavaScript HTML DOM

Finding HTML Elements by Tag Name.

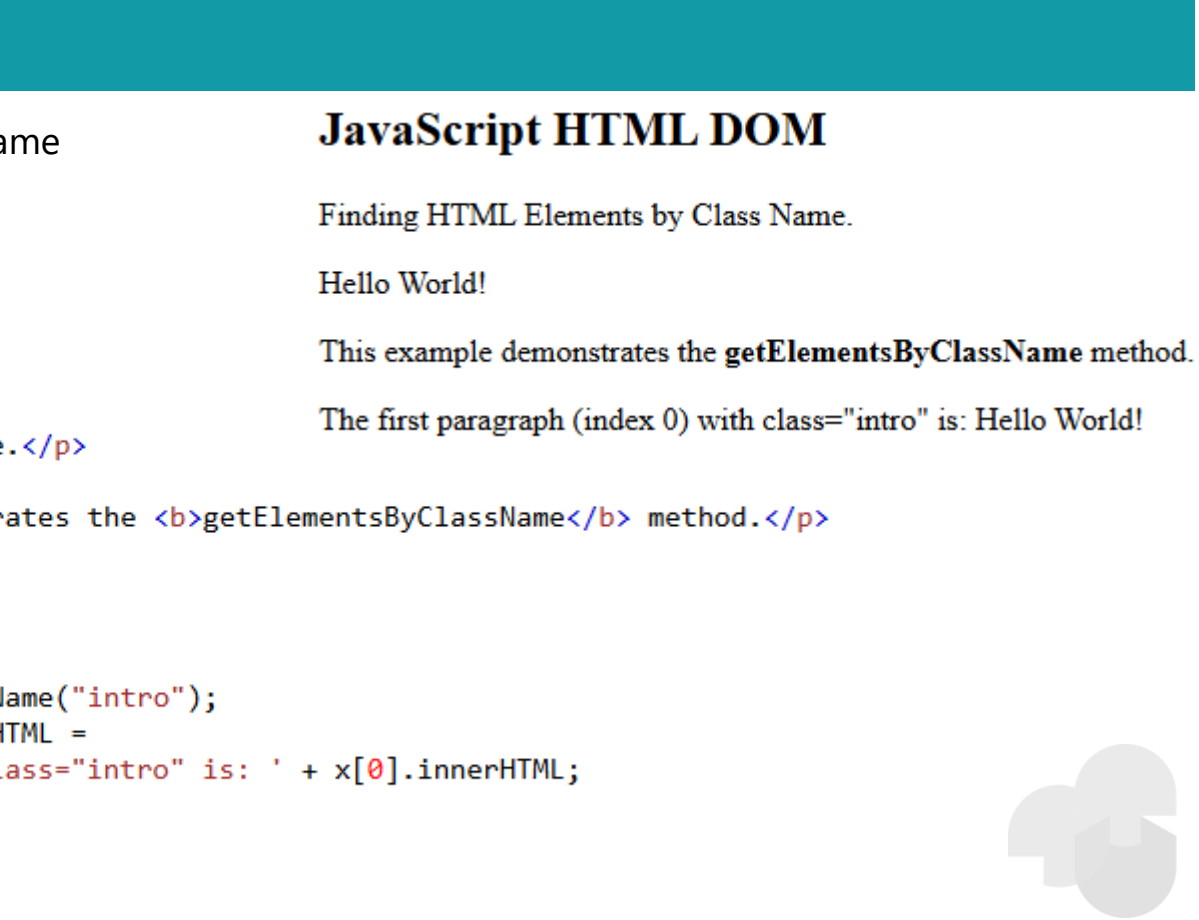
This example demonstrates the **getElementsByTagName** method.

The text in first paragraph (index 0) is: Finding HTML Elements by Tag Name.

JavaScript

Finding HTML Elements by Class Name

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>

<p>Finding HTML Elements by Class Name.</p>
<p class="intro">Hello World!</p>
<p class="intro">This example demonstrates the <b>getElementsByClassName</b> method.</p>

<p id="demo"></p>

<script>
const x = document.getElementsByClassName("intro");
document.getElementById("demo").innerHTML =
'The first paragraph (index 0) with class="intro" is: ' + x[0].innerHTML;
</script>

</body>
</html>
```



JavaScript

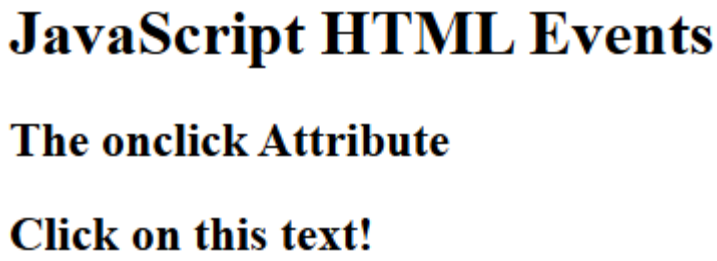
Finding HTML Elements by CSS Selectors

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>

<p>Finding HTML Elements by Query Selector</p>
<p class="intro">Hello World!.</p>
<p class="intro">This example demonstrates the <b>querySelectorAll</b> method.</p>

<p id="demo"></p>

<script>
const x = document.querySelectorAll("p.intro");
document.getElementById("demo").innerHTML =
'The first paragraph (index 0) with class="intro" is: ' + x[0].innerHTML;
</script>

</body>
</html>
```

JavaScript

JavaScript HTML DOM Events

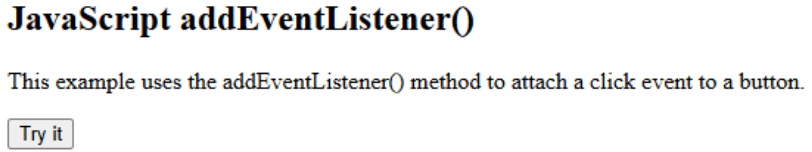
```
<!DOCTYPE html>
<html>
<body>
<h1>JavaScript HTML Events</h1>
<h2>The onclick Attribute</h2>

<h2 onclick="this.innerHTML='Ooops!'">Click on this text!</h2>

</body>
</html>
```



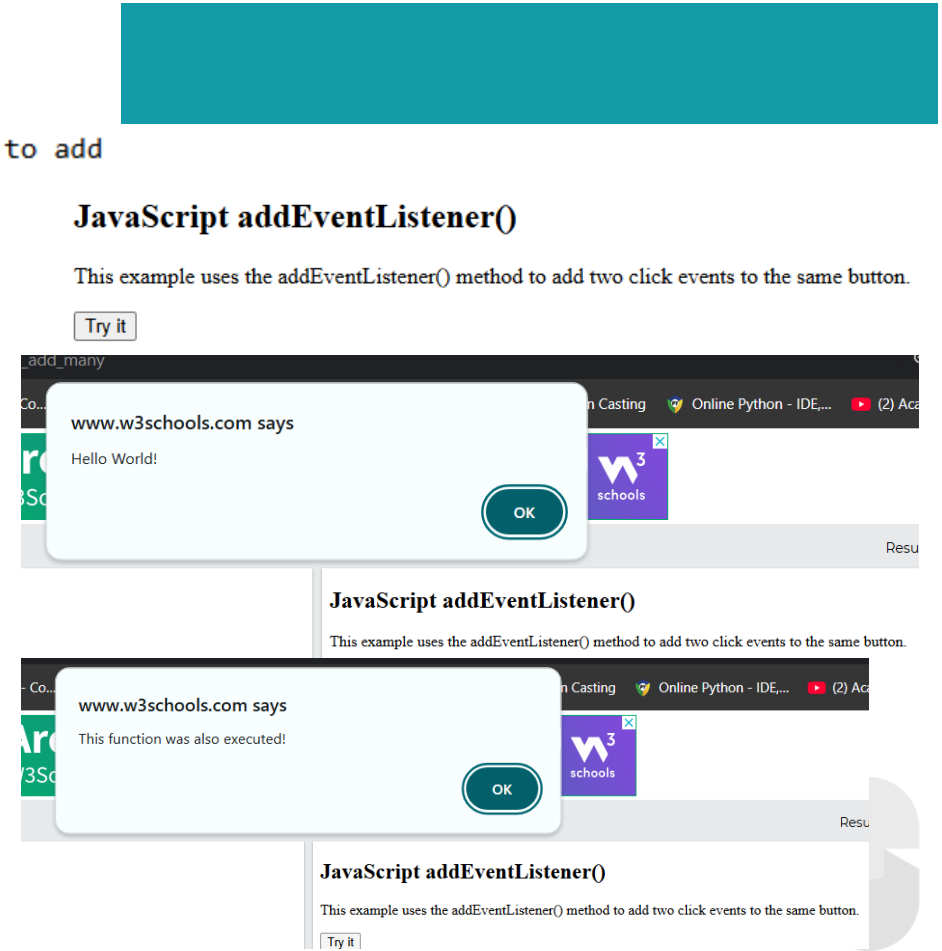
JavaScript

JavaScript HTML DOM EventListener

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript addEventListener()</h2>

<p>This example uses the addEventListener() method to attach a click event to a button.</p>

<button id="myBtn">Try it</button>

<p id="demo"></p>

<script>
document.getElementById("myBtn").addEventListener("click", displayDate);

function displayDate() {
  document.getElementById("demo").innerHTML = Date();
}
</script>

</body>
</html>
```



```html
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript addEventListener()</h2>

<p>This example uses the addEventListener() method to add
two click events to the same button.</p>

<button id="myBtn">Try it</button>

<script>
var x = document.getElementById("myBtn");
x.addEventListener("click", myFunction);
x.addEventListener("click", someOtherFunction);

function myFunction() {
  alert ("Hello World!");
}

function someOtherFunction() {
  alert ("This function was also executed!");
}
</script>

</body>
</html>
```



JavaScript

JavaScript HTML DOM - Changing CSS

```
<!DOCTYPE html>
<html>
<body>

<h2>JavaScript HTML DOM</h2>
<p>Changing the HTML style:</p>


<p id="p1">Hello World!</p>
<p id="p2">Hello World!</p>

<script>
document.getElementById("p2").style.color = "blue";
document.getElementById("p2").style.fontFamily = "Arial";
document.getElementById("p2").style.fontSize = "larger";
</script>


</body>
</html>
```