



University of Tripoli
Faculty of Information Technology



Department of Software Engineering

JavaScript Basics

Introduction to Internet Programming
ITGS 226 -- F2023

By: Fatima Ben Lashihar

JavaScript Basics Introduction to Internet Programming

أساسيات جافا سكريبت مقدمة لبرمجة الإنترنت

ITGS 226 --F2023 By: Fatima Ben Lashihar

ITGS 226--F2023 بقلم: فاطمة بن لشيهار

صفحة (1) | تُرجمت بواسطة @xFxBot

INTRODUCTION

- JavaScript is the most popular and widely used client-side scripting language.
- JavaScript is designed to add interactivity and dynamic effects to the web pages by manipulating the content returned from a web server.
- JavaScript is an object-oriented language, and it also has some similarities in syntax to Java programming language. But, JavaScript is not related to Java in any way.

INTRODUCTION

مقدمة

- JavaScript is the most popular and widely used client-side scripting language.
• JavaScript هي لغة البرمجة النصية الأكثر شيوعًا والأكثر استخدامًا من جانب العميل.
- JavaScript is designed to add interactivity and dynamic effects to the web pages by manipulating the content returned from a web server.
• تم تصميم JavaScript لإضافة تأثيرات تفاعلية وديناميكية إلى صفحات الويب عن طريق معالجة المحتوى الذي يتم إرجاعه من خادم الويب.
- JavaScript is an object-oriented language, and it also has some similarities in syntax to Java programming language.
• جافا سكريبت هي لغة كائنية التوجه، ولها أيضًا بعض أوجه التشابه في بناء الجملة مع لغة برمجة جافا.

But, JavaScript is not related to Java in any way.

لكن جافا سكريبت لا علاقة لها بجافا بأي شكل من الأشكال.

What can do with Javascript

- Modify the content of a web page by adding or removing elements
([Example](#))
- Change the style and position of the elements on a web page
([Example](#))
- Monitor events like mouse click, hover, etc. and react to it ([Example](#))
- Create alert pop-ups to display info or warning messages to the user
([Example](#))
- Validate user inputs before submitting it to the server ([Example](#))

What can do with Javascript

ما يمكن القيام به مع جافا سكريبت

- Modify the content of a web page by adding or removing elements (Example)

• تعديل محتوى صفحة الويب عن طريق إضافة أو إزالة العناصر (مثال)

- Change the style and position of the elements on a web page (Example)

• تغيير نمط وموضع العناصر على صفحة الويب (مثال)

- Monitor events like mouse click, hover, etc.

• رصد الأحداث مثل النقر بالماوس، والتمرير، وما إلى ذلك.

and react to it (Example)

والرد عليه (مثال)

- Create alert pop-ups to display info or warning messages to the user (Example)

• إنشاء تنبيهات منبثقة لعرض المعلومات أو رسائل التحذير للمستخدم (مثال)

صفحة (3) | تُرجمت بواسطة @xFxBot

- Validate user inputs before submitting it to the server (Example)

• التحقق من صحة مدخلات المستخدم قبل إرسالها إلى الخادم (مثال)

صفحة (3) | تُرجمت بواسطة @xFxBot

ADDING JAVASCRIPT TO WEB PAGES

- There are typically three ways to add JavaScript to a web page:
 1. Embedding (internal) the JavaScript code between a pair of `<script>` and `</script>` tag.
 2. Creating an external JavaScript file with the `.js` extension and then load it within the page through the `src` attribute of the `<script>` tag.
 3. Placing the JavaScript code directly inside an HTML tag using the special tag attributes such as `onclick`, `onmouseover`, `onkeypress`, `onload`, etc.

ADDING JAVASCRIPT TO WEB PAGES

إضافة جافا سكريبت إلى صفحات الويب

- There are typically three ways to add JavaScript to a web page:

• توجد عادةً ثلاث طرق لإضافة JavaScript إلى صفحة الويب:

Embedding (internal) the JavaScript code between a pair of tag.

تضمين (داخليًا) كود JavaScript بين زوج من العلامات .

Creating an external JavaScript file with the .js extension and then load it within the page through the src attribute of the

Placing the JavaScript code directly inside an HTML tag

وضع كود JavaScript مباشرةً داخل علامة HTML

using

استخدام

the

ال

special

خاص

tag

بطاقة شعار

attributes

صفات

such

هذه

as onclick, onmouseover, onkeypress, onload, etc.

مثل onclick و onmouseover و onkeypress و onload وما إلى ذلك.

EMBEDDING THE JAVASCRIPT CODE

- Embedding the JavaScript code directly within web pages by placing it between the **<script>** and **</script>** tags.
- The **<script>** tag indicates the browser that the contained statements are to be interpreted as executable script and not HTML ([Example1](#)) ([Example2](#))

```
<body>
  <script>
    var greet = "Hello World!";
    document.write(greet);      // Prints: Hello World!
  </script>
</body>
```

EMBEDDING THE JAVASCRIPT CODE

تضمين كود جافا سكريبت

- Embedding the JavaScript code directly within web pages by placing it between the tags.
- تضمين كود JavaScript مباشرة داخل صفحات الويب عن طريق وضعه بين علامتي .
- The

```
document.write(greet); // Prints: Hello World!
```

```
;document.write(greet) // المطبوعات: أهلاً بالعالم!
```

CALLING AN EXTERNAL JAVASCRIPT FILE

- Placing JavaScript code into a separate file with a (.js) extension, and then call that file in your document through the **src** attribute of the `<script>` tag, like this:

```
<script src="js/hello.js"></script> (Example)
```

- This is useful if aiming to have the same scripts available to multiple documents.

CALLING AN EXTERNAL JAVASCRIPT FILE

استدعاء ملف جافا سكريبت خارجي

• Placing

• التسيب

JavaScript

جافا سكريبت

code

شفرة

into

داخل

separate

متفرق

file

ملف

with

مع

a (.js) extension, and then call that file in your document through the src attribute of the

(Example)

(مثال)

- This is useful if aiming to have the same scripts available to multiple documents.

• يعد هذا مفيدًا إذا كنت تهدف إلى توفير نفس النصوص البرمجية لمستندات متعددة.

PLACING THE JAVASCRIPT CODE INLINE

- Placing JavaScript code inline by inserting it directly inside the HTML tag using the special tag attributes such as onclick, onmouseover, onkeypress, onload, etc.

```
<body>  
  <button type= "button" onclick="alert('Hello World!') ">  
    Click Me  
  </button>  
</body>
```

- Always keep the content and structure of web page (i.e. HTML) separate out from presentation (CSS), and behavior (JavaScript).

PLACING THE JAVASCRIPT CODE INLINE

وضع كود جافا سكريبت في السطر

- Placing JavaScript code inline by inserting it directly inside the

• وضع كود جافا سكريبت في السطر عن طريق إدراجه مباشرة داخل

HTML

لغة البرمجة

tag

بطاقة شعار

using

استخدام

the

ال

special

خاص

attributes

صفات

such

هذه

as onclick, onmouseover, onkeypress, onload, etc.

مثل onclick و onmouseover و onkeypress و onload وما إلى ذلك.

'> Click Me

('< انقر فوق

• Always keep the content and structure of web page (i.e.

• احتفظ دائمًا بمحتوى صفحة الويب وبنيتها (أي:

HTML)

لغة البرمجة)

separate out from presentation (CSS), and behavior (JavaScript).

منفصلة عن العرض (CSS)، والسلوك (جافا سكريبت).

صفحة (7) | تُرجمت بواسطة @xFxBot

POSITIONING OF SCRIPT INSIDE HTML DOCUMENT

- The `<script>` element can be placed in the `<head>`, or `<body>` section of an HTML document. **But ideally, scripts should be placed at the end of the body section, just before the closing `</body>` tag, it will make your web pages load faster, since it prevents obstruction of initial page rendering.**

POSITIONING OF SCRIPT INSIDE HTML DOCUMENT

تحديد موضع النص داخل مستند HTML

• The

• ال

element

عنصر

can

يستطيع

placed

وضعت

the

ال

or section of an HTML document.

أو قسم من مستند HTML.

But ideally, scripts should be placed at the end of the body section, just before the closing tag, it will make your web pages load faster, since it prevents obstruction of initial page rendering.

ولكن من الناحية المثالية، يجب وضع البرامج النصية في نهاية قسم النص، قبل علامة الإغلاق مباشرة، فهذا سيجعل صفحات الويب الخاصة بك يتم تحميلها بشكل أسرع، لأنه يمنع عرقلة العرض الأولي للصفحة.

JAVASCRIPT SYNTAX

- JavaScript is case-sensitive. This means that variables, language keywords, function names, and other identifiers must always be typed with a consistent capitalization of letters.

JAVASCRIPT SYNTAX

بناء جملة جافا سكريبت

- JavaScript

• جافا سكريبت

case-sensitive.

حساسية الموضوع.

This

هذا

means

وسائل

that

الذي - التي

variables,

المتغيرات،

language keywords, function names, and other identifiers must always be typed with a consistent capitalization of letters.

يجب دائماً كتابة الكلمات الأساسية للغة وأسماء الوظائف والمعرفات الأخرى بأحرف كبيرة متنسقة.

صفحة (9) | تُرجمت بواسطة @xFxBot

JAVASCRIPT DATA TYPES

- There are six basic data types in JavaScript :
 - *Primary* data types
 - String: `let b = "Hi there!"`
 - Number: `let b = 80.5;`
 - Boolean: `let isReading = true;`
 - *Composite* data types
 - Object: `let car = { modal: "BMW X3", color: "white", doors: 5 }`
 - Array: `let colors = ["Red", "Yellow", "Green", "Orange"];`
 - Function: `let greeting = function(){ return "Hello World!"; }`
 - *special* data types [Example](#)
 - Undefined
 - Null

JAVASCRIPT DATA TYPES

أنواع بيانات جافا سكريبت

- There are six basic data types in JavaScript :

• هناك ستة أنواع أساسية من البيانات في JavaScript:

- Primary data types

• أنواع البيانات الأساسية

- String: let b = "Hi there!"

• السلسلة: Let b = "مرحبًا!"

- Number: let b = 80.5;

• العدد: دع ب = 80.5؛

- Boolean: let isReading = true;

• منطقية: دع القراءة = صحيح؛

- Composite data types

• أنواع البيانات المركبة

- Object: `let car = { modal: "BMW X3", color: "white", doors: 5 }`

الكائن: `Let car = { modal: "BMW X3", اللون: "أبيض", الأبواب: 5 }`

- Array: `let colors = ["Red", "Yellow", "Green", "Orange"];`

المصفوفة: `دع الألوان = ["أحمر", "أصفر", "أخضر", "برتقالي"];`

- Function: `let greeting = function() { return "Hello World!"`

الوظيفة: `!Let Greeting = function() { return "Hello World"`

- special data types Example

• أنواع البيانات الخاصة مثال

- Undefined

• غير معرف

• Null 10

• لاغية 10

صفحة (10) | تُرجمت بواسطة @xFxBot

JAVASCRIPT COMMENTS

- A comment is simply a line of text that is completely ignored by the JavaScript interpreter.
- JavaScript support single-line as well as multi-line comments.
 - Single-line comments begin with a double forward slash (`//`), followed by the comment text.
 - multi-line comment begins with a slash and an asterisk (`/*`) and ends with an asterisk and slash (`*/`).

JAVASCRIPT COMMENTS

تعليقات جافا سكريبت

- A comment is simply a line of text that is completely ignored by the JavaScript interpreter.
التعليق هو مجرد سطر من النص يتم تجاهله بالكامل بواسطة مترجم JavaScript.
- JavaScript support single-line as well as multi-line comments.
دعم جافا سكريبت للتعليقات ذات السطر الواحد وكذلك التعليقات متعددة الأسطر.
- Single-line comments begin with a double forward slash (//), followed by the comment text.
تبدأ التعليقات المكونة من سطر واحد بشرطة مائلة مزدوجة للأمام (//)، متبوعة بنص التعليق.
- multi-line comment begins with a slash and an asterisk (/*) and ends with an asterisk and slash (*).
التعليق متعدد الأسطر يبدأ بشرطة مائلة وعلامة النجمة (*/) وينتهي بعلامة النجمة والشرطة المائلة (/*).

JAVASCRIPT VARIABLES

- Create a variable with the `let` keyword, whereas the assignment operator (`=`) is used to assign value to a variable, like this:

```
let varName = value, var1Name= value1, ...;
```

- In JavaScript, if a variable has been declared, but has not been assigned a value explicitly, is automatically assigned the value `undefined`.
- Rules for naming a JavaScript variable:
 - A variable name must start with a letter, or underscore, or dollar sign.
 - A variable name cannot start with a number.
 - A variable name can only contain alpha-numeric characters (A-z, 0-9) and underscores.
 - A variable name cannot contain spaces.
 - A variable name cannot be a JavaScript reserved word.

- **What about `var` and `const`?!**

JAVASCRIPT VARIABLES

متغيرات جافا سكريبت

- Create a variable with the let keyword, whereas the assignment

• إنشاء متغير مع الكلمة الأساسية Let، في حين أن المهمة

operator (=) is used to assign value to a variable, like this:

يتم استخدام العامل (=) لتعيين قيمة لمتغير، مثل هذا:

let varName = value, var1Name= value1,...;

دع ;...,varName = value, var1Name= value1

- In JavaScript, if a variable has been declared, but has not been

• في JavaScript، إذا تم الإعلان عن متغير، ولكن لم يتم الإعلان عنه

assigned a value explicitly, is automatically assigned the

يتم تعيين قيمة بشكل صريح، ويتم تعيينها تلقائيًا

value undefined.

القيمة غير محددة.

• Rules for naming a JavaScript variable:

• قواعد تسمية متغير JavaScript:

• A variable name must start with a letter, or underscore, or dollar

• يجب أن يبدأ اسم المتغير بحرف أو شرطة سفلية أو دولار

sign.

لافتة.

• A variable name cannot start with a number.

• لا يمكن أن يبدأ اسم المتغير برقم.

• A variable name can only contain alpha-numeric characters (A-

• يمكن أن يحتوي اسم المتغير فقط على أحرف أبجدية رقمية (A-

z, 0-9) and underscores.

ض، 0-9) والشرطات السفلية.

- A variable name cannot contain spaces.

• لا يمكن أن يحتوي اسم المتغير على مسافات.

- A variable name cannot be a JavaScript reserved word.

• لا يمكن أن يكون اسم المتغير كلمة محجوزة في JavaScript.

- What about var and const?

• ماذا عن var وconst؟

JAVASCRIPT GENERATING OUTPUT

- **Displaying Output in Alert Dialog Boxes:** An alert dialog box is created using the `alert()` method.

```
alert("Hello World!"); // Outputs: Hello World!
```

- **Writing Output to the Browser Window:** the `document.write()` method can be used to write the content to the current document only while that document is being parsed.

```
document.write("Hello World!"); // Prints: Hello World!
```

- Using `document.write()` after an HTML document is loaded, will delete all existing HTML (Example)

JAVASCRIPT GENERATING OUTPUT

جافا سكريبت توليد المخرجات

- Displaying Output in Alert Dialog Boxes: An alert dialog box is

• عرض المخرجات في مربعات حوار التنبيه: يوجد مربع حوار تنبيه

created using the alert() method.

تم إنشاؤها باستخدام طريقة التنبيه ().

```
alert("Hello World!
```

تنبيه ("مرحبا بالعالم!

```
"); // Outputs: Hello World!
```

("; // المخرجات: أهلاً بالعالم!

- Writing

• كتابة

Output

انتاج |

the

ال

Browser

المتصفح

Window:

نافذة او شباك:

the document.write() method

طريقة document.write()

can be used to write the

يمكن استخدامها لكتابة

content to the current document only while that document is being parsed.

محتوى المستند الحالي فقط أثناء تحليل هذا المستند.

```
document.write("Hello World!
```

```
)document.write("مرحبا بالعالم!
```

```
"); // Prints: Hello World!
```

```
"); // المطبوعات: أهلاً بالعالم!
```

- Using `document.write()` after an HTML document is loaded, will delete all existing HTML (Example) 13

• استخدام `document.write()` بعد تحميل مستند HTML، سيؤدي إلى حذف كل HTML الموجود (مثال) 13

JAVASCRIPT GENERATING OUTPUT

- **Inserting Output Inside an HTML Element:** insert output inside an HTML element using the element's `innerHTML` property. However, first select the element using a method such as `getElementById()`

```
Let x = 10;
```

```
let y = 20;
```

```
let sum = x + y;
```

```
Document.getElementById("result").innerHTML =  
sum;
```

- For debugging purposes, you can call the **`console.log()`** method in the browser to display data ([Example](#))

JAVASCRIPT GENERATING OUTPUT

جافا سكريبت توليد المخرجات

- Inserting Output Inside an HTML Element: insert output inside an

• إدراج الإخراج داخل عنصر HTML: إدراج الإخراج داخل

HTML element using the element's innerHTML property.

عنصر HTML يستخدم خاصية HTML الداخلية للعنصر.

However,

لكن،

first select the element using a method such as getElementById()

حدد أولاً العنصر باستخدام طريقة مثل getElementById()

Let x = 10; let y = 20; let sum = x + y; Document.getElementById("result").innerHTML = sum;

Document.getElementById("result").innerHTML = دع س = 10؛ دع ص = 20؛ دع المبلغ = س + ص؛
;sum

- For debugging purposes, you can call the console.log() method in the browser to display data (Example)

• لأغراض التصحيح، يمكنك استدعاء الأسلوب console.log() في المتصفح لعرض البيانات (مثال)

JAVASCRIPT EVENTS

- An event is something that happens when user interact with the web page, such as when he clicked a link or button, moved the mouse pointer, submits a form, etc.
- The names for event handlers always begin with the word "on", so an event handler for the click event is called onclick, similarly an event handler for the load event is called onload, event handler for the blur event is called onblur, and so on.
- The events can be categorized into four main groups:
 - Mouse Events
 - Keyboard Events
 - Form Events
 - Document/Window Events.

- An event is something that happens when user interact with the web page, such as when he clicked a link or button, moved the mouse pointer, submits a form, etc.

• الحدث هو شيء يحدث عندما يتفاعل المستخدم مع صفحة الويب، مثل عندما يقوم بالنقر فوق رابط أو زر، أو تحريك مؤشر الماوس، أو إرسال نموذج، وما إلى ذلك.

- The names for event handlers always begin with the word "on", so an event handler for the click event is called onclick, similarly an event handler for the load event is called onload, event handler for the blur event is called onblur, and so on.

• تبدأ أسماء معالجات الأحداث دائماً بالكلمة "on"، لذلك يُسمى معالج الحدث لحدث النقر onclick، وبالمثل يُسمى معالج الحدث لحدث التحميل onload، ويسمى معالج الحدث لحدث التمويه onblur، و قريباً.

- The events can be categorized into four main groups:

• يمكن تصنيف الأحداث إلى أربع مجموعات رئيسية:

- Mouse Events

• أحداث الماوس

- Keyboard Events

• أحداث لوحة المفاتيح

- Form Events

• أحداث النموذج

- Document/Window Events.

• أحداث المستند/النافذة.

MOUSE EVENTS

- **The Click Event (onclick):** occurs when a user clicks on an element on a web page. A click event can be handled with an onclick event handler ([Example](#))
- **The Mouseover Event (onmouseover) :**occurs when a user moves the mouse pointer over an element.([Example](#))

MOUSE EVENTS

أحداث الماوس

- The Click Event (onclick): occurs when a user clicks on an element on a web page.

• حدث النقر (onclick): يحدث عندما يقوم المستخدم بالنقر فوق عنصر في صفحة ويب.

A click event can be handled with an onclick event handler (Example)

يمكن التعامل مع حدث النقر باستخدام معالج حدث onclick (مثال)

- The Mouseover Event (onmouseover) :occurs when a user moves the mouse pointer over an element.

• حدث تمرير الماوس (onmouseover): يحدث عندما يقوم المستخدم بتحريك مؤشر الماوس فوق أحد العناصر.

(Example)

(مثال)

FORM EVENTS

- **The Focus Event (onfocus):** The focus event occurs when the user gives focus to an element on a web page ([Example](#))
- **The Blur Event (onblur):** The blur event occurs when the user takes the focus away from a form element or a window ([Example](#))
- **The Change Event (onchange):** The change event occurs when a user changes the value of a form element ([Example](#))
- **The Submit Event (onsubmit):** The submit event only occurs when the user submits a form on a web page. ([Example](#))

FORM EVENTS

أحداث النموذج

- The Focus Event (onfocus): The focus event occurs when the user gives focus to an element on a web page (Example)

• حدث التركيز (onfocus): يحدث حدث التركيز عندما يركز المستخدم على عنصر في صفحة ويب (مثال)

- The Blur Event (onblur): The blur event occurs when the user takes the focus away from a form element or a window (Example)

• حدث التمويه (onblur): يحدث حدث التمويه عندما يقوم المستخدم بإبعاد التركيز عن عنصر نموذج أو نافذة (مثال)

- The Change Event (onchange): The change event occurs when a user changes the value of a form element (Example)

• حدث التغيير (onchange): يحدث حدث التغيير عندما يقوم المستخدم بتغيير قيمة عنصر النموذج (مثال)

- The Submit Event (onsubmit): The submit event only occurs when the user submits a form on a web page.

• حدث الإرسال (onsubmit): يحدث حدث الإرسال فقط عندما يرسل المستخدم نموذجًا على صفحة ويب.

صفحة (17) | تُرجمت بواسطة @xFxBot

DOCUMENT/WINDOW EVENTS

- **The Load Event (onload):** The load event occurs when a web page has finished loading in the web browser ([Example](#))
- **The Unload Event (onunload):** The unload event occurs when a user leaves the current web page ([Example](#))

DOCUMENT/WINDOW EVENTS

أحداث الوثيقة/النافذة

- The Load Event (onload): The load event occurs when a web page has finished loading in the web browser (Example)

• حدث التحميل (onload): يحدث حدث التحميل عندما تنتهي صفحة الويب من التحميل في متصفح الويب (مثال)

- The Unload Event (onunload): The unload event occurs when a user leaves the current web page (Example)

• حدث الإلغاء (onunload): يحدث حدث الإلغاء عندما يغادر المستخدم صفحة الويب الحالية (مثال)

JAVASCRIPT CONDITIONAL STATEMENTS

- There are several conditional statements in JavaScript:
 - The **if** statement
 - The **if...else** statement
 - The **if...else if...else** statement
 - The **switch...case** statement

JAVASCRIPT CONDITIONAL STATEMENTS

بيانات جافا سكريبت الشرطية

- There are several conditional statements in JavaScript:

• هناك العديد من العبارات الشرطية في JavaScript:

- The if statement

• بيان إذا

- The if...else statement

• عبارة if...else

- The if...else if...else statement

• عبارة if...else if...else

- The switch...case statement

• التبديل...بيان الحالة

JAVASCRIPT LOOPS

- JavaScript supports five different types of loops:
 - **while** — loops through a block of code as long as the condition specified evaluates to true.
 - **do...while** — loops through a block of code once; then the condition is evaluated. If the condition is true, the statement is repeated as long as the specified condition is true.
 - **for** — loops through a block of code until the counter reaches a specified number.
 - **for...in** — loops through the properties of an object.
 - **for...of** — loops over iterable objects such as arrays, strings, etc.

JAVASCRIPT LOOPS

حلقات جاافا سكريببت

- JavaScript supports five different types of loops:

• تدعم JavaScript خمسة أنواع مختلفة من الحلقات:

- while — loops through a block of code as long as the condition specified evaluates to true.
• while — يتكرر عبر مجموعة من التعليمات البرمجية طالما تم تقييم الشرط المحدد على أنه صحيح.
- do...while — loops through a block of code once; then the condition is evaluated.
• do... while — تكرر مجموعة من التعليمات البرمجية مرة واحدة؛ ومن ثم يتم تقييم الحالة.

If the condition is true, the statement is repeated as long as the specified condition is true.

إذا كان الشرط صحيحا، يتم تكرار العبارة طالما أن الشرط المحدد صحيحا.

- for — loops through a block of code until the counter reaches a specified number.
• for — يتكرر عبر مجموعة من التعليمات البرمجية حتى يصل العداد إلى رقم محدد.

- `for...in` — loops through the properties of an object.

• `for...in` — حلقات من خلال خصائص الكائن.

- `for...of` — loops over iterable objects such as arrays, strings, etc.

• `for...of` — حلقات على كائنات قابلة للتكرار مثل المصفوفات والسلاسل وما إلى ذلك.

صفحة (20) | تُرجمت بواسطة @xFxBot

THE DOCUMENT OBJECT MODEL

- The Document Object Model (DOM) is a platform and language independent model to represent the HTML documents.
- DOM defines the logical structure of the documents and the way in which they can be accessed and manipulated by an application program.
- In the DOM, all parts of the document, such as elements, attributes, text, etc. are organized in a hierarchical tree-like structure.

THE DOCUMENT OBJECT MODEL

نموذج كائن الوثيقة

- The Document Object Model (DOM)

• نموذج كائن المستند (DOM)

is a platform and

هي منصة و

language

لغة

independent

مستقل

model

نموذج

represent

يمثل

the

ال

HTML

لغة البرمجة

documents.

وثائق.

- DOM defines the logical structure of the documents and the way in which they can be accessed and manipulated by an application program.

- يحدد DOM البنية المنطقية للمستندات والطريقة التي يمكن من خلالها الوصول إليها ومعالجتها بواسطة برنامج تطبيقي.

- In the DOM, all parts of the document, such as elements, attributes, text, etc.

- في DOM، يتم حفظ جميع أجزاء المستند، مثل العناصر والسمات والنص وما إلى ذلك.

are organized in a hierarchical tree-like

يتم تنظيمها بشكل هرمي يشبه الشجرة

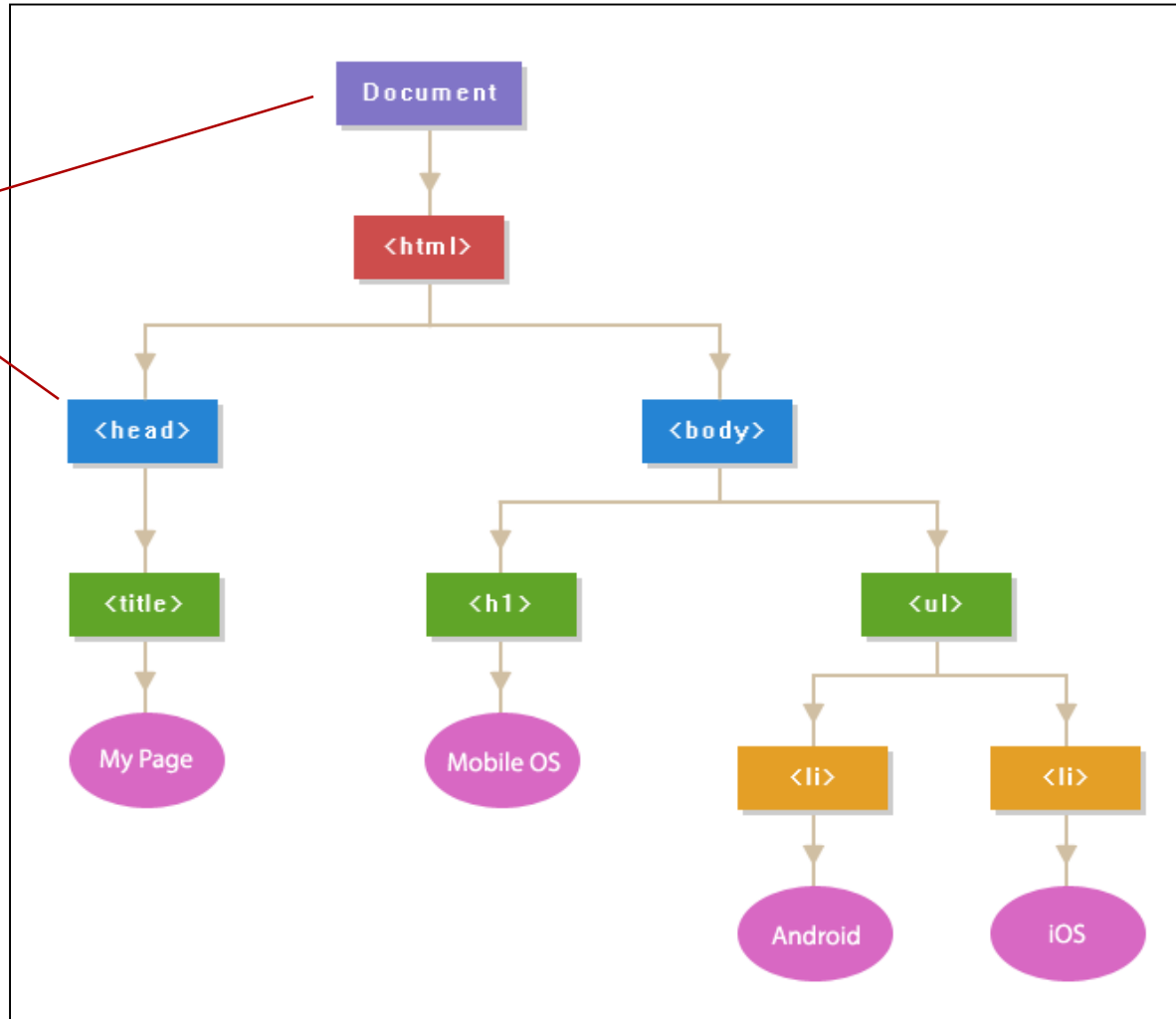
structure.

بناء.

صفحة (21) | تُرجمت بواسطة @xFxBot

HTML DOM

Nodes



HTML DOM

HTML-دوم

Nodes

العقد

صفحة (22) | تُرجمت بواسطة @xFxBot

JAVASCRIPT DOM SELECTORS

- **Selecting the Topmost Elements:** The topmost elements in an HTML document are available directly as document properties.
 - For example:
 - the `<html>` element can be accessed with `document.documentElement` property.
 - the `<head>` element can be accessed with `document.head` property
 - the `<body>` element can be accessed with `document.body` property.
- **Selecting Elements by ID:** the easiest way to find an HTML element in the DOM tree is select an element based on its unique ID with the `getElementById()` method.
 - The `getElementById()` method will return the element as an object if the matching element was found, or null if no matching element was found in the document.

JAVASCRIPT DOM SELECTORS

محددات DOM لجافا سكريبت

- Selecting the Topmost Elements: The topmost elements in an HTML document are available directly as document properties.

• تحديد العناصر العليا: تتوفر العناصر العليا في مستند HTML مباشرةً كخصائص للمستند.

- For example:

• على سبيل المثال:

- the

• ال

<أنتش تي أم آل>

element

عنصر

can

يستطيع

accessed

الوصول إليها

with document.documentElement property.

مع خاصية document.documentElement

<الرأس>

with document.head property

document.head مع خاصية

<الجسم>

with document.body property.

document.body مع خاصية

- Selecting Elements by ID: the easiest way to find an HTML element in the DOM tree is select an element based on its unique ID with the `getElementById()` method.

• تحديد العناصر حسب المعرف: أسهل طريقة للعثور على عنصر HTML في شجرة DOM هي تحديد عنصر بناءً على معرفه الفريد باستخدام طريقة `getElementById()`.

- The `getElementById()` method will return the element as an `23` object if the matching element was found, or `null` if no matching element was found in the document.

• ستعيد طريقة `getElementById()` العنصر ككائن `23` إذا تم العثور على العنصر المطابق، أو فارغة إذا لم يتم العثور على عنصر مطابق في المستند.

JAVASCRIPT DOM SELECTORS

- **Selecting Elements by Class Name:** the `getElementsByClassName()` method is used to select all the elements having specific class names.
 - This method returns an array-like object of all child elements which have all of the given class names.
- **Selecting Elements by Tag Name:** select HTML elements by tag name using the `getElementsByTagName()` method.
 - This method also returns an array-like object of all child elements with the given tag name.

JAVASCRIPT DOM SELECTORS

محددات DOM لجافا سكريبت

• Selecting

• الاختيار

Elements

عناصر

Class

فصل

Name: the `getElementsByClassName()` method is used to select all the elements having specific class names.

الاسم: يتم استخدام طريقة `getElementsByClassName()` لتحديد جميع العناصر التي لها أسماء فئات محددة.

- This method returns an array-like object of all child elements which have all of the given class names.

• تقوم هذه الطريقة بإرجاع كائن يشبه المصفوفة لجميع العناصر التابعة التي تحتوي على كافة أسماء الفئات المحددة.

- Selecting Elements by Tag Name: select HTML elements by tag name using the `getElementsByTagName()` method.

• تحديد العناصر حسب اسم العلامة: حدد عناصر HTML حسب اسم العلامة باستخدام طريقة `getElementsByTagName()`.

- This method also returns an array-like object of all child elements with the given tag name.

• تقوم هذه الطريقة أيضاً بإرجاع كائن يشبه المصفوفة لجميع العناصر التابعة باسم العلامة المحدد.

JAVASCRIPT DOM STYLING

- **Setting Inline Styles on Elements:** the style property is used to get or set the inline style of an element.

JAVASCRIPT DOM STYLING

تصميم جافا سكريبت دوم

- Setting Inline Styles on Elements: the style property is used to

• ضبط الأنماط المضمنة على العناصر: يتم استخدام خاصية النمط

get or set the inline style of an element.

الحصول على أو تعيين النمط المضمّن للعنصر.

JAVASCRIPT DOM GET SET ATTRIBUTES

- **getAttribute() method:** is used to get the current value of a attribute on the element ([Example](#))
- **setAttribute() method:** is used to set an attribute on the specified element. If the attribute already exists on the element, the value is updated; otherwise a new attribute is added with the specified name and value ([Example](#))
- **removeAttribute() method:** is used to remove an attribute from the specified element ([Example](#))

JAVASCRIPT DOM GET SET ATTRIBUTES

جافا سكريبت DOM احصل على السمات المحددة

- `getAttribute()` method: is used to get the current value of a attribute on the element
(Example)

• طريقة `getAttribute()` : تستخدم للحصول على القيمة الحالية للسمات الموجودة على العنصر (مثال)

- `setAttribute()` method: is used to set an attribute on the specified element.

• طريقة `setAttribute()` : تستخدم لتعيين سمات على العنصر المحدد.

If the attribute already exists on the element, the value is updated; otherwise a new attribute is added with the specified name and value (Example)

إذا كانت السمات موجودة بالفعل على العنصر، فسيتم تحديث القيمة؛ وإلا فسيتم إضافة سمات جديدة بالاسم والقيمة المحددين (مثال)

- `removeAttribute()` method: is used to remove an attribute from the specified element
(Example) 26

• طريقة `RemoveAttribute()` : تستخدم لإزالة سمات من العنصر المحدد (مثال) 26

GETTING/SETTING HTML CONTENTS TO DOM

- **The innerHTML property:** this property sets or gets the HTML markup contained within the element i.e. content between its opening and closing tags.
- **The innerHTML property replaces all existing content of an element.**

GETTING/SETTING HTML CONTENTS TO DOM

الحصول على/إعداد محتويات HTML إلى DOM

- The innerHTML property: this property sets or gets the HTML markup contained within the element i.e.

• الخاصية الداخلية HTML: تقوم هذه الخاصية بتعيين أو الحصول على علامة HTML الموجودة داخل العنصر، على سبيل المثال.

content between its opening and closing tags.

المحتوى بين علامتي الفتح والإغلاق.

- The innerHTML property replaces all existing content of an element.

• الخاصية الداخلية HTML تحل محل كل المحتوى الموجود للعنصر.

JAVASCRIPT FORM VALIDATION

- Client-side validation is faster because validation occurs within the user's web browser, whereas server-side validation occurs on the server, which require user's input to be first submitted and sent to the server before validation occurs, also user has to wait for server response to know what exactly went wrong.
- The form validation process typically consists of two parts:
 - **The *Required Fields Validation*:** is performed to make sure that all the mandatory fields are filled in.
 - **The *Data Format Validation*:** is performed to ensure that the type and format of the data entered in the form is valid.

JAVASCRIPT FORM VALIDATION

التحقق من صحة نموذج جافا سكريبت

- Client-side validation is faster because validation occurs within the user's web browser, whereas server-side validation occurs on the server, which require user's input to be first submitted and sent to the server before validation occurs, also user has to wait for server response to know what exactly went wrong.

• يكون التحقق من جانب العميل أسرع نظراً لأن التحقق من الصحة يحدث داخل متصفح الويب الخاص بالمستخدم، بينما يحدث التحقق من جانب الخادم على الخادم، مما يتطلب إرسال إدخلات المستخدم أولاً وإرسالها إلى الخادم قبل حدوث التحقق من الصحة، كما يجب على المستخدم انتظار الرد لمعرفة الخطأ الذي حدث بالضبط.

- The form validation process typically consists of two parts:

• تتكون عملية التحقق من صحة النموذج عادةً من جزأين:

- The Required Fields Validation: is performed to make sure that all the mandatory fields are filled in.

• التحقق من صحة الحقول المطلوبة: يتم إجراؤه للتأكد من ملء جميع الحقول الإلزامية.

- The Data Format Validation: is performed to ensure that the type and format of the data entered in the form is valid.

• التحقق من صحة تنسيق البيانات: يتم إجراؤه للتأكد من صحة نوع وتنسيق البيانات المدخلة في النموذج.

صفحة (28) | تُرجمت بواسطة @xFxBot

JAVASCRIPT FORM VALIDATION

- The value of an individual form field can be accessed and retrieved using the syntax

```
document.formName.fieldName.value
```

or

```
document.getElementsByName(name).value.
```

Good Example at the following link:

<https://www.tutorialrepublic.com/codelab.php?topic=javascript&file=form-validation>

Other useful examples:

<https://www.tutorialrepublic.com/javascript-examples.php>

JAVASCRIPT FORM VALIDATION

التحقق من صحة نموذج جافا سكريبت

- The value of an individual form field can be accessed and retrieved using the syntax `document.formName.fieldName.value` or `document.getElementsByName(name).value`.

- يمكن الوصول إلى قيمة حقل نموذج فردي واسترجاعها باستخدام بناء الجملة `document.getElementsByName(name).value` أو `document.formName.fieldName.value`.

Good Example at the following link: <https://www.tutorialrepublic.com/codelab.php?topic=javascript&file=form-validation>

مثال جيد على الرابط التالي: <https://www.tutorialrepublic.com/codelab.php?topic=javascript&file=form-validation>

Other useful examples: 29 <https://www.tutorialrepublic.com/javascript-examples.php>

أمثلة مفيدة أخرى: 29 <https://www.tutorialrepublic.com/javascript-examples.php>

THE END

THE END

النهاية

صفحة (30) | تُرجمت بواسطة @xFxBot