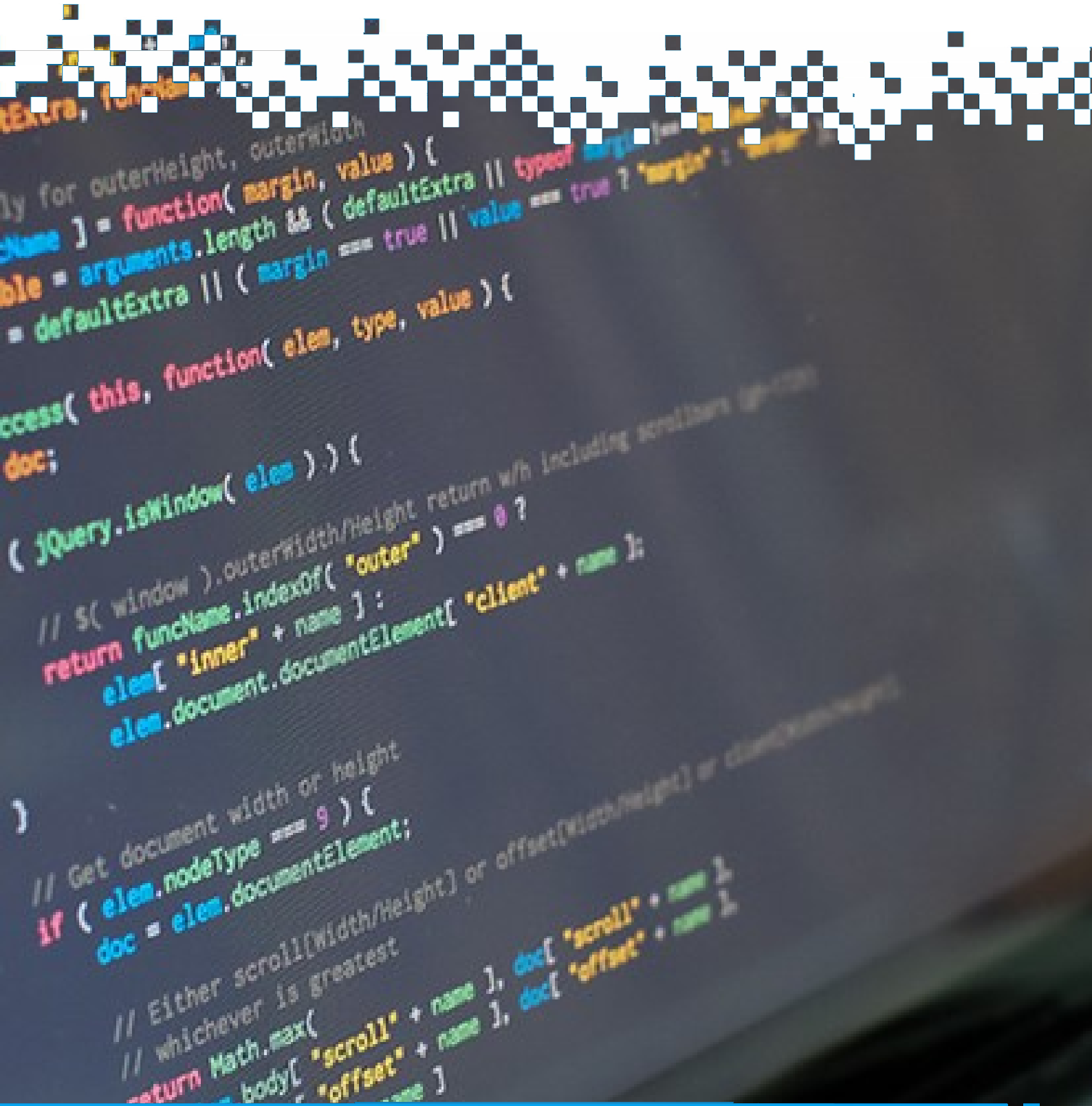
NASSCOM® FOUNDATION

Directorate General of Training

Powered by Microsoft



Session 1: Embed JavaScript in HTML to Display Information in Web Pages

Session 1: Embed JavaScript in HTML to Display Information in Web Pages

JavaScript and HTML

1. JavaScript and HTML are both independent programming languages.
2. They are used together as per pre-defined rules. Certain rules are used to combine JavaScript with HTML.
3. These rules are necessary to ensure that two different programming languages can be integrated.
4. The web pages designed using this potent combination are powerful.
5. These modern web pages are interactive, user-friendly and intuitive.
6. In some cases, JavaScript codes are hidden behind HTML tags as the older versions of web browsers do not support JavaScript.

JavaScript and HTML

CSS Components of an HTML

1. The HTML code has two major sections. They are the head and the body.
2. The most commonly used and preferred location for coding is the head section.
3. The script tag is where the JavaScript code is written.
4. The syntax of element or tag is less than and greater than symbols.

   Syntax of element or tag:

   <<>(less than and greater than symbols)

Script Tags

1. Script tag is used to define the variables, functions, predefined methods of JavaScript code within the HTML page.
2. When HTML compiler encounters the script tag, it understands that there is JavaScript code within it.
3. It enhances the structure including the powerful combination of JavaScript and HTML.
4. All the actions are present in the script.

Uses of Section

1. Sections help the programmer identify the location to write the code.
2. JavaScript code can be executed in multiple sections.
3. Sections are used to identify the start and end of the codes.
4. Sectioning helps to improve the document structure.
5. Sectioning enhances the structure of the powerful combination of JavaScript and

Method to Combine JS with HTML

1. JavaScript code can be written only in Head script,
2. Or only in body script,
3. Or written in both head and body elements, inside the HTML document

Steps to Write Code and Save the File

1. Open **notepad** or any other IDE.
2. Write the code.
3. Use console.log to display values in the console.
4. Save the file with demo.html in a location.
5. Use .html extension after your file name.

6. Go to the file location.
7. Select the file.
8. Press **Enter.**
9. You will be taken to the default browser in your system.
10. To generate output, right-click +
11. Select **Inspect.**
12. Use the shortcut **Ctrl** + **Shift** + **I.**

Tools to Create/ Run/ Execute a JavaScript Code

1. Let us now look at some of the tools used to create, run or execute a JavaScript code.
2. To create, run or execute a JavaScript code, we use tools like Integrated Development Environment (IDE), Browser, Live Servers, Node JS.
3. Applications like Node JS also function like an IDE and can be used to create and run a JavaScript code.
4. Some Extension Packages are also helpful.

- Create
- Run
- Execute

- Integrated Development Environment (IDE)
- Browser
- Live Servers

Some Extension Packages are also helpful.

Use of IDE in JavaScriptOutput:

1. Every programming language has an IDE to write a code and to modify the existing code.
2. IDE is a powerful editor with many features that usually operate on a whole project.
3. As the name suggests, IDE is not just an editor, but a full-scale development environment.
4. IDE loads the project, which can include many files.
5. IDE allows navigation between files.
6. It provides auto completion based on the whole project, not just the open file.
7. It integrates with a version management system like GIT, a testing environment and other "project-level" activities.

Use JavaScript without HTML:

1. To create, run or execute a code without using HTML:
2. The JavaScript code needs to be written and saved as .js file("dot JS").
3. Then, the file is run for the code to be executed.
4. This is done using online compilers and IDEs for writing and executing the JavaScript code.

List of IDE Platforms

Notepad++

- Windows
- Free

Visual Studio Code

- Cross Platform
- Free

WebStorm

- Cross Platform
- Paid

Emacs

- Require learning

Sublime Text

- Cross platform
- Shareware

Atom

- Cross Platform
- Free

Vim

IDE Platforms – Online

Jsfiddle.net

1. Jsfiddle is an online platform to run JavaScript code without HTML file. Let us look at the steps to use this platform.
2. Go to browser and search jsfiddle.net. Then, press Enter.
3. The browser will be redirected to this page.

4. At the left bottom, you can see the area to write your JS code.
5. Write a code to display a message from the document.write() method and from the console.log() method.
6. Simply click the Run button, which is present at the left top.
7. At the bottom right, you can see your result.
8. By default, we can see the console as well in the result area without going to the browser console.

JS.do

1. JS.do is an online platform to run JavaScript code without HTML file. Let us look at the steps to use this platform.
2. Go to browser and enter js.do in the search bar to navigate to the js.do page.
3. You will see a page like this with some default time, and some instructions.
4. At the right side, you will see the output when you click on the Run button.
5. At the left side, you can edit or modify your code without HTML.
6. Change the script as below and check the console log. The page is updated and shows the new message.

JSconsole

1. One of the best online platforms to run JavaScript code without HTML file is jconsole.com. Let us look at the steps to use this platform.
2. Go to any browser and type jsconsole.com in search bar. On clicking, a page like this appears.
3. Enter script at the prompt.
4. This code is meant to create an alert box when run.
5. When the code is run by pressing enter, a pop-up or dialogue box will appear on the screen.

Developer Tools

1. Developers need to identify the flow of the process as well as the errors that may occur.
2. Developer tools used for this activity are:
3. Console and
4. Debugger

Console

Console is an object, which provides access to the browser debugging console.

1. Press F12 for Chrome, Cmd + Opt + J on Mac or right-click + Inspect for Microsoft Edge. The shortcut is Ctrl + Shift + I for Edge to open the console window.
2. The exact look of the developer tools like console, depends on the version of Chrome used.
3. Every browser has its own console for viewing the output, errors and other important information regarding the JavaScript code.
4. There are several methods used with console object. The most commonly used methods include log, clear and count.

Debugger

1. Debugger or break point is an important part of coding that helps in identifying the process flow as well as errors that may occur.
2. The debugger keyword is used in the code to force stop the execution of the code at a breaking point and calls the debugging function. The debugging function is executed if any debugging is needed at all, else no action is performed.

Attributes of a Script

1. The HTML script element is used to embed executable code or data, or to refer to a JavaScript code.
2. Script element either contains scripting statements or it points to an external script file through the src attribute.
3. The script element can also be used with other languages, such as WebGL's GLSL and JSON.

Script Attributes – SRC

1. SRC is used frequently while writing JavaScript code.
2. The value assigned to attribute is the path or URL of the external JavaScript file.

Attribute	Description	Value
SRC	Specifies the URL of an external script file	URL

Types of Scripts

Internal script	External script
The JavaScript code is written within the same HTML document	Uses .js extension file & .html extension file
The document carries both HTML code and JavaScript code	Uses src attribute in the script element
	Used for code splitting

	The html code is written in a particular file and JavaScript in another file

Use Internal JavaScript Files in HTML

Internal JavaScript Files in HTML

1. Write the code on notepad. Save it as demo.html at its location.
2. A script element is used because everything in html is present in tags only. console.log method will display the data entered in double quotes.
3. Save the file and open in a browser. This is the output. The message which was entered in the quotes can be seen at the console on browser.

External JavaScript Files in HTML

1. An external JavaScript code will have src attribute in the script element. src is going to carry the path. This is the syntax for the src attribute.
2. The script element must remain empty when adding an external js file.
3. Similarly, a single <script> tag cannot have both the src attribute and the code. This will not work. Instead, split the code into two script tags.

Example for External JavaScript Files in HTML

1. Write the JavaScript code in an IDE like notepad.
2. Save it as external.js at its location.
3. Write the **source file** on a new notepad.
4. Save it as demo.html in the same location where external.js is saved.
5. Press Enter or double-click the file demo.html.
6. The file will run in default browser.

Choosing between Internal and External Files

Factors to be considered while deciding to choose an external file or internal file

1. Simple scripts that would be used only once are put into internal files. If more than one html file needs to use the same JavaScript code, then it is better to use external script.
2. When we use separate files, we call them external files. Here, the code needs to be written only once and can be used multiple times in the html document simply by using src attribute in the script tag. The benefit of a separate file is that the browser will download it and store it in its cache and other pages can access the cache when needed.
3. When using external files, it is important to avoid using src and content inside the same script while coding. Instead, split the code into two script tags. This helps to avoid errors when the code is being executed.

Using Multiple Script Tags

1. We can use multiple script elements in a single html page.
2. These carry external js files.
3. If you notice, each script is having its own src attribute.
4. Each src is having different values that represent the path where they are present.

Combining Internal and External Scripts

1. The code will be executed in a sequential order.
2. That is, if the internal and external scripts are combined in HTML document, then the script element which comes first will be executed first.
3. If we write the script in head and body element, even then the code will be executed in a sequential order.

Example of Code that Calls Internal Script First

1. Write the code in notepad.

2. Save it as demo.html at its location.
3. Go to the saved location, select file demo.html and double-click to open in a browser.
4. In the code, the internal script comes first from the head element. So, the alert which is present in the internal script element will execute first. The alert from internal script can be seen in the browser. Press OK, it will be executed. By pressing the OK button on the alert box, the alert box will be closed.

Document the Source Code

JavaScript Comments

1. JavaScript comments are used to explain JavaScript code and to make it more readable.
2. They are used to prevent execution of code when testing alternative code.
3. They are used to write notes in the code or to disable sections of code without deleting them.

Types of JavaScript Comments

Single Line Comment	Multi-Line Comment
Single line comment is represented using two backward	Multi-line comment starts with /* and ends with

slash //	*/.
Any text between // until the end of the line will be ignored by JavaScript	Any text between /* and */ will be ignored by JavaScript
Syntax : //code or statements	• Syntax /* code or statements … … .*/

JavaScript Comments

Example of Single Line Comment

1. Write the code in notepad.
2. Save it as demo.html at its location.
3. Go to the saved location, select file demo.html and double-click to open in a browser.

4. Here, the comment has been used in the script element. So, the second alert() will not execute.
5. Only the console.log() method and alert() will be executed.
6. When the file is saved and executed, then the console works first. It displays the message and the alert will work and display the message with button.
7. Now, after clicking OK in the 1st alert, the code has been executed. The second alert will not be displayed because it has been commented.
8. Now, the body code would be executed after skipping the code placed in the comment.

Example of Multi Line Comment

1. Write the code in notepad.
2. Save it as demo.html at its location.
3. Go to the saved location, select file demo.html and double-click to open in a browser.
4. Here, the comment has been used in the script element. So, the second alert() will not execute.
5. Only the console.log() method and alert() will be executed.
6. When the file is saved and executed, then the console works first. It displays the message and the alert will work and display the message with button.
7. Now, after clicking OK in the 1st alert, the code has been executed. The second alert will not be displayed because it has been commented.
8. Now, the body code would be executed after skipping the code placed in the comment.

Formatting Text in HTML Page

Format Text Using JavaScript Write() Method

1. Write() is used to format HTML expressions or JavaScript code to a document.
2. Using write() method is the best way of changing the text in HTML. Using elements in the method, we

can modify the text as required.

3. The write() method writes HTML expressions or JavaScript code to a document.
4. The write() method is mostly used for testing. If it is used after an HTML document is fully loaded, it will delete all existing HTML.
5. It has parameters or expressions. Parameters are optional. And include information like, what to write to the output stream.
6. Multiple arguments can be listed and they will be appended to the document in order of occurrence.
7. The syntax is shown here.

document write(exp1, exp2, exp3, ...) exp1,exp2 are optional.

Example of Using Document Write() Method

1. Open notepad.
2. Write the code.
3. Save the file with the name demo123.html.
4. Double-click the file at its saved location to open in a browser.
5. The code contains head and body elements. The script element inside the head contains a function called hel() in the code. The function is triggered when the user clicks on a button called 'Thank You'. On clicking 'Thank You' button, the document write() code is executed.
6. When the code is run, the page looks like this.

Types of write() Command

document.write()

Displays the different types of content or text in the same line.

document.writeln()

Displays the content or text in different lines.

Open() & Close() Methods

open() method	close() method
Opens an output stream to collect the output from any document.write() or document.writeln() methods.	The close() method closes the output stream previously opened with the document.open() method and displays the collected data in this process.
document.open() method can be used without parameters.	The close() method causes any output written to the output stream to be displayed after all the writes are performed.
Or parameters like Type and Replace can be used instead.	

Open() & Close() Method - Syntax

MIMEtype	replace
Mimetype is an optional parameter and refers to the type of document to which you are writing.	Replace is another optional parameter and if set, refers to the history entry for the new document.
Default value is "text/html".	The new document will be placed before the current item in the history list.

Syntax:

document.open (MIMEtype, replace);

document.close();

Open() & Close() Methods - Example

Open() & Close() Methods - Example

With open and close method

```
document.open();

document.write("Hello. This is first write method");

document.write("This is second write method");

document.close();
```

Without open and close method

```
document.write("Hello. This is first write method");

document.write("This is second write method");
```

Factors to be considered while while using the write() method.

1. document.write(), document.writeln() methods work without document.open() and document.close() methods.
2. The older version document.write() method only works after calling the document.open().
3. It is essential to close the document using the document.close().
4. While the document is loading, the document stream is opened for writing. In that case, there is no need to open and close the output stream.
5. In document.writeln(), use <pre>, otherwise the code will not work.

Recap

1. JavaScript and HTML are both independent programming languages used together as per pre-defined rules.
2. In some cases, the JavaScript codes are hidden behind HTML tags as the older versions of web browsers do not support JavaScript.
3. Sections help the programmer identify the location to write the code. JavaScript code can be executed in multiple sections.
4. To create, run or execute a JavaScript code, we use tools like Integrated Development Environment, that is, IDE or browser or live servers.
5. IDE is a powerful editor with many features that usually operate on a whole project.
6. A few examples of such free online platforms that can be used are:
    - jsfiddle.net
    - js.do
    - jsconsole
7. We have two types of scripts:
    - Internal script and external script
8. The write() method is mostly used for testing. If it is used after an HTML document is fully loaded, it will delete all existing HTML.
9. h1 tag is used to increase the size of message and makes it bold.
10. p tag takes the text message into new line and brings some look.
11. pre tag is used to display a fixed-width font. The text displays exactly as written in the HTML.