Chapter 13: Include JavaScript Code in HTML

Attribute	Details
`src`	Specifies the path to a JavaScript file. Either a relative or absolute URL.
`type`	Specifies the MIME type. This attribute is required in HTML4, but optional in HTML5.
`async`	Specifies that the script shall be executed asynchronously (only for external scripts). This attribute does not require any value (except of XHTML).
`defer`	Specifies that the script shall be executed when the page has finished parsing (only for external scripts). This attribute does not require any value (except of XHTML).
`charset`	Specifies the character encoding used in an external script file, e.g. UTF-8
`crossorigin`	How the element handles crossorigin requests
`nonce`	Cryptographic nonce used in *Content Security Policy* checks CSP3

Section 13.1: Handling disabled Javascript

It is possible that the client browser does not support Javascript or have Javascript execution disabled, perhaps due to security reasons. To be able to tell users that a script is supposed to execute in the page, the **`<noscript>`** tag can be used. The content of **`<noscript>`** is displayed whenever Javascript is disabled for the current page.

```
<script>
  document.write("Hello, world!");
</script>
<noscript>This browser does not support Javascript.</noscript>
```

Section 13.2: Linking to an external JavaScript file

```
<script src="example.js"></script>
```

The `src` attribute works like the href attribute on anchors: you can either specify an absolute or relative URL. The example above links to a file inside the same directory of the HTML document. This is typically added inside the **`<head>`** tags at the top of the html document

Section 13.3: Directly including JavaScript code

Instead of linking to an external file, you can also include the JS code as-is in your HTML:

```
<script>
// JavaScript code
</script>
```

Section 13.4: Including a JavaScript file executing asynchronously

```
<script type="text/javascript" src="URL" async></script>
```