async vs defer

async and defer are two attributes used in HTML <script> tags to control the loading and execution of external JavaScript files. They are used to optimize the loading of scripts and improve page performance. Let's explore the differences between async and defer:

async

When you add the async attribute to a <script> tag, the browser will begin downloading the script file immediately without blocking the rendering of the rest of the page.

As soon as the script file is downloaded, it will be executed, regardless of whether the HTML document has finished parsing.

This can lead to scripts executing out of order if they depend on each other or the DOM structure, potentially causing issues if not managed carefully.

Use async when the script does not depend on the DOM being fully loaded and when order of execution is not critical.

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

defer

When you add the defer attribute to a <script> tag, the browser will also begin downloading the script file immediately.

However, the script will be executed only after the HTML document has finished parsing.

Scripts with the defer attribute are executed in the order they appear in the HTML document, ensuring that they do not interfere with the DOM structure or other scripts.

Use defer when the script depends on the DOM being fully loaded and when you want to maintain order of execution.

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

In summary

Use async when the script can be executed independently and you want to minimize any potential rendering delays.

Use defer when the script depends on the DOM being fully parsed and you want to ensure scripts execute in the order they appear in the document.

It's important to note that both async and defer apply only to external scripts (those with a src attribute). Inline scripts do not have these attributes and are executed synchronously as they are encountered in the document.