



Loading Javascript may slow down your page as the browser will follow several steps for each script the page links to and not continue until the script is fully parsed and all the associated resources loaded:

The browser has no way of knowing what, if any, changes it needs to make to the page based on the script so it'll download and parse the script even if it ends up not using any of it in rendering the page itself.

There are two ways we can bypass the blocking scripts problem. Which one you use will depend on what you want to do. The options are `async` and `defer`.

`Async` scripts will download the script without blocking rendering the page and will execute it as soon as the script finishes downloading. You get no guarantee that scripts will run in any specific order, only that they will not stop the rest of the page from displaying. It is best to use it when the the scripts in the page run independently from each other and depend on no other script on the page.

For example, if you have the following script tags

```
<script async src="js/vendor/jquery.js"></script>
<script async src="js/script2.js"></script>
<script async src="js/script3.js"></script>
```

You can't rely on the order the scripts appear. jQuery may load before or after script2 and script3 and if any functions in those scripts depend on jQuery you will get an error because jQuery will not be defined at the time the script runs.

`Defer` will run the scripts in the order they appear in the page and execute them as soon as the script and content are downloaded.

```
<script defer src="js/vendor/jquery.js"></script>
<script defer src="js/script2.js"></script>
<script defer src="js/script3.js"></script>
```

All the scripts with the `defer` attribute will load in the order they appear on the page. So in the second example we can be sure that jQuery will load before script2 and script3 and that script2 will load before script3.

To summarize:

- If your scripts can run independently without requiring any dependencies then use `async`
- If your scripts depend on other scripts run them using `defer` and put them in the order you want the browser to execute them

Do `async` and `defer` matter in scripts at the bottom of the page

If you search the web for performance and scripts you will see that many authors suggest you put all your scripts at the bottom of the page right before the `</body>` tag.

The rationale for this is: By the time the browser gets to the scripts it will have parsed all the HTML of the page and, likely, displayed it to the user so there is no rendering to block.

Putting the scripts at the bottom of the page is an old recommendation that you may find when searching for information. Using `async` and `defer` there is no problem putting the scripts in the `<head>` of the document.