

Hello, world!

این بخش از آموزش در مورد هسته اوا اسکریپت است.

اما ما به یک محیط کاری برای اجرای اسکریپت های خود نیاز داریم و از آنجایی که این کتاب آنلاین است، مرورگر بعنوان محیط کاری انتخاب خوبی است. ما تعداد دستورات خاص مرورگر (مانند هشدار) را به حداقل می‌رسانیم تا اگر قصد دارید روی محیط دیگری (مانند Node.js) تمرکز کنید، وقت خود را برای آنها صرف نکنید. در قسمت بعدی آموزش روی جاوا اسکریپت در مرورگر تمرکز خواهیم کرد. بنابراین ابتدا، بیایید ببینیم چگونه یک اسکریپت را به یک صفحه وب متصل می‌کنیم. برای محیط های سمت سرور (مانند Node.js)، می‌توانید اسکریپت را با دستوری مانند `node my.js` اجرا کنید.

تگ script

برنامه های جاوا اسکریپت را می‌توان تقریباً در هر جایی در یک سند HTML با استفاده از تگ `<script>` وارد کرد. برای مثال:

```
1 <!DOCTYPE HTML>
2 <html>
3
4 <body>
5
6   <p>Before the script...</p>
7
8   <script>
9     alert( 'Hello, world!' );
10  </script>
11
12   <p>...After the script.</p>
13
14 </body>
15
16 </html>
```

Before the script...

...After the script.

تگ `<script>` حاوی کد جاوا اسکریپت است که زمانی که مرورگر تگ را پردازش می‌کند به طور خودکار اجرا می‌شود.

نشانه گذاری (markup) مدرن

تگ `<script>` دارای چند ویژگی است که امروزه به ندرت مورد استفاده قرار می گیرند اما هنوز در کدهای قدیمی یافت می شوند:

ویژگی `type`: `<script type=...>`

استاندارد قدیمی HTML، HTML4، برای داشتن یک نوع نیاز به یک اسکریپت داشت. معمولاً `type="text/javascript"` بود. دیگر مورد نیاز نیست همچنین استاندارد مدرن HTML معنای این ویژگی را به کلی تغییر داد. اکنون می توان از آن برای مازول های جاوا اسکریپت استفاده کرد. اما این یک موضوع پیشرفته است، ما در بخش دیگری از آموزش در مورد مازول ها صحبت خواهیم کرد.

ویژگی زبان: `<script language=...>`

این ویژگی برای نشان دادن زبان اسکریپت بود. این ویژگی دیگر معنی ندارد زیرا جاوا اسکریپت زبان پیش فرض است. نیازی به استفاده از آن نیست.

کامنت قبل و بعد از اسکریپت

در کتاب ها و راهنماهای واقعاً قدیمی، ممکن است نظراتی در داخل تگ های `<script>` پیدا کنید، مانند این:

```
1 <script type="text/javascript">!--  
2   ...  
3  //--></script>
```

این ترفند در جاوا اسکریپت مدرن استفاده نمی شود. این نظرات کد جاوا اسکریپت را از مرورگرهای قدیمی که نمی دانستند چگونه تگ `<script>` را پردازش کنند پنهان می کند. از آنجایی که مرورگرهای منتشر شده در 15 سال گذشته این مشکل را ندارند، این نوع کامنت می تواند به شما در شناسایی کدهای قدیمی کمک کند.

External scripts

اگر کد جاوا اسکریپت زیادی داریم، می توانیم آن را در یک فایل جداگانه قرار دهیم. فایل های اسکریپت با ویژگی `src` به HTML پیوست می شوند:

```
1 <script src="/path/to/script.js"></script>
```

در اینجا، `path/to/script.js/` یک مسیر مطلق به اسکریپت از `root` سایت است. همچنین می توان یک مسیر نسبی از صفحه فعلی ارائه کرد. برای مثال، `src="script.js"`، درست مانند `src="./script.js"` به معنای فایل `script.js` در پوشه فعلی است. ما می توانیم یک آدرس URL کامل نیز بدهیم. برای مثال:

```
1 <script src="https://cdnjs.cloudflare.com/ajax/libs/lodash.js/4.17.11/lodash.js"></script>
```

برای پیوست کردن چندین اسکریپت، از چندین تگ استفاده میکنیم:

```
1 <script src="/js/script1.js"></script>
2 <script src="/js/script2.js"></script>
3 ...
```

نکته:

به عنوان یک قاعده، فقط ساده ترین اسکریپت ها در HTML قرار می گیرند. موارد پیچیده تر در فایل های جداگانه قرار دارند. مزیت یک فایل جداگانه این است که مرورگر آن را دانلود کرده و در حافظه پنهان خود ذخیره (cache) می کند.

سایر صفحاتی که به همان اسکریپت اشاره می کنند به جای دانلود آن را از حافظه پنهان می گیرند، بنابراین فایل در واقع فقط یک بار دانلود می شود. که ترافیک را کاهش می دهد و صفحات را سریعتر می کند.

نکته:

اگر src تنظیم شده باشد، محتوای اسکریپت نادیده گرفته می شود. یک تگ <script> نمی تواند هم ویژگی src و هم کد را در داخل داشته باشد. این اسکریپت کار نخواهد کرد:

```
1 <script src="file.js">
2   alert(1); // the content is ignored, because src is set
3 </script>
```

ما باید یک <script src="..."> خارجی یا یک <script> معمولی با کد را انتخاب کنیم.

مثال بالا را می توان برای کار به دو اسکریپت تقسیم کرد:

```
1 <script src="file.js"></script>
2 <script>
3   alert(1);
4 </script>
```

خلاصه

ما می توانیم از تگ <script> برای افزودن کد جاوا اسکریپت به صفحه استفاده کنیم. ویژگی های نوع و زبان مورد نیاز نیست. یک اسکریپت در یک فایل خارجی را می توان با <script> src="path/to/script.js"></script> درج کرد.

چیزهای بیشتری برای یادگیری در مورد اسکریپت های مرورگر و تعامل آنها با صفحه وب وجود دارد. اما بیایید در نظر داشته باشیم که این بخش از آموزش به زبان جاوا اسکریپت اختصاص دارد، بنابراین ما نباید حواس خود را با پیاده سازی های خاص مرورگر آن پرت کنیم. ما از مرورگر به عنوان راهی برای

اجرای جاوا اسکریپت استفاده خواهیم کرد، که برای خواندن آنلاین بسیار راحت است، اما تنها یکی از موارد است.