

## ساختار کد

اولین چیزی که ما مطالعه خواهیم کرد ساختن بلوک های کد است.

## عبارات

عبارات، ساختار سینتکس هستند که اعمال (action) را انجام می دهند. ما قبلاً یک عبارت هشدار Hello, world! را دیده ایم که پیام «سلام، دنیا!» را نشان می دهد. ما می توانیم به تعداد دلخواه در کد خود عبارت داشته باشیم. عبارات را می توان با سمی کالن از هم جدا کرد.

به عنوان مثال، در اینجا Hello World را به دو alert تقسیم می کنیم:

```
1 alert('Hello'); alert('World');
```

معمولاً عبارات در خطوط جداگانه برای خوانایی بیشتر کد نوشته می شوند:

```
1 alert('Hello');  
2 alert('World');
```

## سمی کالن

هنگامی که یک خط شکسته وجود دارد، ممکن است سمی کالن حذف شود. این اسکرپیت به درستی کار می کند:

```
1 alert('Hello')  
2 alert('World')
```

در اینجا، جاوا اسکرپیت شکست خط را به عنوان یک سمی کالن «ضمنی» تفسیر می کند. این درج خودکار سمی کالن نامیده می شود. در بیشتر موارد، خط جدید به معنای سمی کالن است. اما "در بیشتر موارد" به معنای "همیشه" نیست!

مواردی وجود دارد که خط جدید به معنای سمی کالن نیست. مثلاً:

```
1 alert(3 +  
2 1  
3 + 2);
```

خروجی کد 6 است زیرا جاوا اسکرپیت سمی کالن را در اینجا وارد نمی کند. به طور شهودی واضح است که اگر خط با علامت مثبت "+" خاتمه یابد، "عبارت ناقص" است، بنابراین سمی کالن در آنجا نادرست است. و در این مورد، همانطور که در نظر گرفته شده است، کار می کند.

اما موقعیت‌هایی وجود دارد که جاوا اسکریپت نمی‌تواند سعی کالن را در جایی که واقعاً مورد نیاز است، فرض کند. پیدا کردن و رفع خطاهایی که در چنین مواردی رخ می‌دهد بسیار دشوار است.

نمونه ای از خطا

اگر کنجکاو هستید که نمونه ای عینی از چنین خطایی را ببینید، این کد را بررسی کنید:

```
1 alert("Hello");
2
3 [1, 2].forEach(alert);
```

هنوز نیازی به فکر کردن در مورد معنی آکولاد ها [] و forEach نیست. بعداً آنها را مطالعه خواهیم کرد. در حال حاضر، فقط نتیجه اجرای کد را به خاطر بسپارید: Hello، سپس 1 و سپس 2 را نشان می‌دهد. حالا بیایید سعی کالن بعد از هشدار را برداریم:

```
1 alert("Hello")
2
3 [1, 2].forEach(alert);
```

تفاوت در مقایسه با کد بالا فقط یک کاراکتر است: سعی کالن در انتهای خط اول از بین رفته است. اگر این کد را اجرا کنیم، فقط اولین Hello نشان داده می‌شود (و خطایی وجود دارد، ممکن است لازم باشد کنسول را باز کنید تا آن را ببینید). دیگر هیچ عددی را نمی‌بینید.

این به این دلیل است که جاوا اسکریپت یک سعی کالن قبل از آکولاد [...] فرض نمی‌کند. بنابراین، کد در مثال آخر به عنوان یک دستور واحد در نظر گرفته می‌شود. در اینجا نحوه مشاهده موتور آن است:

```
1 alert("Hello")[1, 2].forEach(alert);
```

عجیب به نظر می‌رسد، درست است؟ چنین ادغامی در این مورد فقط اشتباه است. باید بعد از alert یک سعی کالن بگذاریم تا کد به درستی کار کند. این می‌تواند در موقعیت‌های دیگر نیز اتفاق بیفتد.

توصیه می‌کنیم بین دستورات سعی کالن قرار دهید حتی اگر با خطوط جدید از هم جدا شده باشند. این قانون به طور گسترده توسط جامعه پذیرفته شده است. بیایید یک بار دیگر توجه کنیم - در بیشتر مواقع می‌توان سعی کالن را کنار گذاشت. اما استفاده از آنها به ویژه برای یک مبتدی امن‌تر است.

**نظرات**

با گذشت زمان، برنامه ها پیچیده تر و پیچیده تر می شوند. اضافه کردن کامنت (توضیحات) ضروری است که توضیح دهد کد چه کار می کند و چرا. کامنتها را می توان در هر مکانی از یک اسکریپت قرار داد. آنها بر اجرای برنامه تأثیر نمی گذارند زیرا موتور به سادگی آنها را نادیده می گیرد.

کامنت یک خطی با دو کاراکتر اسلش شروع می شود `//`. بقیه خط یک کامنت یا توضیح است. ممکن است یک خط کامل را اشغال کند یا از یک عبارت دنبال کند. مثل اینجا:

```
1 // This comment occupies a line of its own
2 alert('Hello');
3
4 alert('World'); // This comment follows the statement
```

نظرات چند خطی با یک اسلش جلو و یک ستاره `/*` شروع می شود و با یک ستاره و یک اسلش جلو `*/` پایان می یابد. مثل این:

```
1 /* An example with two messages.
2 This is a multiline comment.
3 */
4 alert('Hello');
5 alert('World');
```

محتوای نظرات نادیده گرفته می شود، بنابراین اگر کد را در داخل `/* ... */` قرار دهیم، اجرا نمی شود. گاهی اوقات غیرفعال کردن موقت بخشی از کد می تواند مفید باشد:

```
1 /* Commenting out the code
2 alert('Hello');
3 */
4 alert('World');
```

### از کلیدهای میانبر استفاده کنید!

در اکثر ویرایشگرها، یک خط کد را می توان با فشار دادن کلید میانبر `Ctrl+/` برای یک نظر تک خطی و چیزی مانند `Ctrl+Shift+/` برای کامنت چند خطی (تکه ای از کد را انتخاب کنید و کلید میانبر را فشار دهید) توضیح داد. برای مک، `Cmd` را به جای `Ctrl` و `Option` را به جای `Shift` امتحان کنید.

### نظرات تو در تو پشتیبانی نمی شوند!

ممکن است `/*...*/` در `/*...*/` دیگری وجود نداشته باشد. چنین کدی با یک خطا از بین می رود:

```
1  /*
2    /* nested comment ?! ? */
3  */
4  alert( 'World' );
```

لطفا در کامنت کد خود تعلل نکنید.

کامنت پیگیری کلی کد را افزایش می دهند، اما این اصلاً مشکلی نیست. ابزارهای زیادی وجود دارند که کد را قبل از انتشار در سرور کوچک می کنند. آنها نظرات را حذف می کنند، بنابراین در اسکریپت های کاری ظاهر نمی شوند. بنابراین کامنت ها اصلاً تاثیر منفی روی تولید ندارند.