

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

خواص اشیا جاوا اسکریپت (object properties)

مدرس : مهندس افشین رفوآ

[دوره آموزش JQuery](#)

[دوره آموزش JavaScript](#)

خواص اشیا جاوا اسکریپت (object properties)

خواص یا به انگلیسی همان **properties** مهمترین بخش یک شی محسوب می شوند.

خواص (**properties**) مقادیری متصل به/مرتبط با شی جاوا اسکریپت می باشند.

یک شی عبارتند از مجموعه ای نامرتب از خاصیت ها.

خواص را اغلب می توان اصلاح کرد، اضافه کرد و حتی در صورت نیاز آن ها حذف کرد. اما چنانچه **property**

مورد نظر از نوع **readonly** (فقط خواندنی و ثابت) بود، دیگر امکان دستکاری آن وجود ندارد.

دسترسی به خواص جاوا اسکریپت

دستور نگارشی (**syntax**) مورد نیاز برای دسترسی به خاصیت یک شی به ترتیب زیر می باشد:

```
objectName.property // person.age
```

یا

```
objectName["property"] // person["age"]
```

و یا

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 561 - واحد 7

88146323 - 88446780 - 88146330

```
objectName[ expression] // x = "age"; person[x]
```

حلقه ی for...in

دستور **for...in** برای حرکت (تکرار) در درون اعضای یک آرایه یا مجموعه **property** های یک **object** استفاده می شود.

ساختار نگارشی:

```
for ( variable in object) {  
    code to be executed  
}
```

قطعه کد (**code block**) درون حلقه ی **for...in** در حقیقت یکبار به ازای هر یک از خاصیت های شی مورد نظر اجرا می شود یا به عبارتی به ازای خواندن هریک از **property** های شی، یکبار دستورات درون حلقه اجرا می گردد.

مثال:

```
var person = {fname:"John", lname:"Doe", age:25};
```

```
for (x in person) {  
    txt += person[x];  
}
```

افزودن property های جدید

می توان با تخصیص یک مقدار جدید به شی موجود، خاصیت های جدید به آن اضافه کنید.

فرض بگیرید که شی **person** از پیش موجود می باشد – حال می خواهید خواص جدید به آن اضافه کنید:

```
person.nationality = "English";
```

توجه داشته باشید که نمی توان از کلمات رزرو شده به عنوان اسم برای خاصیت یا متد بهره جست. در این زمینه باید از قواعد و دستورات عمل های زبان جاوا اسکریپت پیروی کرد.

حذف کردن خواص

به وسیله ی کلیدواژه ی **delete** می توان یکی از **property** های شی مد نظر را پاک کرد.

مثال:

```
var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};  
delete person.age; // or delete person["age"];
```

لازم به ذکر است که کلیدواژه ی مزبور نه تنها مقدار **property** بلکه خود آن را نیز حذف می کند.

پس از حذف، برای استفاده ی مجدد از **property** پاک شده، لازم است (پیش از اینکه بتوان خاصیت را مورد استفاده قرار داد) آن را دوباره به شی اضافه کنید.

عملگر **delete** برای کار با خواص شی (حذف آن ها) طراحی شده است و هیچ تاثیری بر روی متغیرها و توابع ندارد.

توجه داشته باشید که نباید عملگر **delete** را بر روی خاصیت های از پیش تعریف شده ی شی جاوا اسکریپت بکار برد. این کار ممکن است منجر به از کار افتادن یا خرابی برنامه (**application**) شما شود.

Property attribute (خصیصه های property)

کلید ی خاصیت ها (**property**) علاوه بر اسم، دارای یک مقدار هستند.

مقدار در واقع یکی از خصیصه های **property** حساب می شود.

خصیصه های دیگری نیز وجود دارند. این خصیصه ها عبارتند از: **enumerable**، **configurable** و همچنین **writable**.

خصیصه های ذکر شده نحوه ی دسترسی به خاصیت های یک شی را تعریف می کنند (برای مثال آیا فقط قابل خواندن/readable است یا writable و قابل نوشتن).

کلیده ی خصیصه ها در جاوا اسکریپت قابل خواندن هستند، اما تنها خصیصه ی (attribute) value تغییر پذیر بوده و قابل اصلاح می باشد (آن هم فقط در صورتی که خاصیت writable باشد).

پنجمین ویرایش جاوا اسکریپت (ECMAScript 5) توابعی دارد که می توان برای بازیابی (get) و مقداردهی (set) کلیده ی خصیصه های property از آن ها استفاده کرد.

خواص نمونه ی اولیه (prototype properties)

در زبان جاوا اسکریپت، کلیده ی اشیا خاصیت های نمونه ی اولیه (prototype) خود را به ارث می برند.

کلیدواژه ی delete خواص به ارث برده شده را حذف نمی کند، اما چنانچه به هر دلیلی خاصیت یک نمونه ی اولیه (prototype property) را پاک کردید، در آن صورت تمامی اشیا به ارث گرفته شده از نمونه ی اولیه ی مورد نظر تحت تاثیر قرار می گیرند.