

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

اشیا، متدها، خصوصیات javascript

مدرس: مهندس افشین رفوآ

اشیا، متدها، خصوصیات javascript

اشیا، متدها، خصوصیات

در جهان واقع اتومبیل یک **object** یا شی محسوب می شود.

یک اتومبیل دارای **property** یا خصوصیت هایی از قبیل وزن، رنگ و همچنین توابع یا **method** هایی از قبیل **start** و **stop** دارد.

متدها	خصوصیات	شی
car.start()	car.name = Fiat	
car.drive()	car.model = 500	
car.brake()	car.weight = 850kg	
car.stop()	car.color = white	

تمامی اتومبیل ها دارای خصوصیات یکسان هستند، اما مقادیر **property** آن ها از اتومبیلی به اتومبیل دیگر ممکن است متفاوت باشد.

کلید ی اتومبیل ها دارای متدهای یکسان هستند، اما متدهای آن ها در زمان های مختلف اجرا می شوند.

اشیا جاوا اسکریپت

پیش تر آموختیم که متغیرها در واقع ظرف هایی برای ذخیره ی مقادیر داده محسوب می شوند.

آدرس آموزشگاه. تهران - خیابان شریعی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 561 - واحد 7
88146323 - 88446780 - 88146330

کد زیر یک مقدار ساده (Fiat) به متغیری به نام car تخصیص می دهد.

```
var car = "Fiat";
```

اشیا نیز متغیر (variable) هستند، با این تفاوت که می توانند مقادیر زیادی را در خود جای دهند.

به عنوان مثال، توجه خود را به نمونه ی زیر جلب کنید که در آن چندین مقدار متفاوت (Fiat ,500 , white) به متغیری به نام car اختصاص داده شده.

```
var car = {type:"Fiat", model:500, color:"white"};
```

همان طور که مشاهده می کنید مقادیر به صورت جفت های اسم : مقدار (name:value pairs) نوشته شده اند.

توجه

اشیا جاوا اسکریپت در حقیقت ظرف هایی برای مقادیر نام گذاری شده (named variables) هستند.

خصوصیات اشیا (object properties)

جفت های اسم : مقدار (name:values pairs) خصوصیات یا properties یک شی در جاوا اسکریپت اطلاق می گردند.

```
var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};
```

Property Value / مقدار خصوصیت	Property / خصوصیت
John	firstName
Doe	lastName
50	age
blue	eyeColor

متدهای یک شی (object methods)

متدها در واقع عملیاتی هستند که می توان روی شی انجام داد.

متدها به عنوان **function definitions** یا تعریف تابع در خصوصیات (**property**) ذخیره می گردند (می توان متدها را مانند تعریف یک تابع داخل خصوصیت ذخیره کرد).

Property	Property Value
firstName	John
lastName	Doe
age	50
eyeColor	blue
fullName	function() {return this.firstName + " " + this.lastName;}

مهم

اشیا جاوا اسکریپت ظرف هایی برای نگهداری مقادیر نام گذاری شده (که خصوصیت یا **property** خوانده می شود) و متدها هستند.

تعریف یا ایجاد شی در جاوا اسکریپت

یک شی جاوا اسکریپت را می توان با **object literal** (نوشته ی متنی) تعریف کرد.

مثال

```
var person = {firstName:"John", lastName:"Doe", age:50, eyeColor:"blue"};
```

خط فاصله ها و استفاده از **enter** برای رفتن به خط بعدی اهمیتی نداشته و توسط جاوا اسکریپت نادیده گرفته می شود، به عبارتی دیگر تعریف شی می تواند چندین خط ادامه داشته باشد (در چندین خط قرار گرفته باشد).

مثال

```
var person = {  
  firstName:"John",  
  lastName:"Doe",
```

```
    age:50,  
    eyeColor:"blue"  
};
```

دسترسی به خصوصیت های اشیا

از دو طریق می توان به خصوصیت اشیا دسترسی پیدا کرد.

```
objectName.propertyName
```

یا

```
objectName[propertyName]
```

مثال 1

```
person.lastName;
```

مثال 2

```
person["lastName"];
```

دسترسی به متدهای یک شی

می توان با استفاده از دستور نحوی (**syntax**) زیر به متد یک شی دسترسی پیدا کرد.

```
objectName.methodName()
```

مثال

```
name = person.fullName();
```

چنانچه سعی کنید به خصوصیت / **property**، **fullName** بدون عملگر **()** دسترسی پیدا کنید، در آن صورت تعریف تابع / **function definition** به عنوان نتیجه بازگردانده می شود.

مثال

```
name = person.fullName;
```

توجه داشته باشید که هیچگاه نباید رشته ها (**strings**)، اعداد (**numbers**)، بولی ها (**Booleans**) را به عنوان شی تعریف کنید!

چنانچه متغیری را در جاوا اسکریپت با کلیدواژه ی **"new"** تعریف کنید، آن متغیر به عنوان یک شی ایجاد می گردد.

```
var x = new String();           // Declares x as a String object
var y = new Number();          // Declares y as a Number object
var z = new Boolean();         // Declares z as a Boolean object
```

از اشیا بولی، عددی، رشته ای اجتناب کنید، زیرا که کد را پیچیده تر ساخته و سرعت اجرا را کاهش می دهد.

www.tahlildadeh.com