

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

متدهای مربوط به شی data در جاوا اسکریپت

مدرس: مهندس افشین رفوآ

متد های مربوط به شی data در جاوا اسکریپت

متدهای شی **date** به برنامه نویس اجازه ی بدست آوردن و تنظیم (مقدار) تاریخ و زمان را می دهد.

متدهای **get** (برگرداندن مقدار تاریخ و زمان)

شرح کاربرد	تابع
شماره ی روز جاری را در ماه جاری برمی گرداند (1-31)	getDate()
شماره ی روز جاری در هفته را برمی گرداند (0-6)	getDay()
شماره ی (چهار رقمی) سال جاری را بر می گرداند (سال سال سال سال)	getFullYear()
مقدار ساعت را برمی گرداند (0-23)	getHours()
مقدار (شماره ی) میلی ثانیه را برمی گرداند (0-999)	getMilliseconds()
شماره ی دقیقه را در یک متغیر زمانی برمی گرداند (0-59)	getMinutes()
مقدار ماه (شماره ی ماه جاری) را برمی گرداند (0-11)	getMonth()
شماره ی ثانیه ی جاری را بر می گرداند (0-59)	getSeconds()
هزارم ثانیه هایی که از تاریخ یک ژانویه ی سال 1970 تا کنون سپری شده را برمی گرداند	getTime()

متد **getTime()**

این تابع تعداد ثانیه های سپری شده از تاریخ اول ژانویه سال **1970** تا به امروز را بر می گرداند :

01.01.1970

مثال

```
<script>
var d = new Date();
document.getElementById("demo").innerHTML = d.getTime();
</script>
```

تابع getFullYear()

این تابع شماره ی سال جاری را (به صورت چهار رقمی) بر می گرداند.

مثال

```
<script>
var d = new Date();
document.getElementById("demo").innerHTML = d.getFullYear();
</script>
```

تابع getDay()

شماره ی روز جاری در هفته را برمی گرداند (0-6).

مثال

```
<script>
var d = new Date();
document.getElementById("demo").innerHTML = d.getDay();
</script>
```

نکته

جاوا اسکریپت اولین روز هفته (0) را یکشنبه در نظر می گیرد.

می توان با استفاده از آرایه ای از اسم ها و تابع `getDay()`، روزهای هفته را با اسم شان برگرداند.

مثال

```
<script>
```

```

var d = new Date();
var days = ["Sunday","Monday","Tuesday","Wednesday","Thursday","Friday","Saturday"];
document.getElementById("demo").innerHTML = days[d.getDay()];
</script>

```

متدهای تنظیم (set) تاریخ

شرح	تابع
از این متد برای تغییر مقدار عددی تاریخ روز در یک متغیر زمانی استفاده می شود (مقدار روز را به صورت عددی تنظیم می کند) (1-31)	setDate()
شماره یا مقدار سال و در صورت نیاز روز و ماه را تنظیم می کند	setFullYear()
مقدار ساعت را تنظیم می کند (0-23)	setHours()
مقدار عددی میلی ثانیه (هزارم ثانیه) را تنظیم می کند (0-999)	setMilliseconds()
مقدار عددی دقیقه در یک متغیر زمانی را تنظیم می کند (0-59)	setMinutes()
به منظور اصلاح و تنظیم شماره / مقدار ماه بکار می رود (0-11)	setMonth()
مقدار ثانیه را تنظیم می کند (تغییر مقدار عددی ثانیه در یک متغیر زمانی) (0-59)	setSeconds()
محاسبه و ایجاد یک تاریخ جدید ، به واسطه ی اضافه یا کم کردن تعداد مشخصی هزارم ثانیه به تاریخ اول ژانویه سال 1970 استفاده می شود / مقدار زمان را از تاریخ یک ژانویه ی سال 1970 بر حسب میلی ثانیه محاسبه و تنظیم می کند)	setTime()

تابع setFullYear()

این متد شی **date** را روی تاریخ مشخصی تنظیم می کند، در مثال زیر تاریخ به ژانویه ی 14، 2020 تنظیم شده است.

مثال

```
<script>
var d = new Date();
d.setFullYear(2020, 0, 14);
document.getElementById("demo").innerHTML = d;
</script>
```

متد setDate()

تابع `setDate()` روز ماه مربوطه را تنظیم (مقداردهی) می کند (1-31).

مثال

```
<script>
var d = new Date();
d.setDate(20);
document.getElementById("demo").innerHTML = d;
</script>
```

با استفاده از این متد همچنین می توان روز به یک تاریخ قبلی افزود.

مثال

```
<script>
var d = new Date();
d.setDate(d.getDate() + 50);
document.getElementById("demo").innerHTML = d;
</script>
```

چنانچه اضافه کردن روز باعث شود ماه یا سال تغییر پیدا کنند، شی `date` خود به صورت اتوماتیک تغییرات وارد آمده را مدیریت می کند.

ورودی date – تجزیه (parse) date

چنانچه یک رشته ی معتبر **date** (به صورت رشته یا **string**) دارید که می خواهید به میلی ثانیه تبدیل شود، در آن صورت می توانید از متد **Date.parse()** استفاده کنید.

تابع **Date.parse()** تعداد میلی ثانیه های (سپری شده) بین تاریخ مورد نظر و تاریخ اول ژانویه ی سال **1970** را برمی گرداند.

مثال

```
<script>
var msec = Date.parse("March 21, 2012");
document.getElementById("demo").innerHTML = msec;
</script>
```

در مرحله ی بعد می توانید با استفاده از مقدار بازگشتی رشته را به یک شی **date** تبدیل کنید.

مثال

```
<script>
var msec = Date.parse("March 21, 2012");
var d = new Date(msec);
document.getElementById("demo").innerHTML = d;
</script>
```

مقایسه ی تاریخ ها با یکدیگر

تاریخ ها را می توان به راحتی با یکدیگر مقایسه کرد.

مثال زیر تاریخ امروز را با تاریخ **2100,January 14** مقایسه می کند.

```
var today, someday, text;

today = new Date();
someday = new Date();
someday.setFullYear(2100, 0, 14);

if (someday > today) {
```

```
        text = "Today is before January 14, 2100.";
    } else {
        text = "Today is after January 14, 2100.";
    }
    document.getElementById("demo").innerHTML = text;
```

www.tahlildadeh.com