

به نام خداوند بخشنده مهربان

آموزش کار با jQuery

# jQuery



تالیف و گردآوری :

مهرداد فتاحی

Developer Studio Network

برای دریافت آموزش های بیشتر و سایر مطالب به وب سایت ما سر بزنید .

<http://www.Developer1.ir>

1391

## ه و معرفی jQuery

### jQuery چیست ؟

jQuery یک کتابخانه استاندارد و آماده شامل تعدادی متد و توابع نوشته شده به زبان JavaScript می باشد .  
متدهای این کتابخانه برای کار با عناصر و تگ های موجود در صفحات ASP.Net , HTML و یا PHP استفاده می شود .  
به طور کلی با استفاده از این متدهای از پیش تعریف شده ، می توانید تغییراتی را در عناصر و تگ های HTML ایجاد کرده و یا نحوه نمایش آنها را تغییر دهید . برای مثال می توانید یک عنصر HTML را با کلیک بر روی یک دکمه فرمان پنهان کرده و یا در صورت مخفی بودن آن را نمایش دهید .

jQuery دارای یک شعار معروف است . این شعار می گوید کمتر بنویسید ، بیشتر انجام دهید ( write less , do more ) .

به طور کلی کتابخانه jQuery شامل موارد زیر می باشد :

jQuery شما نیاز ندارید برای انجام کارهای مورد نظرتان یک تابع جدید بنویسید . فقط کافی است عنصر مورد

نظر خود را انتخاب کرده و سپس متد مورد نظر را فراخوانی نموده تا آن عنصر را تغییر دهد .

قبل از مطالعه و استفاده از بخش jQuery ، شما بایستی با مطالب زیر آشنایی کامل داشته باشید :

• [HTML](#)

• [CSS](#)

• [JavaScript](#)

---

### نحوه استفاده از کتابخانه jQuery :

همانطور که گفتیم ، jQuery دارای یک کتابخانه استاندارد و آماده می باشد . شما بایستی برای استفاده از jQuery ، فایل این

کتابخانه را به صفحه خود متصل کنید . کتابخانه jQuery در یک فایل مجزا که از نوع فایل های JavaScript

jQuery.js است ، نگهداری می شود . شما برای اتصال فایل این کتابخانه به صفحه مورد نظرتان از روش می توانید

استفاده کنید :

• دانلود فایل کتابخانه jQuery بر روی کامپیوتر و یا سرور سایت خودتان و سپس اتصال آن به صفحه .

• از فایل کتابخانه jQuery بر روی سرورهای آنلاین مثل سرور Microsoft یا Google .

در ادامه به معرفی روش فوق می پردازیم .

: در روش اول شما بایستی جدیدترین فایل کتابخانه jQuery را از سایت [www.jquery.com](http://www.jquery.com) دانلود کرده و سپس

آن را در پوشه اصلی سایت خود قرار دهید . سپس با استفاده از یک تگ `< Script >` که در قسمت `< Head >` صفحه قرار

می دهید ، فایل کتابخانه را به صفحه متصل کنید . در قسمت زیر این روش را نشان داده ایم :

Syntax	<pre>&lt; head &gt;   &lt; script type = " text/javascript " src="jquery.js" &gt; &lt;/script &gt; &lt;/head &gt;</pre>
--------	---

: در روش دوم ، شما لینک دسترسی به فایل کتابخانه jQuery ، که بروی سرور مورد نظرتان است را در یک [< Script >](#) [< Head >](#) صفحه قرار می دهید . در این حالت صفحه در هنگام لود شدن ، به سرور وصل شده و از کتابخانه آنلاین استفاده می کند . در قسمت زیر هر لینک و طریقه استفاده از آنها را نمایش داده ایم :

Syntax ( مایکروسافت )	<pre>&lt; head &gt;   &lt; script type = " text/javascript " src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js" &gt; &lt;/script &gt; &lt;/head &gt;</pre>
-----------------------	---

Syntax ( )	<pre>&lt; head &gt;   &lt; script type = " text/javascript " src="http://ajax.googleapis.com/ajax/libs/jquery/1.4.2/jquery.min.js" &gt; &lt;/script &gt; &lt;/head &gt;</pre>
------------	---

پس از اتصال کتابخانه jQuery به صفحه خودتان ، هر زمان که یک متد jQuery را فراخوانی می کنید ، برنامه سورس آن متد را از کتابخانه دریافت کرده و سپس اجرا می نماید . در مثال بخش بعد به روش کلی کار با توابع jQuery و یک نمونه از کاربرد آن اشاره می نمایم .

### یک مثال از کاربرد jQuery :

jQuery کاربردهای جالب و فراوانی دارد و به طراحان وب توانایی های زیادی را برای خلق جلوه ها و دینامیک کردن صفحات می دهد . در مثال زیر روش کلی استفاده از ساختار jQuery را نشان داده و یکی از کاربردهای مهم آن یعنی پنهان کردن عناصر موجود بر روی صفحه را نمایش می دهیم .

در مثال زیر یک صفحه فرضی با پاراگراف و یک تیتیر داریم . می خواهیم تا کاربر بتواند با کلیک بر روی یک دکمه فرمان پاراگراف های موجود در صفحه را پنهان کند . روش کار به این شرح است :

در ابتدا به وسیله یکی از لینک های معرفی شده ، صفحه خود را به کتابخانه jQuery متصل کرده ایم . سپس یک دکمه فرمان و یک تیتیر و پاراگراف را بروی صفحه ایجاد کرده ایم . با کلیک کاربر بر روی دکمه فرمان ، متد ( Hide ) که یکی از

متدهای jQuery است ، اجرا شده و باعث می شود تا تمام پاراگراف های ( تگ های < p > ) موجود بر روی صفحه پنهان . همانطور که در خروجی مشاهده می کنید تابع ایجاد شده فقط بر روی عناصری که تگ آنها از نوع < p > ( است ، اثر می گذارد و مثلا با تگ تیتل < h2 > کاری ندارد . برای مشاهده عملکرد مثال ، بر روی دکمه فرمان خروجی کلیک نمایید .

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ("button") . click ( function ( ){         \$ ( " p " ) . hide ( );       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; h2 &gt; This is a heading &lt; /h2 &gt;   &lt; p &gt; This is a paragraph . &lt; /p &gt;   &lt; p &gt; This is another paragraph. &lt; /p &gt;   &lt; button &gt; Click me &lt; /button &gt; &lt; /body &gt; &lt; /html &gt; </pre>	کد
<p><b>This is a heading</b></p> <p>This is a paragraph .</p> <p>This is another paragraph.</p> <p>Click me</p>	خروجی

## ساختار دستوری jQuery

### ساختار دستوری jQuery :

در قسمت مقدمه ، شما را با یک مثال از زبان jQuery آشنا کردیم . در این قسمت قصد داریم تا ساختار کلی jQuery را به شما آموزش دهیم .

jQuery شما بیستی ابتدا عنصر یا عنصری که می خواهید در صفحه تغییر دهید ، را انتخاب کنید .

را بر روی آن اجرا نمایید . ساختار کلی دستور jQuery به صورت زیر است :

Syntax	<code>\$ ( Selector ) . action ( )</code>
--------	---

در ادامه به توضیح مفهوم ساختار فوق می پردازیم :

**\$ :** این علامت ، علامت خاص زبان jQuery . این علامت را بایستی در ابتدای هر خط دستوری زبان jQuery قرار دهید . به این وسیله مرورگر تشخیص می دهد که کد نوشته شده ، به زبان \$ است و باید چگونه آن را اجرا نماید .

**انتخاب کننده ( Selector ) :** در قسمت دوم شما باید عنصر و یا عناصر مورد نظر خود را به وسیله انواع انتخاب کننده ( Selector ) انتخاب نمایید . روش برای انتخاب عنصر مورد نظران وجود دارد . انتخاب خود عنصر جاری ، انتخاب بر حسب نوع عنصر ، انتخاب بر حسب کلاس ( class ) عنصر و یا انتخاب بر حسب شناسه ( Id ) . در مثال های زیر انواع این حالت را نمایش می دهیم . در این مثال ها از متد ( Hide ) که باعث پنهان شدن عنصر می شود ، استفاده شده است . در بخش بعدی به طور کامل تر به نحوه انتخاب عناصر در jQuery می پردازیم . مثال های این بخش صرفا برای آشنایی با ساختار این زبان است .

**انتخاب خود عنصر جاری :** این کد باعث می شود تا متد مورد نظر بر روی عنصر جاری اجرا شده و آن را پنهان می کند.

Syntax	<code>\$ ( this ) . hide ( )</code>
--------	-------------------------------------

**انتخاب بر حسب نوع کلی عنصر :** این کد باعث می شود تا متد مورد نظر بر روی کلیه عناصر از نوع تعیین شده . در مثال زیر اجرا متد کلیه عناصری که از نوع پاراگراف ( < p > ) هستند را پنهان می کند.

Syntax	<code>\$ ( "p" ) . hide ( )</code>
--------	------------------------------------

. انتخاب بر حسب کلاس ( class ) : این کد باعث می شود تا متد مورد نظر بر روی کلیه عناصر از کلاس تعیین شده اجرا . در مثال زیر اجرا متد کلیه عناصری را که از نوع پاراگراف بوده و جز کلاس ( test ) هستند را پنهان می کند.

Syntax	<code>\$ ( "p.text" ) . hide ( )</code>
--------	---

. انتخاب بر حسب شناسه ( Id ) : این کد باعث می شود تا متد مورد نظر بر روی کلیه عناصری که شناسه انحصاری آن ( Id ) تعیین شده اجرا شود . در مثال زیر اجرا متد عناصری را که Id آن برابر با و جز کلاس ( test ) (می باشد را پنهان می کند).

Syntax	<code>\$ ( "#.test" ) . hide ( )</code>
--------	---

: ( action ) : در قسمت سوم شما متدی را که می خواهید بر روی عنصر یا عناصر مورد نظرتان اجرا شود تعیین می کنید .

### : document.ready

این تابع را باید یکبار در ابتدای کلیه دستورات jQuery قرار دهید . قرار دادن این تابع باعث می شود تا قبل از لود ( load ) شدن کامل صفحه ( یا به اصطلاح ready ) ، هیچ یک از دستورات jQuery . این دستورات قبل از لود کامل صفحه می تواند برای مرورگر مشکل ساز شود . برای مثال ، مثال های ارائه شده در قسمت بالا را در نظر بگیرید . تا زمانی که صفحه کاملا لود نشود ، هیچ یک از عناصر تعیین شده هنوز وجود خارجی ندارند تا متد تعیین شده روی آنها انجام شود . این مسئله می تواند باعث شکست مرورگر شود . در کد زیر نحوه تعریف تابع document.ready تشریح شده است :

Syntax	<pre><code>\$ ( document ) . ready ( function ( ) { // را اینجا می نویسید jQuery کد های });</code></pre>
--------	--

### : یک مثال کاربردی :

پس از تعریف ساختار دستوری jQuery در یک مثال کلی نشان می دهیم .  
: در مثال زیر پاراگراف داریم که جز کلاس Ex\_1 هستند . با کد نوشته شده متد ( toggle ) را بر روی آنها اجرا کرده ایم . این متد عناصری که دیده می شوند را پنهان و عناصر پنهان را آشکار می کند . برای اجرای این متد باید بر روی کمه فرمان خروجی مثال کلیک نمایید :

### Example Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function( ) {
      $ ("button") . click ( function( ){
        $ ( " p.Ex_1 " ) . toggle( );
      });
    });
  </script>
</head>
<body>
  <p> This is paragraph 1 . </p>
  <p> This is paragraph 2 . </p>
  <button> Click me</button>
</body>
</html>
```

کد

This is paragraph 1 .

This is paragraph 2 .

Click me

خروجی

## انواع انتخاب کننده ها در JQuery

### انواع انتخاب کننده ها در JQuery :

در بخش قبل به صورت مختصر به روش انتخاب انواع عناصر در صفحات وب با استفاده از JQuery اشاره کردیم . در این بخش با نگاهی دقیقتر و مثال های عملی ، سعی می کنیم تا شما را با روش انتخاب دقیق هر یک از عناصر مورد نظرتان آشنا کنیم . البته تنوع روش های انتخاب عناصر خیلی زیاد است و ذکر تمام حالات در این بخش نمی گنجد ، ولی سعی می شود بیشتر حالت ها را پوشش دهیم . ضمن اینکه این روش ها را می توان با هم ترکیب کرد . JQuery این امکان را به شما می دهد تا از بین تمام عناصر موجود در یک صفحه ، دقیقا عنصر مورد نظر خود را بر حسب مشخصاتی همچون نوع عنصر ، کلاس ، شناسه و ... انتخاب کرده و متدهای مورد نظرتان را روی آنها انجام دهید .

- **انتخاب خود عنصر جاری :** این کد باعث می شود عنصر جاری انتخاب شود.

Syntax	<code>\$ ( this )</code>
--------	--------------------------

- **انتخاب بر حسب نوع کلی عنصر :** این کد باعث می شود تا تمام عناصر از نوع `< p >` ( ) .

Syntax	<code>\$ ( "p" )</code>
--------	-------------------------

- **انتخاب بر حسب کلاس ( class ) :** این کد باعث می شود تا کلیه عناصری که از نوع پاراگراف بوده و جزء کلا `text` هستند ، انتخاب شوند.

Syntax	<code>\$ ( "p.text" )</code>
--------	------------------------------

- **انتخاب بر حسب شناسه ( Id ) :** این کد باعث می شود تا عنصری که شناسه انحصاری ( id ) آن در صفحه `text` .

Syntax	<code>\$ ( "#text" )</code>
--------	-----------------------------

- **انتخاب یک تگ با استفاده از محتویات متنی درون آن :** با استفاده از این کد می توانید یک واژه یا متن را تعیین کرده و عنصر یا عنصرهایی را انتخاب کنید ، که آن کلمات در آنها وجود داشته باشند . برای مثال می خواهید از بین پاراگراف های موجود در یک صفحه پاراگرافی را انتخاب کنید که واژه PHP در آن به کار رفته است.



Syntax	\$( " : contains ( ) " ) example : \$( " p : contains ( PHP ) " )
--------	--

: در مثال زیر روش استفاده از کد بالا را به شما نشان داده ایم . این کد باعث می شود تا از بین دو پاراگرافی موجود در صفحه فرضی ، پاراگراف که کلمه PHP مخفی شود .

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt; \$(document). ready ( function( ) { \$( "button" ) . click ( function( ){ \$( " p:contains (PHP)" ) . hide(); }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt; p &gt; Java Script is a client side programing language &lt; /p &gt; &lt; p &gt; PHP is a server side programing language &lt; /p &gt; &lt;button&gt; Click me&lt;/button&gt; &lt;/body&gt; &lt;/html&gt;</pre>	کد
<p>Java Script is a client side programing language</p> <p>PHP is a server side programing language</p>	خروجی

- انتخاب عناصر بر حسب خواص آنها و مقدارشان : این کد باعث می شود تا تمام عناصری که دارای خاصیت href در دورن تگ خودشان هستند، انتخاب شوند.

Syntax	<code>\$ ( "[href]" )</code>
--------	------------------------------

- این کد باعث می شود تا تمام عناصری که مقدار خاصیت href آنها برابر با مقدار # .

Syntax	<code>\$ ( "[href]= jQuery;# jQuery; " )</code>
--------	---

- این کد باعث می شود تا تمام عناصری که مقدار خاصیت href آنها برابر با مقدار # نیست، انتخاب شوند.

Syntax	<code>\$ ( "[href] != jQuery;# jQuery; " )</code>
--------	---

- این کد باعث می شود تا تمام عناصری که مقدار خاصیت href آنها با مقدار jpg به پایان می رسد ، انتخاب شوند .

Syntax	<code>\$ ( "[href\$='.jpg']" )</code>
--------	---------------------------------------

- اساس مکان و یا ترتیب قرار گیری آنها : این کد باعث می شود تا آیتم `< li >` اول لسیت `< ol >` موجود در صفحه انتخاب شود.

Syntax	<code>\$ ( "ol li:first" )</code>
--------	-----------------------------------

- در مثال زیر این مسئله را به صورت عملی نشان می دهیم . در این مثال یک لیست ترتیبی با آیتم داریم . کدی طراحی کردیم که با اجرا شدن آن ، فونت آیتم اول به صورت bold در خواهد آمد . برای اجرای دستور بر روی دکمه فرمان کلیک کنید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ("button") . click ( function ( ){         \$ ( " ol li:first " ) . css( "font-weight","bold" ) );       } );     } );   &lt; /script &gt;</pre>	کد

<pre> }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;ol&gt; &lt;li&gt; HTML &lt;/li&gt; &lt;li&gt; CSS &lt;/li&gt; &lt;li&gt; Java Script &lt;/li&gt; &lt;/ol&gt; &lt;button&gt; Click me&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<ol style="list-style-type: none"> <li>1. HTML</li> <li>2. CSS</li> <li>3. Java Script</li> </ol>	<p style="text-align: center;">خروجی</p>

- انتخاب تمام تگ های هدر موجود در صفحه : این کد باعث می شود تا تمام تگ های هدر ( h1 ) ( h6 ) صفحه انتخاب شود.

Syntax	\$( " : header " )
--------	--------------------

: در مثال زیر روش استفاده از کد بالا را به شما نشان داده ایم . فرض کنید دو تا تگ هدر در صفحه وجود دارد . کدی طراحی کردیم که در هنگام اجرا تمام تگ های هدر موجود در صفحه را انتخاب کرده و رنگ آنها را به آبی تغییر می دهد :

Example	
<pre> &lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt; </pre>	کد

<pre> \$ (document). ready ( function ( ) {     \$ ("button") . click ( function ( ){         \$ ( " : header " ) . css( "color","Blue" ) );     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt; h1 &gt; HTML &lt; /h1 &gt; &lt; h4 &gt; Java Script &lt; /h4 &gt; &lt;button&gt; Click me&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<b>HTML</b>  <b>Java Script</b>	خروجی

- انتخاب اولین نمونه از یک نوع تگ در صفحه : این کد باعث می شود تا اولین نمونه موجود از یک تگ در صفحه

Syntax	<pre> \$( " :first " ) example : \$( " p:first " ) </pre>
--------	---

: در مثال زیر روش استفاده از کد بالا را به شما نشان داده ایم . این کد باعث می شود تا از بین دو پاراگراف موجود در صفحه فرضی ، پاراگراف اول انتخاب شده و فونت آن را به یک فونت دیگر تغییر کند .

<b>Example Example</b>	
<pre> &lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; </pre>	کد

<pre> &lt; script type="text/javascript" &gt;   \$ (document). ready ( function( ) {     \$ ("button") . click ( function( ){       \$ ( " p : first " ) . css( "font-family","Arial" ) );     });   }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt; p &gt; HTML &lt; /p &gt;   &lt; p &gt; Java Script &lt; /p &gt;   &lt;button&gt; Click me&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>HTML</p> <p>Java Script</p>	<p>خروجی</p>

- انتخاب آخرین نمونه از یک نوع تگ در صفحه : : این کد باعث می شود تا آخرین نمونه موجود از یک تگ در صفحه انتخاب شود . در واقع در کد و مثال های قبلی کافی است واژه کلیدی first را به last تبدیل کنید.

<p>Syntax</p>	<pre> \$( "      :last " ) example : \$( " p:last " ) </pre>
---------------	--

## رویدادها ( events ) jQuery

### رویدادها ( events ) jQuery :

رویدادهای jQuery ، متدهایی هستند که در هنگام اجرای یک واقعه در HTML مثل کلیک بر روی یک عنصر یا عبور موس از روی آن ، فراخوانی و اجرا می شوند .

برای مثال شما یک رویداد jQuery طراحی می کنید که در هنگام عبور موس از روی یک پاراگراف ، رنگ پس زمینه آن تغییر کند . رویدادهای jQuery به شما ابزاری می دهند که بوسیله آنها می توانید خصوصیات عناصر مثل رنگ ، پس زمینه و یا قلم نوشته را تغییر داده یا آنها را مخفی و سپس آشکار کنید .

syntax یا شکل کلی نوشتن و استفاده از رویدادهای jQuery در صفحات وب به صورت زیر است . در این syntax هر یک از اجزاء را به صورت مجزا توضیح داده ایم :

Syntax	<code>\$ ( " selector " ).event ( function ( ) { ... some code ... } ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
\$ در ابتدای کلیه دستورات jQuery قرار می گیرد و به مرورگر می فهماند که این دستور یک کد jQuery .	\$
selector عنصری است که رویداد لازم برای اجرای متد روی آن صورت می گیرد . مثلا کاربر بر روی آن کلیک کرده و یا نشانه گر موس از روی آن عبور می کند .	selector
این رویداد اتفاقی است که باید بر روی selector اتفاق بیفتد تا متد مورد نظر ما اجرا شود . مثلا کاربر بر روی دکمه فرمان کلیک کرده و یا موس از روی عنصر selector عبور کند . آنها کد موجود در function اجرا می شود .	event
این تابع کد دستوری است که می خواهیم در هنگام اجرای متد ، انجام شود .	function ( )

: در مثال زیر یک نمونه کد نوشته شده با syntax بالا را نشان داده ایم . در این کد در هنگام کلیک ( event ) بر روی

دکمه فرمان ( selector ) ( کد تابع و متد آن ) مخفی می شود :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function( ) {
      $ ("button") . click ( function( ){
        $ ( " p " ) . hide( );
      });
    });
  < /script >
< /head >
< body >
  < p > example paragraph . click to hide it ! < /p >
  < button > Click me < /button >
< /body >
< /html >
```

کد

example paragraph . click to hide it !

Click me

خروجی

## لیست رویدادها ( events ) jQuery

### لیست رویدادها ( events ) jQuery :

در این بخش به معرفی مهمترین و پرکاربردترین رویدادهای زبان jQuery می پردازیم . در بخش قبل گفتیم رویدادها اتفاقاتی هستند که بر روی عناصر موجود در صفحات وب انجام می شوند مثل کلیک کردن و یا عبور موس از روی آن . شما می توانید کدهای مورد نظر خود را طوری طراحی کنید که پس از اجرای یک رویداد خاص اجرا شوند .

لیست زیر شامل مهمترین رویدادهای jQuery می باشد . برای دریافت اطلاعات و مشاهده مثال های عملی روی نام هر کدام کلیک نمایید :

نام رویداد	
<a href="#">blur ( )</a>	این رویداد زمانی اتفاق می افتد که عنصر مورد نظر فوکوس برنامه را از دست می دهد .
<a href="#">change ( )</a>	این رویداد زمانی اتفاق می افتد که در عنصر مورد نظر تغییری ایجاد می شود .
<a href="#">click ( )</a>	این رویداد زمانی اتفاق می افتد که کاربر بر روی عنصر مورد کلیک می نماید .
<a href="#">dblclick ( )</a>	این رویداد زمانی اتفاق می افتد که کاربر بر روی عنصر مورد دابل کلیک می نماید .
<a href="#">focus ( )</a>	این رویداد زمانی اتفاق می افتد که عنصر مورد فوکوس برنامه را در اختیار می گیرد .
<a href="#">focusin ( )</a>	این رویداد زمانی اتفاق می افتد یک عنصر زیر مجموعه ( child object ) فوکوس برنامه را در اختیار می گیرد .
<a href="#">focusout ( )</a>	این رویداد زمانی اتفاق می افتد یک عنصر زیر مجموعه ( child object ) فوکوس برنامه را از دست می دهد .
<a href="#">hover ( )</a>	این رویداد دو تابع را در هنگام حرکت موس بر روی عنصر مورد نظر ، اجرا می کند .
<a href="#">mousedown ( )</a>	این رویداد در هنگام فشردن دکمه موس بر روی یک عنصر اجرا می شود .



## رویدادها در jQuery - رویداد click

رویداد click ( ) jQuery :

رویداد click زمانی اتفاق می افتد که کاربر بر روی عنصر مورد نظر کلیک نماید . با این رویداد برای مثال می توان کدی طراحی کرد که در هنگام کلیک کاربر بر روی یک عنصر رنگ پس زمینه آن عوض شده و یا عنصر مخفی شود . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).click ( function ( ) { ... some code ... } ;</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector کنترلی است که کاربر بر روی آن کلیک می کند .	selector
این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .	function ( )

Example	<code>\$( " :p " ).click ( function ( ) { ... some code } ;</code>
---------	--

در چند مثال سعی می کنیم تا استفاده از رویداد click را در عمل به شما نمایش دهیم .  
: در مثال اول یک پاراگراف داریم . کدی طراحی کرده ایم که در صورتی که کاربر بروی پاراگراف مورد نظر کلیک نماید ، رنگ پس زمینه آن تغییر می کند . برای مشاهده خروجی بر روی پاراگراف کلیک نمایید:

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document). ready ( function( ) {       \$( "# P_1 " ) . click ( function( ){         \$( "# P_1 " ) . css( " background-color " , " Yellow " );       }     )   } &lt; /script &gt;</pre>	کد

<pre> }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;p id="P_1" style="font-size: medium"&gt; jQuery is a very intelligent way for programming . &lt;/p &gt; &lt;/body&gt; &lt;/html&gt; </pre>	
jQuery is a very intelligent way for programming .	خروجی

: در مثال دوم یک تگ div طراحی کرده ایم که در صورت کلیک کاربر بر روی آن ، با استفاده از متد ( ) hide مخفی می شود . برای مشاهده خروجی روی عبارت خروجی مثال کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt;/script &gt;   &lt; script type="text/javascript" &gt;     \$(document). ready ( function( ) {       \$( "# div_1 " ) . click ( function( ){         \$( "# div_1 " ) . hide ( );       });     });   &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt; div id="div1" style="font-size: medium" border-width: thin; background-color: Lime &gt;   In jQuery you can do a lot of things with hmtl elements . &lt;br /&gt; even hide and show them .   &lt;/div &gt; </pre>	کد

<pre>&lt;/body&gt; &lt;/html&gt;</pre>	
<p>In jQuery you can do a lot of things with hmtl elements . even hide and show them .</p>	<p>خروجی</p>

## رویدادها در jQuery - dbclick رویداد

رویداد dbclick ( ) jQuery :

رویداد dbclick زمانی اتفاق می افتد که کاربر بر روی عنصر مورد نظر دابل کلیک نماید . با این رویداد برای مثال می توان کدی طراحی کرد که در هنگام دابل کلیک کاربر بر روی یک عنصر رنگ پس زمینه آن عوض شده و یا عنصر مخفی شود . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).dbclick ( function ( ) { ... some code ... } );</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector	کنترلی است که کاربر بر روی آن دابل کلیک می کند .
function ( )	این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .

Example	<code>\$( " :p " ).dbclick ( function ( ) { ... some code } );</code>
---------	---

در چند مثال سعی می کنیم تا استفاده از رویداد dbclick را در عمل به شما نمایش دهیم .

: در مثال اول یک پاراگراف داریم . کدی طراحی کرده ایم که در صورتی که کاربر بروی پاراگراف مورد نظر دابل کلیک نماید ، فونت نوشته به صورت توپر ( bold ) در می آید . برای مشاهده خروجی بر روی پاراگراف دابل کلیک نمایید:

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt; \$( document ). ready ( function ( ) {     \$( "# P_1 " ) . dbclick ( function ( ){         \$( "# P_1 " ) . css( " font-weight " , " bold " );     } ); } );</pre>	کد

<pre> }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;p id="P_1" style="font-size: medium"&gt; jQuery offers developers a lot of tools to work with html elements . &lt;/p &gt; &lt;/body&gt; &lt;/html&gt; </pre>	
jQuery offers developers a lot of tools to work with html elements .	خروجی

در مثال دوم یک تگ div طراحی کرده ایم که در صورت دابل کلیک کاربر بر روی آن ، با استفاده از متد ( ) slideUp به آرامی مخفی می شود . برای مشاهده خروجی روی عبارت خروجی مثال دابل کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document). ready ( function ( ) {       \$ ( "# div_1 " ) . dblclick ( function ( ){         \$ ( "# div_1 " ) . slideUp ( );       });     });   &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt; div id="div1" style="font-size: medium" border-width: thin; background-color: Lime &gt;   In jQuery you can do a lot of things with hmtl elements . &lt;br /&gt; even hide and show them .   &lt; /div &gt; </pre>	کد

<pre>&lt;/body&gt; &lt;/html&gt;</pre>	
<p>In jQuery you can do a lot of things with hmtl elements . even hide and show them .</p>	<p>خروجی</p>

## رویدادها در jQuery - رویداد blur

رویداد blur ( ) jQuery :

رویداد blur jQuery زمانی اتفاق می افتد که کنترل یا عنصر مورد نظر فوکوس \* برنامه را از دست می دهد . می توانید کدها و توابع مورد نظر خود را طوری طراحی کنید که در هنگام رخ دادن این رویداد ، اجرا شوند .

\* توضیح فوکوس ( focus ) : در هر لحظه کنترلی که در یک صفحه یا فرم ، انتخاب شده است و به دور آن کادر قرار گرفته به اصطلاح فوکوس برنامه را در اختیار دارد . کنترلی که انتخاب شود ، فوکوس را به دست آورده و کنترلی که از انتخاب خارج شود ، فوکوس را از دست می دهد .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).focus ( function ( ) { ... some code ... } ) ;</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector عنصر مورد نظری است که فوکوس برنامه را از دست می دهد . مثل یک کادر ورود متن که پس از پرکردن آن توسط کاربر و کلیک بر روی چیزی دیگر ، فوکوس را از دست می دهد .	selector
این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .	function ( )

Example	<code>\$( ":input " ).focus ( function ( ) { ... some code } ) ;</code>
---------	---

در مثال زیر روش استفاده از رویداد blur را نمایش داده ایم . در این مثال یک کادر متن داریم که کاربر باید نام خود را د کند . پس از اینکه نام خود را در کادر وارد کرده و بر روی چیز دیگری کلیک نمایید ، کنترل کادر متن فوکوس را از دست داده و تابعی اجرا می شود که رنگ پس زمینه آن را به خاکستری تبدیل می کند :

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;</pre>	کد

<pre>&lt; script type="text/javascript" &gt;   \$ (document). ready ( function( ) {     \$ ( "# Txt1 " ) . blur ( function( ){       \$ ( " # Txt1 " ) . css( " background-color " , " Gray " );     });   }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   Enter Your Name : &lt; input id="Txt1" type="text" /&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
Enter Your Name : <input data-bbox="443 898 572 949" type="text"/>	خروجی



## رویدادها در jQuery - رویداد change

رویداد change ( ) jQuery :

رویداد change زمانی اتفاق می افتد که در کنترل یا عنصر مورد نظر تغییری ایجاد می شود . این رویداد معمولاً برای کنترل هایی مثل کادر متن ، گزینه های انتخاب و یا منوهای کرکره ایی که محتویات یا حالات آنها قابل تغییر هستند ، به کار می رود .

زمانی که شما متن یک کنترل متن را تغییر داده و یا گزینه یک کادر انتخابی را عوض می کنید ، رویداد change اتفاق می افتد . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).change ( function ( ) { ... some code ... } );</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector کنترلی است که کاربر آن را تغییر می دهد .	selector
این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .	function ( )

Example	<code>\$( " :input " ).change ( function ( ) { ... some code } );</code>
---------	--

در چند مثال سعی می کنیم تا کاربرد این رویداد را در عمل نمایش دهیم .

در مثال اول یک کنترل کادر متن قرار داده ایم که کاربر باید نام خود را در آن وارد کند . در صورت وارد کردن متنی

در این کادر و تغییر محتویات آن ، رویداد change کنترل رخ داده و کدی را اجرا می کند . این کد رنگ پس زمینه کنترل را

به خاکستری تبدیل می کند :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;</pre>	کد

<pre>\$ (document). ready ( function( ) {   \$ (" # Txt1 ") . change ( function( ){     \$ ( " # Txt1 " ) . css( " background-color " , " Gray " );   }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   Enter Your Name : &lt; input id="Txt1" type="text" /&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<input type="text"/>	خروجی

## رویدادها در jQuery - رویداد focus

### رویداد focus ( ) jQuery :

رویداد focus زمانی اتفاق می افتد که کنترل یا عنصر مورد نظر فوکوس برنامه را به دست آورد . یک کنترل یا عنصر زمانی که انتخاب شده و یا به واسطه فشردن متوالی کلید Tab ، مرورگر بر روی آن متمرکز می شود ، فوکوس برنامه را در اختیار می گیرد . می توان کدهایی طراحی کرد که مثلا رنگ پس زمینه یک کادر دریافت متن در هنگام در اختیار گرفتن فوکوس تغییر نماید .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).focus ( function ( ) { ... some code ... } ) ;</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector کنترل یا عنصری است که فوکوس برنامه را به دست می آورد .	selector
این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .	function ( )

Example	<code>\$( " :input " ).focuss ( function ( ) { ... some code } ) ;</code>
---------	---

در چند مثال سعی می کنیم تا استفاده از رویداد focus را در عمل به شما نمایش دهیم .

در مثال دو کادر دریافت متن داریم . کدی طراحی کرده ایم ، که در هنگام انتخاب هر کدام از کادرهای متن و انتقال فوکوس برنامه به آن رنگ پس زمینه آن عوض می شود . برای مشاهده خروجی کادر های متن را انتخاب کرده و یا با زدن پشت سر هم کلید Tab ، فوکوس برنامه را به آنها انتقال دهید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;</pre>	کد

<pre>\$ (document). ready ( function( ) {   \$ (" :input ") . click ( function( ){     \$ ( this ) . css( " background-color " , " Gray " );   }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt; input type="text" id="Text1" size="10" /&gt;   &lt; input type="text" id="Text2" size="10" /&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<input data-bbox="188 902 322 952" type="text"/> <input data-bbox="188 958 322 1008" type="text"/>	خروجی

## رویدادها در jQuery - رویداد focusin

رویداد focusin ( ) jQuery :

رویداد focusin زمانی اتفاق می افتد که یک عنصر فرزند ( عنصری که در درون یک عنصر دیگر قرار گرفته است ) فوکوس برنامه را به دست آورد . عنصری که در هر لحظه انتخاب شده و تمرکز برنامه بر روی آن است ، فوکوس برنامه را در اختیار دارد . دقت کنید که برای رخ دادن این رویداد ، باید یک عنصر فرزند فوکوس را به دست آورد و نه خود عنصر . بدیهی است این رویداد در مورد تگ هایی که عنصر فرزند ندارند ، اجرا نمی شود . برای مثال فرض کنید که یک تگ < div > طراحی کرده ایم که در درون آن دو کادر متن قرار دارد . زمانی که هر کدام از کادرهای متنی درون آن ، فوکوس برنامه را به دست آورد ، این رویداد اتفاق می افتد . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).focusin ( function ( ) { ... some code ... } ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
selector	کنترل یا عنصری است که یک عنصر فرزند آن فوکوس برنامه را به دست می آورد .
function ( )	این تابع کد دستوری است که می خواهیم در هنگام رخ دادن رویداد ، انجام شود .

Example	<code>\$( "#div_1 " ).focusin ( function ( ) { ... some code } ;</code>
---------	---

در چند مثال سعی می کنیم تا استفاده از رویداد focusin را در عمل به شما نمایش دهیم .  
در مثال اول یک تگ < div > داریم که دو عدد کادر متن به عنوان عنصر فرزند در درون آن قرار گرفته است . کدی طراحی کرده ایم که با اجرای آن ، در هنگامی که یکی از کادرهای متن کنترل فوکوس برنامه را به دست آورد ، رنگ پس زمینه کنترل مادر یعنی تگ < div > تغییر می کند . برای مشاهده خروجی هر کدام از کادر های متن را انتخاب کنید :

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;</pre>	کد

<pre> &lt; script type="text/javascript" &gt;   \$ (document). ready ( function( ) {     \$ (" #div_1 ") . focusin ( function( ){       \$ ( this ) . css( " background-color " , " Gray " );     });   }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;div id="div_1" style="border: 1px solid black ; padding : 10px" &gt;     Name : &lt; input type="text" id="Text1" size="10" /&gt;     Family : &lt; input type="text" id="Text2" size="10" /&gt;   &lt; div &gt; &lt;/body&gt; &lt;/html&gt; </pre>	
Name : <input type="text"/> Family : <input type="text"/>	خروجی

در مثال دوم همان مثال اول را تکرار کرده ایم با این تفاوت که در این مورد ، با بدست آوردن فوکوس برنامه توسط هر یک از کادر های متن ، فونت نوشته عنصر مادر یعنی تگ div به صورت bold در می آید .

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt;/script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function( ) {       \$ (" #div_2 ") . focusin ( function( ){         \$ ( this ) . css( " font-weight " , " bold " );       });     });   &lt;/script&gt; </pre>	کد

<pre>&lt;/head&gt; &lt;body&gt;   &lt;div id="div_2" style="border: 1px solid black ; padding : 10px" &gt;     Name : &lt; input type="text" id="Text3" size="10" /&gt;     Family : &lt; input type="text" id="Text4" size="10" /&gt;   &lt; div &gt; &lt;/body&gt; &lt;/html&gt;</pre>	
Name : <input type="text"/> Family : <input type="text"/>	خروجی

## رویدادها در jQuery - hover رویداد

رویداد hover ( ) jQuery :

رویداد ( ) hover ، دو تابع را تعیین می کند که در هنگام عبور موس از روی عنصر مورد نظر ، اجرا می شوند . در هنگام ورود موس بر روی عنصر و تابع دوم در هنگام خروج عنصر از روی عنصر اجرا می شود . این رویداد هر دو رویداد [mouseenter](#) [mouseleave](#) را فعال می کند . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).hover ( inFunction , outFunction ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
selector	، تعیین کننده id عنصری است که می خواهیم رویداد ( ) hover در هنگام عبور از روی آن اتفاق بیفتد .
inFunction	این تابع زمانی اتفاق می افتد که رویداد <a href="#">mouseenter</a> رخ می دهد . یعنی زمانی که موس بر روی عنصر مورد نظر وارد می شود .
outFunction	این تابع زمانی اتفاق می افتد که رویداد <a href="#">mouseleave</a> رخ می دهد . یعنی زمانی که موس از روی عنصر مورد نظر خارج می شود .

**نکته مهم :** اگر فقط یک تابع در دستور رویداد ( ) hover تعیین شود ، این تابع برای هر رویداد اجرا می شود .

در چند مثال سعی می کنیم تا استفاده از رویداد ( ) hover را در عمل به شما نمایش دهیم .

: در مثال زیر یک پاراگراف داریم که برای رویداد ( ) hover آن دو تابع را تعریف کرده ایم ، که در هنگام

خارج شدن موس از روی پاراگراف مثال ، رنگ پس زمینه آن را تغییر می دهد :

Example	
< html >	کد



<pre>&lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function(){       \$("p").hover(function(){         \$("p").css("background-color","yellow");       },function(){         \$("p").css("background-color","pink");       });     });   &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt; p id="p1" &gt; ! موس را بر روی پاراگراف حرکت دهید ! &lt;/body&gt; &lt;/html&gt;</pre>	
موس را بر روی پاراگراف حرکت دهید !	خروجی

## رویدادها در jQuery - mousedown رویداد

### رویداد mousedown ( ) jQuery :

رویداد mousedown زمانی اتفاق می افتد که دکمه سمت چپ موس را بر روی عنصر مورد نظر فشار دهید .

رویداد mousedown می تواند دو کار را انجام دهد :

باعث رخ دادن رویداد mousedown برای عنصر مورد نظر می شود . شکل استفاده از این حالت به صورت زیر

:

Syntax	<code>\$( " selector " ).mousedown ( ) ;</code>
--------	---

: در مثال زیر عبور موس از روی تگ div ، باعث رخ دادن رویداد mousedown کنترل button می شود

. همانند این است که دکمه موس را بر روی دکمه فرمان button فشار دهید . برای مشاهده عملکرد مثال ، موس را

بر روی تگ div ببرید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function(){       \$("button").mousedown(function( ){ \$("p").slideToggle( ); });       \$("div").mouseover(function( ){ \$("button").mousedown( ); });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; p &gt; This is a paragraph. &lt; /p &gt;   &lt; button &gt; Toggle &lt; /button &gt; &lt; br &gt; &lt; br &gt;   &lt; div &gt; mouse over this text to trigger the mousedown event for the button   above. &lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
<p>This is a paragraph.</p> <p>Toggle</p> <p>Mouse over this text to trigger the mousedown event for the button above.</p>	خروجی

. یا باعث اجرا شدن تابع یا دستوری در هنگام فشردن دکمه موس بر روی عنصر مورد نظر می شود . شکل

استفاده از این حالت به صورت زیر است:

Syntax	<code>\$( " selector " ).mousedown ( functoin ) ;</code>
--------	--

. : در مثال زیر فشار دادن دکمه موس بر روی تگ `div` در خروجی مثال ، باعث اضافه شدن یک پاراگراف به انتهای مثال می شود:

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function(){       \$("div").mousedown(function(){         \$(this).after("&lt;p&gt;Mouse button pressed down.&lt;/p&gt;");       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; div &gt; Press down the mouse button over this div element.&lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کا
Mouse over this text to trigger the mousedown event for the button above.	خروجی

## رویدادها در jQuery - رویداد mouseup

### رویداد mouseup ( ) jQuery :

رویداد mouseup زمانی اتفاق می افتد که دکمه فشرده شده موس را بر روی عنصر مورد نظر ، رها دهید .

رویداد mouseup می تواند دو کار را انجام دهد :

باعث رخ دادن رویداد mouseup برای عنصر مورد نظر می شود . شکل استفاده از این حالت به صورت زیر

:

Syntax	<code>\$( " selector " ).mouseup ( ) ;</code>
--------	---

: در مثال زیر عبور موس از روی تگ div ، باعث رخ دادن رویداد mouseup کنترل button می شود .  
همانند این است که دکمه موس را بر روی دکمه فرمان button فشار داده و رها نمایید . برای مشاهده عملکرد مثال ،  
موس را بر روی تگ div ببرید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function(){       \$("button").mouseup(function( ){ \$("p").slideToggle( ); });       \$("div").mouseover(function( ){ \$("button").mouseup( ); });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; p &gt; This is a paragraph. &lt; /p &gt;   &lt; button &gt; Toggle &lt; /button &gt; &lt; br &gt; &lt; br &gt;   &lt; div &gt; mouse over this text to trigger the mouseup event for the button   above. &lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
<p>This is a paragraph.</p> <p>Toggle</p> <p>Mouse over this text to trigger the mouseup event for the button above.</p>	خروجی

یا باعث اجرا شدن تابع یا دستوری در هنگام فشرده شدن دکمه موس بر روی عنصر مورد نظر و پس از رها کردن

آن می شود . شکل استفاده از این حالت به صورت زیر :

Syntax	<code>\$( " selector " ).mouseup ( functoin ) ;</code>
--------	--

. در مثال زیر فشار دادن دکمه موس و سپس رها کردن آن بر روی تگ `div` در خروجی مثال ، باعث اضافه شدن یک پاراگراف به انتهای مثال می شود:

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function(){       \$("div").mouseup(function(){         \$(this).after("&lt;p&gt;Mouse button pressed down.&lt;/p&gt;");       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; div &gt; Press down the mouse button over this div element.&lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
Mouse over this text to trigger the mouseup event for the button above.	خروجی

## رویدادها در jQuery - رویداد ( ) ready

### رویداد ( ) ready : jQuery

رویداد ( ) ready در یک صفحه وب زمانی اتفاق می افتد که شی [DOM](#) به طور کامل لود شده و همچنین صفحه نیز به طور کامل لود شده باشد (حتی تصاویر آن).

به دلیل اینکه رویداد ( ) ready پس از لود شدن کامل صفحه اتفاق می افتد ، مکان مناسبی برای قرار دادن کدها و توابع jQuery .

شما می توانید رویدادها و توابعی که می خواهید به محض تمام شدن عملیات load صفحه ، اجرا شوند را در این تابع قرار دهید .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<pre>\$ ( document ).ready ( function ) ; یا می توان بدون به کار بردن نام سند بر روی صفحه جاری نیز ، تابع را اجرا نمود \$ ( function ) ;</pre>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده صفحه یا سندی است که می خواهیم تابع در آن اجرا شود . document به صفحه جاری اشاره دارد .	document
نام تابعی است که می خواهیم به محض لود شدن صفحه .	function

: در مثال های قبلی آموزش jQuery ، این رویداد را همواره به کار برده ایم . به این مثال نیز دقت نمایید . در رویداد تابعی را تعیین کرده ایم که به محض لود شدن صفحه ، اجرا خواهد شد :

## Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $(document).ready(function () {
      $("button").click(function () {
        $("#p1").slideToggle();
      });
    });
  < /script >
< /head >
< body >
  < p id="p1" > This is a paragraph.< /p >
  < button > برای اجرا تابع کلیک کنید < /button >
< /body >
< /html >
```

کد

This is a paragraph.

برای اجرا تابع کلیک کنید

خروجی

## رویدادها در jQuery - رویداد ( ) resize

### رویداد ( ) resize : jQuery

رویداد ( ) resize زمانی اتفاق می افتد که اندازه پنجره مرورگر تغییر نماید .  
رویداد ( ) resize هم می تواند باعث تغییر اندازه پنجره مرورگر شده و یا اینکه در هنگام تغییر اندازه آن ، تابع تعیین شده را اجرا نماید .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).resize ( ) ;</code> یا <code>\$( " selector " ).resize ( function ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده نام پنجره ای است که می خواهید در هنگام تغییر اندازه آن ، رویداد انجام شود . معمولاً مقدار این پارامتر window بوده ، که به پنجره جاری مرورگر اشاره می کند .	selector
این پارامتر تابعی را تعیین می کند که می خواهیم در هنگام تغییر اندازه پنجره مرورگر اجرا شود .	function

: در مثال زیر تابعی را نوشته ایم که در هر بار تغییر اندازه پنجره مرورگر ، کدی را اجرا کرده و مقدار یک متغیر را افزایش می دهد . برای مشاهده خروجی مثال در عمل ، پنجره مرورگر خود را کوچک و بزرگ نمایید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     x = 0;     \$(document).ready(function () {       \$(window).resize(function () {         \$("span").text(x += 1);       });     });   &lt; /script &gt; &lt; /html &gt;</pre>	کد



<pre>}); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;p&gt; &lt;span id="Span1"&gt; 0 &lt;/span&gt; تغییر اندازه مرورگر &lt;/p&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
تغییر اندازه مرورگر 0	خروجی

## رویدادها در jQuery - رویداد ( scroll )

### رویداد scroll ( ) در jQuery :

رویداد ( scroll ) زمانی اتفاق می افتد که کاربر در یک عنصر صفحه ، عمل scroll ( بالا و پایین رفتن ) را انجام دهد . این رویداد برای تمام عناصری که دارای خاصیت scroll هستند ، م ( window , textarea ... ) . رویداد هم می تواند باعث scroll یک عنصر شده و یا اینکه در هنگام scroll آن عنصر ، یک تابع تعیین شده را اجرا نماید . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).scroll ( ) ;</code> یا <code>\$( " selector " ).scroll ( function ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر ، تعیین کننده id عنصری است که عمل scroll بر روی آن انجام می شود .	selector
این پارامتر تابعی را تعیین می کند که می خواهیم در هنگام scroll	function

: در مثال زیر یک عنصر div scrollbar قرار داده ایم . هر بار که شما عمل scroll را بر روی آن انجام دهید ، تابعی اجرا شده و مقدار یک متغیر را افزایش می دهد . برای مشاهده خروجی مثال ، تگ div scroll کنید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     x = 0;     \$(document).ready(function () {       \$('#div1').scroll(function () {         \$("#span1").text(x += 1);       });     });   &lt; /script &gt; &lt; /html &gt;</pre>	کد

<pre> });  }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;div style="border: 1px solid black; width: 200px; height: 100px; overflow: scroll;" id="div1"&gt;     scroll ( زمانی اتفاق می افتد که کاربر در یک عنصر صفحه ، عمل scroll ( ) رویداد     . و پایین رفتن ) را انجام دهد     , window ( هستند ، مثل صفحه scroll این رویداد برای تمام عناصر که دارای خاصیت     textarea ( ... .     scroll یک عنصر شده و یا اینکه در هنگام scroll رویداد هم می ت     : ، یک تابع تعیین شده را اجرا نماید . شکل کلی استفاده از این متد به شرح زیر است     &lt;/div&gt;     &lt;p&gt; Scrolled &lt; span id="span1" &gt; 0 &lt;/span &gt; times. &lt; /p &gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>بالا و پایین رفتن ) ( scroll زمانی اتفاق می افتد که کاربر در یک عنصر صفحه ، عمل scroll ( ) رویداد انجام دهد .</p> <p>( ... window , textarea ) هستند ، مثل صفحه scroll این رویداد برای تمام عناصر که دارای خاصیت</p> <p>آن عنصر ، یک تابع تعیین شده scroll یک عنصر شده و یا اینکه در هنگام scroll رویداد هم می تواند باعث</p> <p>: را اجرا نماید . شکل کلی استفاده از این متد به شرح زیر است</p> <p>Scrolled 0 times.</p>	<p>خروجی</p>

## رویدادها در jQuery - رویداد ( ) select

رویداد ( ) select : jQuery

رویداد ( ) select زمانی اتفاق می افتد که متن موجود در یک کادر متن ( textfield ) یا textarea . این رویداد دو کار را انجام می دهد :

باعث بروز و رخ دادن رویداد select ( ) برای یک کادر متن می شود . یعنی همانند این است که متن آن را انتخاب نماییم :

Syntax	<code>\$ ( " selector " ).select ( ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم متن آن را انتخاب نماییم .	selector

در مثال زیر یک دکمه فرمان قرار داده ایم . با کلیک بر روی آن تابعی اجرا می شود ، که باعث رخ دادن رویداد ( ) select کنترل کادر متن می شود . رخ دادن این رویداد هم باعث اجرای تابع اول می شود :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function () {       \$("#input1").select(function(){         \$("#input1").after(" Text marked!");       });       \$("#btn1").click(function(){         \$("#input1").select();;       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt; &lt; input id="input1" type="text" value="Hello World"&gt; &lt; p id="p1"&gt;متنی را دورن کادر متن انتخاب کنید &lt; /p &gt; &lt; button id="Btn1"&gt; کلیک نمایید &lt; /button &gt; &lt; /body &gt;</pre>	کد

</html>	
<input type="text" value="Hello World"/> متنی را دورن کادر متن انتخاب کنید کلیک نمایید	خروجی

. در صورت انتخاب متن یک کادر متن ، این رویداد رخ می دهد . می توان تابعی را تعیین نمود تا در هنگام رخ دادن این رویداد اجرا شود .  
شکل کلی استفاده از این رویداد به صورت زیر است:

Syntax	<code>\$( " selector " ).select ( function ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم متن آن را انتخاب نماییم و یا متنش انتخاب می شود .	selector

. در مثال زیر یک کادر متن قرار داده ایم . در صورتی که متن آن را انتخاب نمایید ، رویداد ( ) select رخ داده و دستوری را اجرا می کند که یک متن را به صفحه اضافه خواهد کرد :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function () {       \$("#input2").select(function () {         \$("#input2").after(" Text marked!");       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt; &lt; input id="input2" type="text" value="Hello World" &gt; &lt; p id="p2" &gt;متنی را دورن کادر متن انتخاب کنید &lt; /p &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
<input type="text" value="Hello World"/> متنی را دورن کادر متن انتخاب کنید	خروجی

## رویدادها در jQuery - رویداد ( ) submit

رویداد submit ( ) jQuery :

رویداد ( ) submit زمانی اتفاق می افتد که یک فرم submit شده و به سرور ارسال شود .  
برای مثال از رویداد ( ) submit می توان برای صدور یک پیام هشدار به کاربر ، برای اعلام submit شدن فرم استفاده کرد .

**نکته مهم :** رویداد ( ) submit form کار می کند .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).submit ( ) ;</code> یا <code>\$( " selector " ).submit ( function ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id فرمی است که submit	selector
تعیین کننده تابعی است که می خواهیم پس از submit	function

: در مثال زیر یک فرم طراحی کرده ایم . هنگامی که کاربر با کلیک بر روی دکمه فرمان ، فرم را submit می کند ، یک پیام هشدار مبنی بر ثبت شدن فرم به وی نشان داده شده و سپس فرم به سرور ارسال می شود :

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt;     \$(document).ready(function () {         \$("#form1").submit(function(){             alert("      ");         });     });</pre>	کد

<pre>}); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;form action="" id="form1"&gt;     First name:     &lt;input type="text" name="FirstName" value="Ali"&gt;&lt;br&gt;     Last name:     &lt;input type="text" name="LastName" value="Rezaee"&gt;&lt;br&gt;     &lt;input type="submit" value="Submit"&gt;   &lt;/form&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>First name: <input type="text" value="Ali"/></p> <p>Last name: <input type="text" value="Rezaee"/></p> <p><input type="submit" value="Submit"/></p>	<p>خروجی</p>

## رویدادها در jQuery - رویداد ( ) toggle

رویداد toggle ( ) jQuery :

افکت ( ) toggle بین دو افکت [show \( \)](#) و [hide \( \)](#) تغییر وضعیت می دهد .

افکت ( ) toggle وضعیت نمایش و یا عدم نمایش عنصر مورد نظر را بر روی صفحه می سنجد . اگر عنصر در حال نمایش Hide مخفی کرده و اگر عنصر مخفی باشد آن را با استفاده از متد Show نمایش می دهد .

**نکته :** این متد برای عناصری که از ابتدای اجرای صفحه مخفی تعیین شده اند ، کار نمی کند .

برای مشاهده نحوه کار افکت ( ) toggle به مثال عملی دقت نمایید .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( selector ).toggle( speed , easing , callback ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم افکت ( ) toggle ، بر روی آن اجرا شود .	selector
این پارامتر تعیین کننده سرعت انجام افکت و تغییر حالت عنصر از حالت اول به دوم است . این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از : ( MiliSecond : مدت زمان اجرای افکت بر حسب میلی ثانیه . ( Slow : سرعت آهسته ( Fast : سرعت زی	speed
این پارامتر تعیین کننده سرعت انجام افکت در ابتدا و انتهای آن است . استفاده از این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از : ( Swing : سرعت انجام افکت در ابتدا و انتهای فرایند آهسته ولی در وسط آن زیاد است . ( Liner : سرعت در کل طول فرایند یکسان است .	easing



<p>به وسیله این پارامتر می توانیم یک تابع را تعیین کنیم ، تا در پایان انجام افکت ( toggle ) استفاده از این پارامتر اختیاری است .</p>	<p>function</p>
--	-----------------

: در مثال زیر یک پاراگراف داریم . برای آن یک رویداد ( toggle ) نوشته ایم ، با هر بای کلیک کاربر بر روی دکمه فرمان رویداد ( toggle ) برای آن اجرا شده و وضعیت نمایش یا عدم نمایش آن تغییر می کند :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function () {       \$("#btn1").click(function(){         \$("#p1").toggle();       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; p id="p1" style="font-size: 16px"&gt; بزرگ و کوچک عنصر &lt; /p &gt;   &lt; button id="btn1" type="button"&gt; Toggle ( ) &lt; /button &gt; &lt; /body &gt; &lt; /html &gt; </pre>	<p>کد</p>
<p>نحوه تغییر وضعیت عنصر</p> <p>Toggle ( )</p>	<p>خروجی</p>

## رویدادها در jQuery - unload ( )

رویداد **unload ( )** jQuery :

رویداد ( ) unload زمانی اتفاق می افتد که کاربر از صفحه جاری خارج می شود . برای مثال یکی از استفاده های جالب این رویداد ، می تواند صدور یک پیام خداحافظی برای کاربر در زمانی که می خواهد صفحه را ببندد ، باشد .  
رویداد ( ) unload در مواقع زیر رخ می دهد :

- کلیک کاربر بر روی لینکی که باعث خروج از صفحه جاری و رفتن به صفحه دیگری می شود.
- یک آدرس جدید در نوار آدرس مرورگر نوشته شده و کلید Enter را بزنید.
- کلیدهای ( forward رفتن به جلو ) ( backward رفتن به عقب ) را در مرورگر کلیک نمایید.
- پنجره جاری مرورگر بسته شود.
- صفحه مجدد لود و فراخوانی شود.

رویداد ( ) unload ، می تواند کد یا تابعی را در زمانی که صفحه جاری بسته می شود ، اجرا نماید .  
شکل کلی استفاده از رویداد ( ) unload به صورت زیر است :

Syntax	<code>\$( selector ).unload( function ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده تابعی است که می خواهیم در هنگام رخ دادن رویداد ( ) unload .	function

**چند نکته :**

**نکته :** رویداد ( ) unload فقط بایستی با شی window به کار برود .

**نکته :** رویداد ( ) unload ممکن است در مرورگر های مختلف ، عملکرد متفاوتی داشته باشد . آن را در مرورگرهای تست نمایید .

**مثال عملی :**

در مثال زیر یک تابع را تعیین کرده ایم که در هنگام ، خروج از صفحه جاری یک پیام خداحافظی را به کاربر اعلام می نماید .  
برای مشاهده خروجی از صفحه خارج شوید !! :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      $(window).unload(function () {
        alert(" ! به امید دیدار ");
      });
    });
  < /script >
< /head >
< body >
< /body >
< /html >
```

ک

## لیست افکت ها ( effects ) jQuery

### لیست افکت ها ( effects ) jQuery :

افکت ها در jQuery یکی از مهمترین و پرکاربردترین قابلیت ها در این زبان هستند که به طراحان وب امکانات گسترده و جذابی را می دهند . افکت ها jQuery شامل تعدادی متد از پیش تعریف شده هستند که برای مثال به شما امکان می دهند عناصر مورد نظر خود را مخفی و نمایان کرده و یا نحوه نمایش آن را تغییر دهید . استفاده از افکت ها در jQuery بسیار ساده و شبیه رویداد ها در این زبان هستند . بهتر است قبل از مطالعه این بخش ، [آموزش رویدادها در jQuery](#) را مطالعه نمایید .

در لیست زیر به معرفی مهمترین افکت ها در jQuery پرداخته ایم . برای دریافت اطلاعات درباره هر افکت بر روی نام آن کلیک نمایید :

نام افکت	
<a href="#">fadeIn ( )</a>	این افکت باعث می شود عنصر مورد اثر خود را به آرامی از حالت نمایان به حالت مخفی در بیاورد .
<a href="#">fadeOut ( )</a>	این افکت باعث می شود عنصر مورد اثر خود را به آرامی از حالت مخفی به حالت نمایان در بیاورد .
<a href="#">fadeTo ( )</a>	این افکت میزان وضوح نمایش عنصر مورد اثر خود را به یک میزان دلخواه تغییر می دهد .
<a href="#">hide ( )</a>	این افکت عنصر مورد اثر خود را مخفی می کند .
<a href="#">show ( )</a>	این افکت عنصر مورد اثر خود را نمایان می کند .
<a href="#">slideDown ( )</a>	این افکت با افزایش تدریجی ارتفاع ، عنصر مورد اثر خود را نمایان می کند .
<a href="#">slideUp ( )</a>	این افکت با کاهش تدریجی ارتفاع ، عنصر مورد اثر خود را مخفی می کند .
<a href="#">slideToggle ( )</a>	این افکت وضعیت نمایش عنصر را برعکس می کند . اگر عنصر مخفی باشد آن را نمایان ساخته و یا اگر نمایان را مخفی می کند .

## افکت ها در jQuery - افکت ( ) animate

افکت ( ) animate : jQuery

این متد بوسیله تغییر یک یا چند خاصیت CSS عنصر مورد نظر از یک حالت اولیه به مقداری جدید یک عمل انیمیشن را انجام می دهد . مقدار خاصیت یا خاصیت های CSS تعیین شده برای تغییر در عنصر مورد نظر توسط یک افکت ( ) animate بصورت آهسته از مقدار اولیه به مقدار ثانویه تغییر می کنند که باعث ایجاد یک حالت انیمیشن می شود . فرض کنید یک عنصر div به رنگ پس زمینه سبز دارید . مقدار خاصیت height ( ) آن در حالت اولیه 100 px می خواهد بوسیله خاصیت ( ) animate مقدار ارتفاع آن را به 300 px افزایش دهید . در هنگام اجرای ( ) animate ارتفاع عنصر به صورت آهسته از به افزایش می یابد و باعث ایجاد یک افکت تصویری می شود .

**نکته مهم :** افکت ( ) animate CSS ای که مقادیر آنها عددی است مثل ( Height : 30 px ) کار می کند و نمی تواند آن را با خواصی که مقدار رشته ای دارند مثل ( background-color:red ) به کار ببرید .

شکل کلی استفاده از این افکت بصورت زیر است :

Syntax	( selector ).animate ( {styles} , speed , easing , callback ) ;
--------	---

وضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم ، متد ( ) animate را روی آن اجرا نماییم .	selector
این پارامتر یک یا چند خاصیت CSS عنصر مورد نظر را برای تغییر و ایجاد انیمیشن تعیین می کند . استفاده از این پارامتر اجباری است .	styles
<b>نکته :</b> نکته مهم : در هنگام به کار بردن خواص CSS ( ) animate بایستی کاراکترهایی مثل - خواص حذف کنید . برای مثال خاصیت Padding-Left بکار ببرید . انواع خواص قابل استفاده در این پارامتر : backgroundPositionX * backgroundPositionY * borderWidth *	

borderBottomWidth *	
borderLeftWidth *	
borderRightWidth *	
borderTopWidth *	
borderSpacing *	
margin *	
marginBottom *	
marginLeft *	
marginRight *	
marginTop *	
outlineWidth *	
padding *	
paddingBottom *	
paddingLeft *	
paddingRight *	
paddingTop *	
height *	
width *	
maxHeight *	
maxWidth *	
minHeight *	
minWidth *	
fontSize *	
bottom *	
left *	
right *	
top *	
letterSpacing *	
wordSpacing *	
lineHeight *	
textIndent *	
این پارامتر تعیین کننده سرعت اتمام انیمیشن و تغییر خاصیت عنصر از حالت اول به دوم است .	speed

<p>استفاده از این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از :</p> <p>. MiliSecond : مدت زمان اجرای انیمیشن بر حسب میلی ثانیه .</p> <p>. Slow : سرعت آهسته .</p> <p>. Fast : سرعت زیاد .</p>	
<p>این پارامتر تعیین کننده سرعت انجام انیمیشن در ابتدا و انتهای فرایند متد ( ) animate . استفاده از این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از :</p> <p>. Swing : سرعت انجام انیمیشن در ابتدا و انتهای فرایند آهسته ولی در وسط آن زیاد است .</p> <p>. Liner : سرعت در کل طول فرایند یکسان است .</p>	<p>easing</p>

: در مثال زیر ، یک تگ div داریم ، که با کلیک بر روی دکمه فرمان ، ارتفاع آن از 100 px به 300 px افزایش می یابد ، که باعث ایجاد یک انیمیشن تصویری می شود :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt;/script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$("#btn1").click(function () {         \$("#box").animate({ height: "300px" });       });       \$("#btn2").click(function () {         \$("#box").animate({ height: "100px" });       });     });   &lt;/script&gt; &lt;/head&gt;</pre>	<p>کد</p>

<pre>&lt;body&gt;   &lt;button id="btn1" &gt; افزایش ارتفاع &lt;/button&gt;   &lt;button id="btn1" &gt; کاهش ارتفاع &lt;/button&gt;   &lt;div id="Div1" style="background: #98bf21; height: 100px; width: 100px; margin: 6px;"&gt; &lt;/div&gt; &gt; &lt;/body&gt; &lt;/html&gt;</pre>	
کاهش ارتفاع افزایش ارتفاع	خروجی



## افکت ها - jQuery - افکت delay

### افکت delay ( ) : jQuery

این افکت باعث ایجاد یک تاخیر در اجرای افکت دیگری می شود .

برای مثال فرض کنید که یک افکت ( ) fadeIn را برای یک عنصر تعیین کرده اید . این افکت در هنگام کلیک کاربر بر روی دکمه فرمان اجرا می شود . اما می خواهید کاری کنید تا مثلا ثانیه بعد از کلیک کاربر بر روی دکمه فرمان ، افکت ( ) fadeIn . برای این منظور بایستی از افکت ( ) delay به همراه آن استفاده کنید .  
شکل کلی استفاده از این افکت بصورت زیر است :

Syntax	\$( selector ).delay( speed ) ;
--------	---------------------------------

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم در اجرای یک افکت بر روی آن ، تاخیر ایجاد نماییم .	selector
این پارامتر تعیین کننده مدت زمان تاخیر در اجرای افکت است . انواع مقادیر ممکن عبارتند از : MiliSecond : مدت زمان تاخیر در اجرای افکت بر حسب میلی ثانیه . Slow : سرعت آهسته . Fast : سرعت زیاد .	speed

: در مثال زیر چندین تگ div داریم ، که در هنگام کلیک بر روی دکمه فرمان مثال ، بر روی صفحه ظاهر می شوند .

اما برای اجرای هر کدام یک تاخیر با استفاده ( ) delay ایجاد کرده ایم :

## Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      $("button").click(function () {
        $("#div1").delay("slow").fadeIn();
        $("#div2").delay("fast").fadeIn();
        $("#div3").delay( ( ) delay 00).fadeIn();
        $("#div4").delay(2000).fadeIn();
        $("#div5").delay(4000).fadeIn();
      });
    });
  < /script >
< /head >
< body >
  < button type="button" > برای مشاهده باکس ها با تاخیر کلیک نمایید < /button >
  < br >
  < div id="div1" style="width: 90px; height: 90px; display: none;
background-color: black;" > < /div >
  < br >
  < div id="div2" style="width: 90px; height: 90px; display: none;
background-color: green;" > < /div >
  < br >
  < div id="div3" style="width: 90px; height: 90px; display: none;
background-color: blue;" > < /div >
  < br >
  < div id="div4" style="width: 90px; height: 90px; display: none;
background-color: red;" > < /div >
  < br >
  < div id="div5" style="width: 90px; height: 90px; display: none;
```

کد

<pre>background-color: purple;"&gt; &lt;/div&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>برای مشاهده باکس ها با تاخیر کلیک نمایید</p>	<p>خروجی</p>

## افکت ها در jQuery - افکت fadeIn

افکت fadeIn ( ) : jQuery

این افکت باعث می شود تا عنصر مورد اثر آن به تدریج از حالت مخفی به حالت نمایان در بیاید. این افکت در یک افکت تصویری با زیاد کردن نور عنصر به تدریج باعث نمایان شدن آن می شود . بدیهی است این افکت بر روی عناصری قابل اجراست که در ابتدا مخفی باشند . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).fadeIn ( speed , callback )</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector	عنصری است که این افکت باعث نمایان شدن آن می شود .
speed	این پارامتر اختیاری بوده و مقدار پیش فرض آن صفر است . این پارامتر سرعت نمایان شدن عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقاردهی می با . <ul style="list-style-type: none"><li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر به تدریج نمایان شود را تعیین می کنید . ms</li><li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت نمایان شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li>"slow" ○</li><li>"normal" ○</li><li>"fast" ○</li></ul></li></ul>
callback	توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و نمایان شدن عنصر . به کار بردن این پارامتر اختیاری است .

در چند مثال سعی می کنیم تا استفاده از افکت ( ) fadeIn را در عمل به شما نمایش دهیم .

: در مثال اول یک تگ < div > داریم که در ابتدا از دید کاربر مخفی است . یک دکمه فرمان طراحی کرده ایم که کاربر برای مشاهده تگ < div > مخفی باید بر روی آن کلیک نماید . با کلیک بر روی دکمه فرما می شود :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ( "# Btn_1 " ) . click ( function ( ){         \$ ( "# div_1 " ) . fadeIn ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_1" value="click me for fadeIn!" /&gt;   &lt; div id="div_1" style="font-size: medium; border: solid 1px black" &gt;     در این سایت آموزش زبان های زیر را داریم     &lt; ul &gt;       &lt; li &gt; HTML &lt; /li &gt;       &lt; li &gt; CSS &lt; /li &gt;       &lt; li &gt; Java Script &lt; /li &gt;       &lt; li &gt; jQuery &lt; /li &gt;     &lt; /ul &gt;   &lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
	خروجی

speed در افکت fadeIn در مثال دوم ، مثال قبل را بازنویسی کرده ایم با این تفاوت که در :

حالت به پارامتر speed افکت fadeIn مقدار داده ایم . برای مشاهده تاثیر مقدار پارامتر بر روی افکت بر روی خروجی

های مثال کلیک نمایید :

slow برای انجام افکت :	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ( "# Btn_3 " ) . click ( function ( ) {         \$ ( "# div_3 " ) . fadeIn ( "slow" )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_3" value="click me for fadeIn!" /&gt;   &lt; div id= "div_3" style = "font-size: medium ; border: solid 1px black "   &gt;   در این سایت آموزش زبان های زیر را داریم   &lt; ul &gt;     &lt; li &gt; HTML &lt; /li &gt;     &lt; li &gt; CSS &lt; /li &gt;     &lt; li &gt; Java Script &lt; /li &gt;     &lt; li &gt; jQuery &lt; /li &gt;   &lt; /ul &gt;   &lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
	خروجی

میلی ثانیه برای انجام افکت :

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      $ ( " # Btn_2 " ) . click ( function ( ){
        $ ( " # div_2 " ) . fadeIn( 2000 )
      });
    });
  < /script >
< /head >
< body >
  < input type="button" id="Btn_2" value="click me for fadeIn!" />
  < div id= "div_2" style = "font-size: medium ; border: solid 1px black "
  >
  در این سایت آموزش زبان های زیر را داریم
  < ul >
    < li > HTML < /li >
    < li > CSS < /li >
    < li > Java Script < /li >
    < li > jQuery < /li >
  < /ul >
  < /div >
< /body >
< /html >
```

کد

خروجی

## افکت ها در jQuery - افکت fadeOut

افکت fadeOut ( ) : jQuery

این افکت باعث می شود تا عنصر مورد اثر آن به تدریج از حالت نمایان به حالت مخفی در بیاید. این افکت در یک افکت تصویری با کم کردن نور عنصر به تدریج باعث مخفی شدن آن از دید کاربر می شود. شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).fadeOut ( speed , callback )</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector عنصری است که این افکت باعث مخفی شدن آن می شود .	selector
<p>این پارامتر اختیاری بوده و مقدار پیش فرض آن صفر است .</p> <p>این پارامتر سرعت مخفی شدن عنصر توسط افکت را تعیین می کند .</p> <p>این پارامتر به دو صورت قابل مقاردهی می باشد .</p> <ul style="list-style-type: none"><li>• تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر به تدریج مخفی شود را تعیین می کنید . ms</li><li>• تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت مخفی شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .</li></ul> <ul style="list-style-type: none"><li>○ "slow"</li><li>○ "normal"</li><li>○ "fast"</li></ul>	speed
توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و مخفی شدن .	callback
به کار بردن این پارامتر اختیاری است .	

در چند مثال سعی می کنیم تا استفاده از افکت ( ) fadeOut را در عمل به شما نمایش دهیم .



: در مثال اول یک تگ < div > داریم که با کلیک کاربر بر روی آن افکت fadeOut اجرا شده و باعث مخفی شدن آن می شود . برای اجرای افکت بر روی متن تگ < div > کلیک نمایید .

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# div_1 " ) . click ( function ( ){         \$( "# div_1 " ) . fadeOut ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; div id= "div_1" style = "font-size: medium ; border: solid 1px black " &gt;     jQuery is a very inteligent way for programming . many wonderful things you can do with it !   &lt; /div &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
<p>jQuery is a very inteligent way for programming . many wonderful things you can do with it !</p>	خروجی

: **speed در افکت fadeOut** در مثال دوم ، مثال قبل را بازنویسی کرده ایم با این تفاوت که حالت به پارامتر speed افکت fadeOut مقدار داده ایم . برای مشاهده تاثیر مقدار پارامتر بر روی افکت بر روی خروجی های مثال کلیک نمایید :

: میلی ثانیه برای انجام افکت	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;</pre>	کد

<pre> &lt; script type="text/javascript" &gt;   \$ (document). ready ( function( ) {     \$ (" # div_2 ") . click ( function(){       \$ ( " # div_2 " ) . fadeOut( 2000 )     });   }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;div id= "div_2" style = "font-size: medium ; border: solid 1px black " &gt;     jQuery is a very inteligent way for programming . many wonderful things you     can do with it !   &lt; /div &gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>jQuery is a very inteligent way for programming . many wonderful things you can do with it !</p>	<p>خروجی</p>

<p>slow برای انجام افکت :</p>	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function( ) {       \$ (" # div_3 ") . click ( function(){         \$ ( " # div_3 " ) . fadeOut( "slow" )       });     });   &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;div id= "div_3" style = "font-size: medium ; border: solid 1px black " &gt;     jQuery is a very inteligent way for programming . many wonderful things you     can do with it !   &lt; /div &gt; &lt;/body&gt; &lt;/html&gt; </pre>	<p>کد</p>
<p>jQuery is a very inteligent way for programming . many wonderful things you can do with it !</p>	<p>خروجی</p>

## افکت ها در jQuery - افکت fadeTo

**افکت fadeTo () jQuery** : این متد میزان شفافیت و وضوح نمایش یک عنصر را به تدریج به یک مقدار تعیین شده از سوی کاربر تغییر می دهد .

به عبارت دیگر این افکت میزان نور نمایش یک عنصر را کم یا زیاد می کند . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).fadeTo ( speed , opacity , callback )</code>
--------	---

توضیح syntax	
توضیح	Parameter
selector	عنصری است که این افکت باعث تغییر وضوح نمایش آن می شود .
speed	این پارامتر در این متد اجباری بوده و حتما باید مقدار دهی شود . در غیر این صورت افکت کار نخواهد کرد . این پارامتر سرعت تغییر وضوح تصویر عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقداردهی می <ul style="list-style-type: none"> <li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن وضوح تصویر عنصر تغییر کند را تعیین می کنید . ms</li> <li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت تغییر وضوح تصویر عنصر را آرام ، معمولی و یا سریع انتخاب می نمایید . <ul style="list-style-type: none"> <li>"slow" ○</li> <li>"normal" ○</li> <li>"fast" ○</li> </ul> </li> </ul>
opacity	این پارامتر در این متد اجباری بوده و حتما باید مقدار دهی شود . در غیر این صورت افکت کار نخواهد کرد . این پارامتر میزان تغییر وضوح تصویر عنصر را تعیین می کند . مقداری که به این پارامتر می ت باید عددی بین . باشد و هر چه عدد انتخاب شده کمتر باشد ، وضوح تصویر عنصر هم کمتر خواهد
callback	توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و تغییر وضوح

تصویر عنصر مورد نظر ، آن تابع اجرا شود . به کار بردن این پارامتر اختیاری است .
---

در چند مثال سعی می کنیم تا استفاده از افکت ( ) fadeTo را در عمل به شما نمایش دهیم .

: در مثال اول یک تگ < div > داریم که در ابتدا به صورت عادی نمایش داده می شود . کاربر می تواند با کلیک بر روی دکمه فرمان تعیین شده وضوح تصوی آن تگ را در مدت ثانیه به مقدار نصف کاهش دهد . برای مشاهده مثال در عمل بر روی دکمه فرمان در خروجی کلیک نمایید :


#### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > </script >
  < script type="text/javascript" >
    $(document).ready ( function( ) {
      $( "# Btn_1 " ) . click ( function( ){
        $( " # div_1 " ) . fadeTo ( 2000 , 0.5 )
      });
    });
  </script>
</head>
<body>
  < input type="button" id="Btn_1" value="click me for fadeTo half!" />
  <div id= "div_1" style = "font-size: medium ; border: solid 1px black " >
    در این سایت آموزش زبان های زیر را داریم
    <ul>
      <li>HTML </li>
      <li>CSS </li>
      <li>Java Script </li>
      <li>jQuery </li>
    </ul>
  </div >
</body>
```

کد

</html>	
<p>در این سایت آموزش زبان های زیر را داریم</p> <ul style="list-style-type: none"> <li>• HTML</li> <li>• CSS</li> <li>• Java Script</li> <li>• jQuery</li> </ul>	خروجی

: در مثال دوم یک عکس ( < div > ) داریم که در ابتدا به صورت عادی است . کاربر می تواند با کلیک بر روی دکمه فرمان مثال وضوح نمایش آن عکس را به میزان خیلی کمی کاهش دهد . برای مشاهده خروجی مثال بر روی دکمه فرمان کلیک نمایید . توجه داشته باشید این افکت به علت استفاده از پارامتر "slow" به آرامی انجام می شود :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# Btn_2 " ) . click ( function ( ){         \$( "# Pic_1 " ) . fadeTo ( "slow" , 0.1 )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_2" value="click me for fadeTo!" /&gt;   &lt; img alt="firefox" src=" ../Pic/firefox.gif" id="Pic_1" /&gt; &lt; /body &gt; &lt; /html &gt; </pre>	کد
	خروجی

## افکت ها در jQuery - افکت hide

افکت hide ( ) jQuery :

این افکت باعث مخفی شدن عنصر مورد نظر می شود . عملکرد این افکت بسیار شبیه به افکت [fadeOut](#) می باشد . بوسیله این افکت برای مثال می توانید این اختیار را به کاربر بدهید تا عناصری را که در صفحه لازم ندارد ، مخفی کند . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).hide ( speed , callback )</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector عنصری است که این افکت باعث مخفی شدن آن می شود .	selector
<p>این پارامتر در این افکت اختیاری بوده و مقدار پیش فرض آن صفر است .</p> <p>این پارامتر سرعت مخفی شدن عنصر توسط افکت را تعیین می کند .</p> <p>این پارامتر به دو صورت قابل مقداردهی می باشد .</p> <ul style="list-style-type: none"><li>• تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر مخفی شود را تعیین می کنید . ms .</li><li>• تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت مخفی شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li>○ "slow"</li><li>○ "normal"</li><li>○ "fast"</li></ul></li></ul>	speed
توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان پافتن اجرای افکت و مخفی شدن .	callback

به کار بردن این پارامتر اختیاری است .

در چند مثال سعی می کنیم تا استفاده از افکت ( ) hide عمل به شما نمایش دهیم .

: در مثال اول یک تگ < div > داریم که در ابتدا نمایش داده می شود . یک دکمه فرمان طراحی کرده ایم که کاربر برای مخفی کردن تگ < div > باید بر روی آن کلیک نماید . با کلیک بر روی دکمه فرمان مثال ، تگ < div > مخفی می :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $(document). ready ( function( ) {
      $( "# Btn_1 " ) . click ( function( ){
        $( "# div_1 " ) . hide( )
      });
    });
  < /script >
< /head >
< body >
  < input type="button" id="Btn_1" value="click me for hide!" />
  < div id= "div_1" style = "font-size: medium ; border: solid 1px black " >
    در این سایت آموزش زبان های زیر را داریم
    < ul >
      < li > HTML < /li >
      < li > CSS < /li >
      < li > Java Script < /li >
      < li > jQuery < /li >
    < /ul >
  < /div >
```

کا

<pre>&lt;/body&gt; &lt;/html&gt;</pre>	
<p>در این سایت آموزش زبان های زیر را داریم</p> <ul style="list-style-type: none"> <li>• HTML</li> <li>• CSS</li> <li>• Java Script</li> <li>• jQuery</li> </ul>	<p>خروجی</p>

: در مثال دوم دو لیست از اسامی زبان های برنامه نویسی را با آیتم های مختلف به کاربر نمایش داده ایم . در ابتدا آیتم های لیست ها به کاربر نمایش داده می شود . اما کاربر می تواند با برداشتن تیک کنترل انتخابی checkbox در مثال ، آیتم های لیست را مخفی کند . برای مخفی کردن آیتم های لیست ها تیک کادر انتخاب را در خروجی مثال بردارید :

**راهنمایی مثال :** در این مثال افکت hide اعضای کلاس Items را که همان آیتم های لیست ها هستند ، را مخفی می کند .

<p>Example</p>	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ( "# Check_1 " ) . change ( function ( ){         \$ ( ".Items " ) . hide ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="checkbox" id="Check_1" checked="checked" /&gt; نمایش زیر مجموعه ها   طراحی وب</pre>	<p>کد</p>



<pre> &lt;ul&gt;   &lt;li class="Items"&gt;HTML &lt;/li&gt;   &lt;li class="Items"&gt;CSS &lt;/li&gt;   &lt;li class="Items"&gt;Flash &lt;/li&gt; &lt;/ul&gt; برنامه نویسی وب &lt;ul&gt;   &lt;li class="Items"&gt;ASP.Net &lt;/li&gt;   &lt;li class="Items"&gt;PHP &lt;/li&gt;   &lt;li class="Items"&gt;Java Script &lt;/li&gt;   &lt;li class="Items"&gt;jQuery &lt;/li&gt; &lt;/ul&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p style="text-align: right;"> <input checked="" type="checkbox"/> نمایش زیر مجموعه ها  طراحی وب </p> <ul style="list-style-type: none"> <li>HTML •</li> <li>CSS •</li> <li>Flash •</li> </ul> <p style="text-align: right;">برنامه نویسی وب</p> <ul style="list-style-type: none"> <li>ASP.Net •</li> <li>PHP •</li> <li>Java Script •</li> <li>jQuery •</li> </ul>	<p>خروجی</p>

## کتاب ها در jQuery - افکت show

افکت show ( ) jQuery :

این افکت عنصری که مخفی است را نمایان می کند . عملکرد این افکت همانند افکت [fadeIn](#) می باشد . بدیهی است عنصری که با این افکت نمایان می شود باید در ابتدا مخفی باشد .

این افکت بر روی عناصری که با یکی از افکت های [hide](#) یا [fadeOut](#) jQuery مخفی شده اند و یا خاصیت display آنها در CSS none باشد ، اثر خواهد داشت .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).show ( speed , callback )</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector عنصری است که این افکت باعث نمایان شدن آن می شود .	selector
این پارامتر در این افکت اختیاری بوده و مقدار پیش فرض آن صفر است . این پارامتر سرعت نمایان شدن عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقاردهی می باشد . <ul style="list-style-type: none"><li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر نمایان شود را تعیین می کنید . ms .</li><li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت نمایان شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li>"slow" ○</li><li>"normal" ○</li><li>"fast" ○</li></ul></li></ul>	speed

توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و نمایان شدن به کار بردن این پارامتر اختیاری است .	callback
--	----------

در چند مثال سعی می کنیم تا استفاده از افکت ( show ) را در عمل به شما نمایش دهیم .

: در مثال اول یک تصویر ( < img > ) داریم که در ابتدا مخفی است . یک دکمه فرمان طراحی کرده ایم که کاربر می تواند با کلیک بر روی آن باعث نمایان شدن و نمایش عکس شود . برای مشاهده عکس مخفی بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# Btn_1 " ) . click ( function ( ){         \$( " # img_1 " ) . show ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_1" value="click me for show image!" /&gt;   &lt; img id="img_1" src=" ../Pic/chrome.gif" /&gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد
	خروجی

: در مثال دوم دو لیست از اسامی زبان های برنامه نویسی را با آیتم های مختلف به کاربر نمایش داده ایم . در ابتدا آیتم های لیست ها مخفی هستند . اما کاربر می تواند با علامت زدن تیک کنترل انتخابی checkbox در مثال ، آیتم های لیست را نمایان کند . برای نمایان کردن آیتم های لیست ها تیک کادر انتخاب را در خروجی مثال علامت بزنید :

**راهنمایی مثال :** در این مثال افکت show اعضای کلاس Items را که همان آیتم های لیست ها هستند ، را نمایان می کند .

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ (" # Check_1 ") . change ( function ( ){         \$ ( " .Items " ) . show ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="checkbox" id="Check_1" /&gt; نمایش زیر مجموعه ها   طراحی وب   &lt; ul &gt;     &lt; li class="Items"&gt;HTML &lt; /li &gt;     &lt; li class="Items"&gt;CSS &lt; /li &gt;     &lt; li class="Items"&gt;Flash &lt; /li &gt;   &lt; /ul &gt;   برنامه نویسی وب   &lt; ul &gt;     &lt; li class="Items"&gt;ASP.Net &lt; /li &gt;     &lt; li class="Items"&gt;PHP &lt; /li &gt;     &lt; li class="Items"&gt;Java Script &lt; /li &gt;     &lt; li class="Items"&gt;jQuery &lt; /li &gt;   &lt; /ul &gt; </pre>	<p>کد</p>

<pre>&lt;/body&gt; &lt;/html&gt;</pre>	
<p>نمایش زیر مجموعه ها <input type="checkbox"/></p> <p>طراحی وب</p> <p>برنامه نویسی وب</p>	<p>خروجی</p>

## افکت ها در jQuery - افکت slideDown

افکت `slideDown()` jQuery :

این افکت با افزایش تدریجی ارتفاع یک عنصر مخفی ، آن را از حالت مخفی به حالت نمایان در می آورد . در واقع این افکت این کار را در قالب یک افکت تصویری انجام می دهد . بدیهی است این افکت بر روی عنصری موثر است که در ابتدا مخفی

این افکت بر روی عناصری که با یکی از افکت های `hide` یا `fadeOut` jQuery مخفی شده اند و یا خاصیت `display` آنها در `css` `none` باشد ، اثر خواهد داشت .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).slideDown ( speed , callback )</code>
--------	--

توضیح هر یک از موارد `syntax` :

توضیح syntax	
توضیح	Parameter
<code>selector</code> عنصری است که این افکت باعث نمایان شدن آن می شود .	<code>selector</code>
این پارامتر در این افکت اختیاری بوده و مقدار پیش فرض آن صفر است . این پارامتر سرعت نمایان شدن عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقداردهی می باشد . <ul style="list-style-type: none"><li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر نمایان شود را تعیین می کنید . <code>ms</code> .</li><li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت نمایان شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li><code>"slow"</code> ○</li><li><code>"normal"</code> ○</li><li><code>"fast"</code> ○</li></ul></li></ul>	<code>speed</code>

<p>توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و نمایان شدن .</p> <p>به کار بردن این پارامتر اختیاری است .</p>	<p>callback</p>
---	-----------------

در چند مثال سعی می کنیم تا استفاده از افکت ( `slideDown` ) را در عمل به شما نمایش دهیم .

در مثال اول یک تصویر ( `<img>` ) داریم که در ابتدا مخفی است . یک دکمه فرمان طراحی کرده ایم که کاربر می تواند با کلیک بر روی آن باعث نمایان شدن و نمایش عکس شود . برای مشاهده عکس مخفی بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$ ( "# Btn_1 " ) . click ( function ( ){         \$ ( " # img_1 " ) . slideDown ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_1" value="click me for show image!" /&gt;   &lt; img id="img_1" src=" ../Pic/computer_banner.jpg" style="display: none" /&gt; &lt; /body &gt; &lt; /html &gt; </pre>	<p>کد</p>
	<p>خروجی</p>

: مثال قبلی را بازنویسی نموده ایم با این تفاوت که در آن پارامتر speed افکت را با مقدار ثانیه مقدار دهی کرده ایم . برای مشاهده عکس مخفی و تاثیر پارامتر speed بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# Btn_2 " ) . click ( function ( ){         \$( " # img_2 " ) . slideDown ( 2000 )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_2" value="click me for show image!" /&gt;   &lt; img id="img_2" src=" ../Pic/computer_banner.jpg" style="display: none" /&gt; &lt; /body &gt; &lt; /html &gt; </pre>	کد
	خروجی



## افکت ها در jQuery - افکت slideUp

افکت `slideUp ( )` jQuery :

این افکت با کاهش تدریجی ارتفاع یک عنصر باعث مخفی شدن آن می شود . در واقع این افکت این کار را در قالب یک افکت تصویری انجام می دهد .

شکل کلی استفاده از این متد به شرح زیر است :


Syntax	<code>\$( " selector " ).slideUp ( speed , callback )</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
<code>selector</code> عنصری است که این افکت باعث مخفی شدن آن می شود .	<code>selector</code>
این پارامتر در این افکت اختیاری بوده و مقدار پیش فرض آن صفر است . این پارامتر سرعت مخفی شدن عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقداردهی می باشد . <ul style="list-style-type: none"><li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر مخفی شود را تعیین می کنید . ms .</li><li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت مخفی شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li>"slow" ○</li><li>"normal" ○</li><li>"fast" ○</li></ul></li></ul>	<code>speed</code>
توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت و مخفی شدن . به کار بردن این پارامتر اختیاری است .	<code>callback</code>

در چند مثال سعی می کنیم تا استفاده از افکت `slideUp ( )` را در عمل به شما نمایش دهیم .

: در مثال اول یک تصویر ( < img > ) داریم . یک دکمه فرمان طراحی کرده ایم که کاربر می تواند با کلیک روی آن باعث مخفی شدن عکس شود . برای مخفی کردن عکس بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# Btn_1 " ) . click ( function ( ){         \$( "# img_1 " ) . slideUp ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_1" value="click me to hide image!" /&gt;   &lt; img id="img_1" src=" ../Pic/computer_banner.jpg" " /&gt; &lt; /body &gt; &lt; /html &gt; </pre>	کنا
	خروجی

## افکت ها در jQuery - افکت slideToggle

افکت slideToggle ( ) : jQuery

این افکت همانند یک عملگر دوجانبه بین دو افکت [slideUp](#) و [slideDown](#) بر روی عنصر مورد اثر خود عمل می کند .  
به عبارت دیگر افکت slideToggle ابتدا وضعیت نمایش و یا عدم نمایش عنصر را بررسی کرده و در صورتی که عنصر نمایان باشد ، با کاهش تدریجی ارتفاع آن باعث مخفی شدن عنصر می . و چنانچه عنصر مخفی باشد با افزایش تدریجی ارتفاع آن ، عنصر را نمایان می سازد .  
این افکت کاربرد بسیار سودمندی می تواند داشته باشد . به وسیله آن می توانید این اختیار را به کاربر بدهید تا عناصری را که می خواهد در صفحه مخفی کرده و یا آنها را نمایان سازد .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).slideToggle ( speed , callback )</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector	عنصری است که این افکت باعث مخفی و یا نمایان شدن آن می شود .
speed	این پارامتر در این افکت اختیاری بوده و مقدار پیش فرض آن صفر است . این پارامتر سرعت مخفی و یا نمایان شدن عنصر توسط افکت را تعیین می کند . این پارامتر به دو صورت قابل مقاداردهی می باشد . <ul style="list-style-type: none"><li>تعیین به وسیله زمان بر حسب میلی ثانیه : در این حالت مدت زمانی که می خواهید در آن عنصر مخفی یا نمایان شود را تعیین می کنید . ms</li><li>تعیین بر اساس حالت کلی زیر : در این حالت با انتخاب یکی از مقدار زیر سرعت مخفی یا نمایان شدن را آرام ، معمولی و یا سریع انتخاب می نمایید .<ul style="list-style-type: none"><li>"slow" ○</li><li>"normal" ○</li></ul></li></ul>

"fast" ○	
توسط این پارامتر می توان نام یک تابع را تعیین کنید تا پس از پایان یافتن اجرای افکت ، آن تابع اجرا به کار بردن این پارامتر اختیاری است .	<b>callback</b>

در چند مثال سعی می کنیم تا استفاده از افکت ( ) slideToggle را در عمل به شما نمایش دهیم .

: در مثال اول یک تصویر ( < img > ) داریم . یک دکمه فرمان طراحی کرده ایم که کاربر می تواند با کلیک بر روی آن باعث تغییر وضعیت نمایش عکس شود . با کلیک بر روی دکمه فرمان چنانچه عکس نمایان باشد آن را مخفی کرده و یا برعکس . برای تغییر وضعیت نمایش عکس بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document). ready ( function( ) {       \$ ( " # Btn_1 " ) . click ( function( ){         \$ ( " # img_1 " ) . slideToggle( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="button" id="Btn_1" value="click me to toggle image!" /&gt;   &lt; img id="img_1" src=" ../Pic/computer_banner.jpg" " /&gt; &lt; /body &gt; &lt; /html &gt; </pre>	کد
	خروجی



: مثال دوم دو لیست از اسامی زبان های برنامه نویسی را با آیتم های مختلف به کاربر نمایش داده ایم . در ابتدا آیتم های لیست ها به کاربر نمایش داده می شود . اما کاربر می تواند با برداشتن تیک کنترل انتخابی checkbox در مثال ، آیتم های لیست را مخفی کرده و دوباره با علامت زدن آن ، آیتم را نمایش دهد . برای مخفی کردن و نمایش دوباره آیتم های لیست ها تیک کادر انتخاب را در خروجی مثال تغییر دهید :

**راهنمایی مثال :** در این مثال افکت slideToggle اعضای کلاس Items را که همان آیتم های لیست ها هستند ، را مخفی و یا نمایان می کند .

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$( "# Check_1 " ) . change ( function ( ){         \$( ".Items " ) . slideToggle ( )       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; input type="checkbox" id="Check_1" checked="checked" /&gt; نمایش زیر   مجموعه ها   طراحی وب   &lt; ul &gt; </pre>	<p>کد</p>

<pre> &lt;li class="Items"&gt;HTML &lt;/li&gt; &lt;li class="Items"&gt;CSS &lt;/li&gt; &lt;li class="Items"&gt;Flash &lt;/li&gt; &lt;/ul&gt; برنامه نویسی وب &lt;ul&gt; &lt;li class="Items"&gt;ASP.Net &lt;/li&gt; &lt;li class="Items"&gt;PHP &lt;/li&gt; &lt;li class="Items"&gt;Java Script &lt;/li&gt; &lt;li class="Items"&gt;jQuery &lt;/li&gt; &lt;/ul&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
	<p style="text-align: right;"> <input checked="" type="checkbox"/> نمایش زیر مجموعه ها  طراحی وب </p> <ul style="list-style-type: none"> <li>HTML •</li> <li>CSS •</li> <li>Flash •</li> </ul> <p style="text-align: right; color: red;">خروجی</p> <p style="text-align: right;">برنامه نویسی وب</p> <ul style="list-style-type: none"> <li>ASP.Net •</li> <li>PHP •</li> <li>Java Script •</li> <li>jQuery •</li> </ul>

## افکت ها در jQuery - افکت stop

افکت stop ( ) : jQuery

به کار بردن این افکت باعث توقف انجام یک عمل یا انیمیشن برای عنصر مورد نظر می شود . برای مثال فرض کنید که یک Div دارید که می خواهید ارتفاع آن را از به بیکسل افزایش دهید . انجام این کار چند ثانیه طول می کشد . می توانید دکمه فرمانی قرار دهید تا با اجرای افکت ( ) stop انجام انیمیشن فوق را برای تگ Div سطر فرایند متوقف

شکل کلی استفاده از این افکت بصورت زیر است :

Syntax	<code>\$( selector ).stop( stopAll , goToEnd ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
selector تعیین کننده id عنصری است که می خواهیم اجرای یک افکت بر روی آن با متد ( ) stop	selector
این پارامتر یک مقدار Boolean را تعیین می کند . مقدار پیش فرض آن false است ولی اگر روی true تنظیم شود، در هنگام اجرای افکت stop ، کلیه انیمیشن های مرتبط با این افکت را به صورت یکباره متوقف می سازد . استفاده از این پارامتر اختیاری است .	stopAll
این پارامتر یک مقدار Boolean را تعیین می کند . مقدار پیش فرض آن false است ولی اگر روی true تنظیم شود ، در هنگام اجرای افکت stop ، انیمیشن های مرتبط با آن را به یکباره متوقف کرده و به مرحله آخر می رساند . استفاده از این پارامتر اختیاری است .	goToEnd

مثال عملی :

**توقف اجرای افکت در وسط پروسه :** در مثال زیر یک افکت ( ) animate را افزایش می دهد .  
با کلیک بر روی دکمه فرمان stop می توانید انجام این عملیات را در وسط پروسه ، متوقف نمایید :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      var div1 = $("#div1");
      $("#start1").click(function(){
        div1.animate({height:300},"slow");
        div1.animate({width:300},"slow");
        div1.animate({height:100},"slow");
        div1.animate({width:100},"slow");
      });

      $("#stop1").click(function(){
        div1.stop(true);
      });
    });
  < /script >;
< /head >
< body >
  < p >
    < button id="start1" > Start < /button >
    < button id="stop1" > Stop < /button >
  < /p >

  < div id ="div1" style="background: #9 ( ) stopbf21; height: 100px; width: 100px">
< /div >
< /body >
< /html >
```

کا

Start Stop

خروجی



**توقف و اتمام یکباره افکت در حال اجرا :** در مثال زیر یک افکت ( ) animate را افزایش می دهد . با کلیک بر روی دکمه فرمان stop می توانید انجام این عملیات را در وسط پروسه ، متوقف کرده و به یکباره به انتهای آن برسانید :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      var div2 = $("#div2");
      $("#start2").click(function(){
        div2.animate({height:300},"slow");
        div2.animate({width:300},"slow");
        div2.animate({height:100},"slow");
        div2.animate({width:100},"slow");
      });

      $("#stop2").click(function(){
        div2.stop(true,true);
      });
    });
  < /script >;
< /head >
< body >
  < p >
    < button id="start2" > Start < /button >
    < button id="stop2" > Stop < /button >
  < /p >

  < div id="div2" style="background: #9( ) stopbf21; height: 100px; width:
100px" > < /div >
```

کد

<code>&lt;/body&gt;</code> <code>&lt;/html&gt;</code>	
Start Stop	خروجی

## رویدادها در jQuery - رویداد ( ) toggle

رویداد toggle ( ) jQuery :

افکت ( ) toggle بین دو افکت [show \( \)](#) و [hide \( \)](#) تغییر وضعیت می دهد .

افکت ( ) toggle وضعیت نمایش و یا عدم نمایش عنصر مورد نظر را بر روی صفحه می سنجد . اگر عنصر در حال نمایش Hide مخفی کرده و اگر عنصر مخفی باشد آن را با استفاده از متد Show نمایش می دهد .

**نکته :** این متد برای عناصری که از ابتدای اجرای صفحه مخفی تعیین شده اند ، کار نمی کند .

برای مشاهده نحوه کار افکت ( ) toggle به مثال عملی دقت نمایید .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( selector ).toggle( speed , easing , callback ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم افکت ( ) toggle ، بر روی آن اجرا شود .	selector
این پارامتر تعیین کننده سرعت انجام افکت و تغییر حالت عنصر از حالت اول به دوم است . استفاده از این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از : ( MiliSecond : مدت زمان اجرای افکت بر حسب میلی ثانیه . ( Slow : سرعت آهسته ( Fast : سرعت زیاد	speed
این پارامتر تعیین کننده سرعت انجام افکت در ابتدا و انتهای آن است . استفاده از این پارامتر اختیاری است . انواع مقادیر ممکن عبارتند از : ( Swing : سرعت انجام افکت در ابتدا و انتهای فرایند آهسته ولی در وسط آن زیاد است . ( Liner : سرعت در کل طول فرایند یکسان است .	easing

<p>به وسیله این پارامتر می توانیم یک تابع را تعیین کنیم ، تا در پایان انجام افکت ( toggle ) استفاده از این پارامتر اختیاری است .</p>	<p>function</p>
--	-----------------

: در مثال زیر یک پاراگراف داریم . رای آن یک رویداد ( toggle ) نوشته ایم ، با هر بای کلیک کاربر بر روی دکمه فرمان رویداد ( toggle ) برای آن اجرا شده و وضعیت نمایش یا عدم نمایش آن تغییر می کند :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready(function () {       \$("#btn1").click(function(){         \$("#p1").toggle();       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; p id="p1" style="font-size: 16px"&gt; نحوه تغییر وضعیت عنصر &lt; /p &gt;   &lt; button id="btn1" type="button"&gt; Toggle ( ) &lt; /button &gt; &lt; /body &gt; &lt; /html &gt; </pre>	<p>کد</p>
<p>نحوه تغییر وضعیت عنصر</p> <p>Toggle ( )</p>	<p>خروجی</p>

## کاربرد jQuery - HTML ( ) after

### jQuery ( ) after :

به وسیله متد `jQuery after` ، می توانید محتویات مورد نظر خود ( می تواند شامل متن یا تگ های HTML ) از عنصر مورد نظر خود ، در صفحه HTML وارد نمایید .  
برای مثال فرض کنید که یک پاراگراف در صفحه دارید . به وسیله این متد می توانید یک پاراگراف یا یک تگ دیگر HTML را پس از آن عنصر ، به صفحه اضافه نمایید .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( " selector " ).after ( content ) ;</code>
--------	---

توضیح هر یک از موارد `syntax` :

توضیح syntax	
توضیح	Parameter
عنصر مورد نظری است که می خواهید محتویات مورد نظرتان را پس از آن ، در صفحه وارد نمایید .	selector
تعیین کننده محتویاتی است که می خواهید پس از عنصر مورد نظر در صفحه وارد نمایید . این محتویات می تواند شامل متن یا سایر تگ های HTML .	content

<code>\$( " p#P1 " ).after ( " &lt; p &gt; This is what you want to add &lt; /p &gt; " ) ;</code>
---

### مثال عملی:

در مثال زیر استفاده از این متد را در عمل به شما نمایش داده ایم .  
در این مثال ، یک پاراگراف داریم که درباره `jQuery` توضیحات مختصری را ارائه کرده است . اما یک پاراگراف دوم نیز در نظر گرفتیم که از دید کاربر مخفی است و قرار است توضیحات بیشتری راجع به `jQuery` نمایش دهد . کاربر در صورت تمایل می تواند با کلیک بر روی دکمه فرمان موجود در خروجی مثال ، پاراگراف دوم را نیز به صفحه اضافه کرده و آن را

برای مشاهده خروجی مثال ، بر روی دکمه فرمان در خروجی کلیک نمایید :

Example	
<pre>&lt;p id="P_1"&gt;   jQuery is a new scripting method . &lt;/p&gt; &lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt;   \$(document).ready( function ( ) {     \$("button").click (function ( ) {       \$( "#P_1" ).after ( " &lt; p &gt; with jQuery you can do very useful things . you can hide or toggle objects &lt; /p &gt; " );     });   }); &lt;/script&gt; &lt;button&gt; Click to add second paragraph&lt;/button&gt;</pre>	کد
<p>jQuery is a new scripting method .</p> <p>Click to add second paragraph</p>	خروجی

## کاربرد jQuery - HTML attr ( )

### jQuery ( ) attr :

jQuery attr ، برای منظور می تواند به کار برده شود :

مقدار تعیین شده برای خواص عناصر را خوانده و به کاربر نمایش می دهد . برای مثال فرض کنید که برای خاصیت

خوانده و به کاربر نمایش دهد .  
( width ) یک عنصر عکس `<img>` را تعیین کرده اید . این متد می تواند این مقدار را

می تواند مقدار خاصیت عناصر را تعیین کرده و یا در صورت نیاز تغییر دهد . برای مثال به وسیله این متد می توانید  
سایز فونت یک پاراگراف `<p>` را به مقدار `pt` تغییر دهید .

در ادامه به تشریح نحوه استفاده از این متد در هر دو روش ذکر شده می پردازیم .

### کاربرد اول - نمایش مقدار خاصیت یک عنصر :

در کاربرد اول این متد می خواهیم به شما نمایش دهیم چگونه می توان مقدار خاصیت یک عنصر را استخراج کرده و سپس  
نمایش دهیم .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$ ( " selector " ).attr ( attribute ) ;</code>
--------	---

توضیح هر یک از موارد syntax :


توضیح syntax	
توضیح	Parameter
عنصر مورد نظری است که می خواهیم مقدار یک خاصیت آن را نمایش دهیم .	selector
نام خاصیتی از عنصر مورد نظر است که این متد قرار است مقدار آن را استخراج کرده و به	attribute

کاربر نمایش دهد .	
-------------------	--

\$( "#p#P1" ).attr ( " width " ) ;	
------------------------------------	--

### مثال عملی:

در مثال زیر یک عنصر تصویر داریم که می خواهیم مقدار عرض ( خاصیت width ) آن را به کاربر نمایش دهیم . برای مشاهده خروجی بر روی دکمه فرمان خروجی مثال کلیک نمایید :

Example	
<pre>&lt;img id="Img_1" src="../Pic/ADS/JQ1.jpg" width="150px" height="100px" /&gt; &lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt;     \$(document).ready( function ( ) {         \$("button").click (function ( ) {             alert ( " Image size is = " + \$("# Img_1").attr ( "width" ) ) ;         });     }); &lt;/script&gt; &lt;button&gt; برای نمایش مقدار خاصیت کلیک نمایید &lt;/button&gt;</pre>	کد
 <p>برای نمایش مقدار خاصیت کلیک نمایید</p>	خروجی

کاربرد دوم - تغییر مقدار خاصیت یک عنصر :



در کاربرد دوم این متد می خواهیم به شما نمایش دهیم چگونه می توان مقدار خاصیت یک عنصر را به مقدار مورد نظرمان تغییر دهیم .

شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$ ( " selector " ).attr ( attribute , value ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
عنصر مورد نظری است که می خواهیم مقدار یک خاصیت آن را به مقدار دلخواه تغییر دهیم .	selector
نام خاصیتی از عنصر مورد نظر است که این متد قرار است مقدار آن را تغییر دهد . برای مثال مقدار خاصیت عرض width یک عنصر را به یک مقدار دیگر تغییر دهیم .	attribute
مقدار جدید تعیین شده است که می خواهیم مقدار خاصیت را به آن مقدار ، تغییر دهیم .	value

	<code>\$ ( " #img_2 " ).attr ( " width " , " 150 " ) ;</code>
--	---

**مثال عملی:**

در مثال زیر یک عکس ( `<img >` ) داریم ، که می خواهیم مقدار عرض آن را به px تغییر دهیم . برای این منظور بر روی دکمه فرمان در خروجی مثال کلیک نمایید :

Example	
<code>&lt;img id="lmg_2" src="../Pic/ADS/JQ1.jpg" width="100px" height="100px" /&gt;</code>	کد

<pre>&lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt; \$(document).ready( function ( ) {     \$("button").click (function ( ) {         \$("# Img_2").attr ( "width" , "200" ) );     }); }); &lt;/script&gt; &lt;button&gt; برای تغییر مقدار خاصیت کلیک نمایید &lt;/button&gt;</pre>	
 <p>برای تغییر مقدار خاصیت کلیک نمایید</p>	خروجی

## کاربرد jQuery HTML - append ( )

### append ( ) jQuery :

jQuery append می تواند یک متن را به ادامه یک پاراگراف دیگر اضافه نماید . متن اضافه شده به ادامه پاراگراف اضافه می شود نه اینکه در سطر بعد ( after ( ) ) .  
از این متد برای زمانی استفاده می شود که بخواهید یک متن را به ادامه یک پاراگراف موجود در صفحه اضافه نمایید .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$ ( " selector " ).append ( content ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
پاراگراف یا عنصر مورد نظری است که می خواهید متن مورد نظرتان را به ادامه آن اضافه نمایید .	selector
تعیین کننده محتویاتی است که می خواهید به ادامه عنصر مورد نظر در صفحه اضافه نمایید . این محتویات می تواند شامل متن باشد .	content

	<code>\$ ( " p#P1 " ).append ( " &lt; b &gt; This is what you want to add &lt; /b &gt; " ) ;</code>
--	---

### مثال عملی:

در مثال زیر استفاده از این متد را در عمل به شما نمایش داده ایم .  
در این مثال ، یک پاراگراف داریم که درباره jQuery توضیحات مختصری را ارائه کرده است . اما یک ادامه متنی را نیز در نظر گرفتیم که از دید کاربر مخفی است و قرار است توضیحات بیشتری راجع به jQuery نمایش دهد . کاربر در صورت

تمایل می تواند با کلیک بر روی دکمه فرمان موجود در خروجی مثال ، متن اضافه را پاراگراف جاری اضافه کرده و آن را

برای مشاهده خروجی مثال ، بر روی دکمه فرمان در خروجی کلیک نمایید :

Example	
<pre>&lt;p id="P_1"&gt;   jQuery is a new scripting method . &lt;/p&gt; &lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt;   \$(document).ready( function ( ) {     \$("button").click (function ( ) {       \$( "#P_1" ).append ( " &lt; b &gt; with jQuery you can do very useful things . you can hide or toggle objects . &lt; /b &gt; " );     });   }); &lt;/script&gt; &lt;button&gt; Click to add second paragraph&lt;/button&gt;</pre>	کد
<p>jQuery is a new scripting method .</p> <p>Click to add second paragraph</p>	خروجی

## کاربرد jQuery HTML - before ( )

### jQuery : ( ) before

jQuery before برای وارد نمودن یک متن یا پاراگراف در سطر قبلی یک پاراگراف استفاده می شود .  
این متد متن مورد نظر شما را به سطر قبلی پاراگراف تعیین شده در صفحه ، اضافه می کند .  
شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$ ( " selector " ).before ( content ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
عنصر مورد نظری است که می خواهید محتویات مورد نظرتان را قبل از آن ، در صفحه وارد نمایید .	selector
تعیین کننده محتویاتی است که می خواهید قبل از عنصر مورد نظر در صفحه وارد نمایید . این محتویات می تواند شامل متن یا سایر تگ های HTML .	content

<code>\$ ( " p#P1 " ).before ( " &lt; p &gt; This is what you want to add &lt; /p &gt; " ) ;</code>
---

### مثال عملی:

در مثال زیر استفاده از این متد را در عمل به شما نمایش داده ایم .  
در این مثال ، یک پاراگراف داریم که jQuery توضیحات مختصری را ارائه کرده است . اما یک پاراگراف دوم نیز در نظر گرفتیم که از دید کاربر مخفی است و قرار است توضیحات بیشتری راجع به jQuery نمایش دهد . کاربر در صورت تمایل می تواند با کلیک بر روی دکمه فرمان موجود در خروجی مثال ، پاراگراف دوم را نیز به صفحه اضافه کرده و آن را برای مشاهده خروجی مثال ، بر روی دکمه فرمان در خروجی کلیک نمایید :

Example	
<pre> &lt;p id="P_1"&gt;   jQuery is a new scripting method . &lt;/p&gt; &lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt;   \$(document).ready( function ( ) {     \$("button").click (function ( ) {       \$( "#P_1" ).before ( " &lt; p &gt; with jQuery you can do very useful things . you can hide or toggle objects &lt; /p &gt; " );     });   }); &lt;/script&gt; &lt;button&gt; Click to add second paragraph&lt;/button&gt; </pre>	کد
<p>jQuery is a new scripting method .</p> <p>Click to add second paragraph</p>	خروجی

## کاربرد jQuery - HTML - CSS ( )

### متد CSS ( ) : jQuery

jQuery CSS می تواند یک یا چند خاصیت CSS یک عنصر HTML موجود در صفحه را برگردانده و یا به مقدار دلخواه تغییر دهد .  
به عبارت دیگر شما به وسیله این متد می توانید به مقدار خواص CSS یک عنصر مورد نظر دسترسی داشته و یا آن خواص را به مقدار دلخواه تغییر دهید .  
برای مثال می توانید به مقدار خاصیت رنگ نوشته یک پاراگراف ( خاصیت Color ) دسترسی داشته و یا آن را به یک مقدار دلخواه تغییر دهید .  
این متد در حالت می تواند مورد استفاده قرار گیرد . در ادامه به تشریح این حالت خواهیم پرداخت .

### : دسترسی به مقدار خاصیت CSS یک عنصر:

در حالت اول شما می توانید به مقدار یک خاصیت CSS عنصر مورد نظرتان دسترسی داشته و مقدار آن را به صفحه برگردانید .  
شکل کلی استفاده از متد CSS در حالت اول به صورت زیر است :

Syntax	<code>\$( " selector " ).css ( property ) ;</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
عنصر مورد نظری است که می خواهید محتویات مقدار یک خاصیت CSS آن را خوانده و یا به مقدار دلخواه تغییر دهید .	selector
این پارامتر تعیین کننده خاصیت CSS عنصری است که می خواهید مقدار آن را خوانده و یا به یک مقدار جدید تغییر دهید .	property

### نکته مهم : در هنگام استفاده از حالت خواندن مقدار خواص CSS یک عنصر توسط متد CSS :

در زمانی که شما می خواهید مقدار خاصیت CSS یک عنصر HTML را بخوانید ، این متد مقدار آن خاصیت را برای اولین عنصری از نوع تعیین شده که در صفحه پیدا کند ، بر می گرداند .

گر بخواهید رنگ متن یک پاراگراف را در صفحه بخوانید ، این متد رنگ اولین پاراگراف موجود در صفحه را بر می . مگر دقیقا در دستور خود ، id پاراگراف مورد نظر را تعیین نمایید .

: در مثال زیر ، رنگ پاراگراف موجود در مثال ، که خاصیت color آن می باشد را به وسیله متد CSS خروجی نمایش داده ایم :

Example	
<pre>&lt;p id="p_1" style="color: Green"&gt;     رنگ این پاراگراف در خروجی نوشته شده ، آیا درست است ؟ !! &lt;/p&gt;  &lt;script type="text/javascript" src="http://ajax.microsoft.com/ajax/jquery/jquery-1.4.2.min.js"&gt;&lt;/script&gt; &lt;script type="text/javascript"&gt;     \$(document).ready(function () {         document.getElementById("p_2").innerHTML = "Color : " +         (\$("#p_1").css("color"));     }); &lt;/script&gt;  &lt;p id="p_2"&gt; &lt;/p&gt;</pre>	کد
<pre>!! رنگ این پاراگراف در خروجی نوشته شده ، آیا درست  Color : green</pre>	



## کاربرد jQuery HTML - detach ( )

### jQuery ( ) detach :

( ) detach HTML تعیین شده برای آن ( شامل تمام متن و اشیای فرزند ) را از روی صفحه پاک می کند . اطلاعات و رویدادهای شی مورد نظر بر روی صفحه باقی می مانند . این متد همچنین یک کپی از عنصر پاک شده را در خود نگه داشته و بعدا می توانید آن عنصر را مجددا به صفحه وارد نمایید . برای مثال فرض کنید یک پاراگراف یا عکس را بر روی صفحه دارید . به وسیله این متد می توانید آن را از روی صفحه بردارید .

شکل کلی استفا ( ) detach به شرح زیر است :

Syntax	<code>\$( " selector " ).detach ( ) ;</code>
--------	--

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ی است که می خواهید آن را از صفحه پاک کنید .	selector

**نکته :** برای حذف کامل یک ( حذف عنصر به همراه تمامی متن و رویدادهای مربوط به آن ) `remove ( )` استفاده نمایید .

**نکته :** برای پاک کردن محتویات داخلی یک عنصر و نه حذف آن از صفحه ، از متد `empty ( )` استفاده نمایید .

**مثال عملی :** پاک کردن و وارد کردن مجدد یک عنصر HTML در صفحه .

در مثال زیر یک تابع طراحی کرده ایم . در کد این تابع دو کد قرار دارد که کد اول پاراگراف P\_EX را از صفحه پاک کرده و کد دوم آن را مجددا به صفحه وارد می نماید . برای مشاهده عملی مثال بر روی دکمه فرمان های تعیین شده کلیک نمایید :

Example	
<code>&lt;script type="text/javascript"&gt;</code>	کد

<pre> src="//ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;   \$(document).ready(function () {     var x;     \$("#btn1").click(function () {       x = \$("#P_EX").detach();     });     \$("#btn2").click(function () {       \$("body").prepend(x);     });   }); &lt;/script&gt;  &lt;p id="P_EX"&gt; This is a paragraph. &lt;/p&gt;  &lt;button id="Button1"&gt; برای پاک شدن پاراگراف کلیک نمایید &lt;/button&gt; &lt;button id="Button2"&gt; برای وارد کردن مجدد پاراگراف کلیک نمایید &lt;/button&gt; </pre>	
<p>This is a paragraph.</p> <p>برای وارد کردن مجدد پاراگراف کلیک نمایید برای پاک شدن پاراگراف کلیک نمایید</p>	

## کاربرد jQuery HTML - empty ( )

متد empty ( ) jQuery :

empty ( ) ، محتویات متنی و سایر اشیای فرزند عنصر HTML تعیین شده برای آن را پاک می کند . نکته مهم درباره متد empty ( ) اینست که این متد خود عنصر و یا خواص آن را حذف نمی کند ، بلکه فقط محتویات درونی آن را خالی می کند . بنابراین عنصر همچنان بر روی صفحه قرار داشته و قابل دسترس اس . برای مثال می توانید به وسیله متد empty ( ) ، محتویات درون یک پاراگراف یا آیتم های موجود در یک لیست HTML پاک نمایید و

شکل کلی استفاده از متد empty ( ) به شرح زیر است :

Syntax	<code>\$( " selector " ).empty ( ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ی است که می خواهید محتویات درون آن را پاک کنید .	selector

: در مثال زیر یک تابع تعریف کرده ایم که در صورت کلیک بر روی دکمه فرمان موجود در مثال ، محتویات پاراگراف EX را پاک می کند .

Example	
<pre>&lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.8.3/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt; \$(document).ready(function () {     \$("#btn1").click(function () {</pre>	کد

<pre>        \$("#EX").empty();     }); }); &lt;/script&gt; &lt;p id="EX"&gt; This is a paragraph. &lt;/p&gt;  &lt;button id="btn1"&gt;برای خالی شدن محتویات پاراگراف کلیک نمایید&lt;/button&gt;</pre>	
<p>This is a paragraph.</p> <p>برای خالی شدن محتویات پاراگراف کلیک نمایید</p>	

## کاربرد jQuery - HTML - ( ) html

html ( ) jQuery :

html ( ) jQuery دو کار را انجام می دهد :

- . به وسیله این متد می توانید محتویات درونی یک عنصر HTML یا همان خاصیت innerHtml آن را خوانده و به صفحه برگردانید . برای مثال می توان به محتویات درونی یگ تگ پاراگراف اشاره کرد.
- . به وسیله این متد می توانید محتویات درونی یک عنصر HTML را به مقدار دلخواه خود تغییر دهید .
- برای مثال می توانید محتویات یک پاراگراف در صفحه را به مقدار دلخواه خود تغییر دهید.

html ( ) :

در حالت اول گفتیم که شما می توانید به وسیله این متد ، محتویات درونی یک عنصر را بخوانید . شکل کلی استفاده از متد ( ) html در حالت اول به شرح زیر است :

Syntax	<code>\$ ( " selector " ).html ( ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده ID عنصری است که می خواهید محتویات درونی آن بخوانید .	selector

## کاربرد jQuery - HTML - height ( )

### jQuery ( ) height :

jQuery height ( ) می تواند مقدار ارتفاع یک عنصر HTML را خوانده و بر گرداند و یا به وسیله آن مقدار ارتفاع عنصر را به یک مقدار دلخواه تغییر دهید .

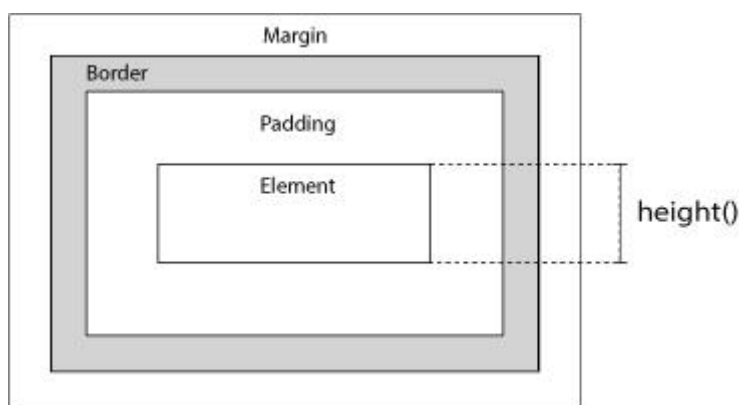
برای مثال اگر یک عنصر div داشته باشیم که ارتفاع آن 300 px height ( ) می توان دو کار را

:

مقدار ارتفاع آن را که 300 px است ، به صفحه برگردانیم .

مقدار جدیدی را برای ارتفاع عنصر تعیین کرده و آن را به مقدار جدید تغییر دهید .

**نکته :** همانطور که در تصویر زیر نمایش داده ایم ، متد height ( ) شامل حاشیه بیرونی ( margin ) ، حاشیه درونی ( padding ) و خطوط حاشیه عنصر ( border ) نمی شود و فقط ارتفاع خود عنصر را بر می گرداند .



### height ( ) - خواندن ارتفاع یک عنصر :

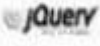
گفتیم که در حالت اول متد height ( ) ، از این متد برای خواندن مقدار ارتفاع یک عنصر و بازگرداندن آن مقدار به صفحه استفاده می شود . شکل کلی استفاده از متد height ( ) در حالت اول به شرح زیر است :

Syntax

```
$( " selector " ).height ( ) ;
```

در مثال زیر یک تصویر در مثال داریم . height ، ارتفاع این عنصر را خوانده و در صفحه نمایش

داده ایم :

Example	
<pre> &lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" height="70px" /&gt; &lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.height.3/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;     \$(document).ready(function () {         document.getElementById("EX").innerHTML = "    : " + \$("#Pic_1").height();     }); &lt;/script&gt; &lt;p id="EX"&gt;    &lt;/p&gt; </pre>	کد
 :	خروجی

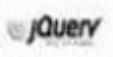
### height ( ) - تنظیم ارتفاع یک عنصر :

گفتیم که در حالت دوم متد height ( ) ، از این متد برای تنظیم مقدار ارتفاع یک عنصر به یک مقدار جدید استفاده می شود .  
 شکل کلی استفاده از متد height ( ) در حالت دوم به شرح زیر است :

Syntax	<pre> \$( " selector " ).height ( value ) ; * value : </pre>
--------	--

در مثال زیر یک تصویر در مثال داریم .  
 height ، ارتفاع این عنصر را به یک مقدار جدید تغییر داده ایم :

Example	
<pre> &lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" height="70px" /&gt; &lt;script type="text/javascript" </pre>	کد

<pre>src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;   \$(document).ready(function () {     \$("#Pic_2").height( "200px");   }); &lt;/script&gt;</pre>	
	خروجی



## کاربرد jQuery HTML - insertAfter ( )

متد insertAfter ( ) jQuery :

( ) insertAfter ، یک متن یا کنترل HTML را بعد از عنصر مورد نظرتان در صفحه وارد می نماید .  
در این متد بایستی یک متن یا عنصر HTML را انتخاب کرده و سپس تعیین نمایید تا پس از چه عنصری در صفحه وارد شود .  
برای مثال فرض کنید یک پاراگراف در صفحه دارید و می خواهید کدی طراحی نمایید تا در صورت کلیک کاربر بر روی یک دکمه فرمان ، متن جدیدی به انتهای پاراگراف اضافه شود .

شکل کلی استفاده از متد ( ) insertAfter به شرح زیر است :

Syntax	<code>\$( content ).insertAfter ( " selector " );</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده متن یا id HTML ای است که می خواهید محتویات آن را پس از عنصر مورد نظر در صفحه وارد نمایید .	content
تعیین کننده id عنصری است که می خواهید متن یا عنصر دیگر HTML صفحه وارد نمایید .	selector

## کاربرد jQuery HTML - insertBefore ( )

متد empty ( ) insertBefore :

insertBefore ( ) ، یک متن یا کنترل HTML را قبل از عنصر مورد نظران در صفحه وارد می نماید .  
در این متد بایستی یک متن یا عنصر HTML را انتخاب کرده و سپس تعیین نمایید تا قبل از چه عنصری در صفحه وارد شود .  
برای مثال فرض کنید یک پاراگراف در صفحه دارید و می خواهید کدی طراحی نمایید تا در صورت کلیک کاربر بر روی یک دکمه فرمان ، متن جدیدی به ابتدای پاراگراف اضافه شود .

شکل کلی استفاده از متد insertBefore ( ) به شرح زیر است :

Syntax	<code>\$( content ).insertBefore ( " selector " ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده متن یا id HTML ای است که می خواهید محتویات آن را قبل از نصر مورد نظر در صفحه وارد نمایید .	content
تعیین کننده id عنصری است که می خواهید متن یا عنصر دیگر HTML صفحه وارد نمایید .	selector

## کاربرد jQuery HTML - position ( )

### position ( ) jQuery :

position ( ) تصات محل قرار گیری یک عنصر را بر حسب عنصر مادر ( parent ) آن تعیین کرده و به صفحه بر می گرداند .

این تابع مختصات عنصر مورد نظر را با دو مقدار عددی نمایش می دهد که یکی تعیین کننده میزان فاصله از بالای عنصر ( parent ) است است که به آن top می گویند . دوم تعیین کننده میزان فاصله از سمت چپ عنصر مادر ( parent ) است که به آن left می گویند . واحد خروجی این مثال بر حسب پیکسل px می باشد .

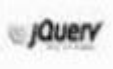
شکل کلی استفاده از متد ( ) position به شرح زیر است :

Syntax	<code>\$( " selector " ).position ( ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهید مختصات آن را نمایش دهید .	selector

: در مثال زیر با استفاده از متد ( ) position ، موقعیت قرار گیری عنصر عکس مثال را نسبت به عنصر مادر آن در صفحه نمایش داده ایم :

Example	
<pre>&lt;img src="../Pic/Icons/JQ1.jpg" id="Img1" /&gt; &lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.position.3/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt; \$(document).ready(function () {     x = \$("#Pic_1").position();     \$("#EX").text( + (x.top)); });</pre>	کد

<pre>\$("#EX2").text( + فاصله از چپ عکس }); &lt;/script&gt; &lt;span id="EX"&gt;&lt;/span&gt; &lt;span id="EX2"&gt;&lt;/span&gt;</pre>	
 <p>فاصله از بالا عکس : فاصله از چپ عکس :</p>	خروجی

## کاربرد jQuery HTML - ( ) prependTo

### prependTo ( ) jQuery :

( ) prependTo ، می تواند عنصر HTML تعیین شده برای آن را به ابتدای یک عنصر دیگر HTML اضافه نماید . برای مثال فرض کنید که یک تگ div و یک پاراگراف ( div ) دارید . می خواهید کدی طراحی نمایید تا در هنگام کلیک کاربر بر روی یک دکمه فرمان ، عنصر پاراگراف به ابتدای تگ div اضافه شود . برای این منظور باید از ( ) prependTo استفاده نمایید .

شکل کلی استفاده از متد ( ) به شرح زیر است :

Syntax	\$ ( content ).prependTo ( selector ) ;
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده عنصر HTML ای است که می خواهید آن را به ابتدای عنصر مورد نظر خود اضافه نمایید . ( این پارامتر باشد از نوع تگ های HTML ) . استفاده از این پارامتر اجباری است . <b>نکته مهم :</b> HTML ای که می خواهید آن را در ابتدای عنصر دیگر وارد نمایید ، از قبل در صفحه وجود داشته باشد ، در هنگام اضافه شدن از مکان خود به ابتدای عنصر مورد نظر منتقل می شود .	content
این پارامتر تعیین کننده id عنصری است که می خواهید عنصر مورد نظر خود را به ابتدای آن اضافه کنید . استفاده از این پارامتر اجباری است .	selector

: در مثال زیر یک پاراگراف و یک تگ div داریم . کدی طراحی کرده ایم که به واسطه کلیک کاربر بر روی دکمه فرمان مثال ، تگ پاراگراف به ابتدای تگ div اضافه می شود .

Example	
<pre>&lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.10.2/jquery.min.js"&gt; &lt;/script&gt;</pre>	کد

<pre> &lt;script type="text/javascript"&gt;   \$(document).ready(function () {     \$("button").click(function ()       {         \$("p").prependTo("div");       });   }); &lt;/script&gt; &lt;div&gt; This is our div element .&lt;/div&gt; &lt;p&gt; This is a paragraph.&lt;/p&gt; &lt;button&gt; برای جا به جایی عنصر پاراگراف کلیک نمایید &lt;/button&gt; </pre>	
<p>This is our div element .</p> <p>This is a paragraph.</p> <p>. کلیک نمایید برای جا به جایی عنصر پار</p>	<p>خروجی</p>

## کاربرد jQuery HTML - ( ) prop

متد ( ) prop : jQuery

( ) prop ، می تواند دو کار را انجام دهد :

- مقدار یک یا چند خاصیت یک عنصر HTML را خوانده و به صفحه بر می گرداند.
- می تواند مقدار یک یا چند خاصیت یک HTML را به مقدار تعیین شده و مورد نظر شما تغییر دهد.

برای مثال فرض کنید که یک عنصر پاراگراف در صفحه داریم . به وسیله این متد می توانید برای مثال رنگ متن پاراگراف یا همان خاصیت color را خوانده و یا اینکه به یک مقدار مورد نظر تغییر دهید .

**نکته :** هنگامی که متد ( ) prop را برای خواندن مقدار خاصیت یک عنصر HTML به کار می برید ، این متد مقدار خاصیت اولین عنصری که با مشخصات داده شده یکسان است را نمایش می دهد .  
ولی هنگامی که برای تغییر مقدار یک خاصیت عنصر HTML به کار می رود ، مقدار خاصیت کلیه عناصری که با مش داده شده مطابقت داشته و در صفحه وجود دارند ، را تغییر می دهد .

### ( ) prop : خواندن مقدار یک خاصیت عنصر HTML .

( ) prop گفتیم که این متد می تواند برای خواندن مقدار یک خاصیت یک عنصر HTML به کار

شکل کلی استفاده از متد ( ) prop در حالت اول به شرح زیر است :

Syntax	<code>\$( selector ).prop ( property ) ;</code>
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ای است که می خواهید مقدار یک خاصیت CSS آن را بخوانید . تعیین این پارامتر اجباری است .	selector
این پارامتر تعیین کننده نام خاصیتی است که می خواهید مقدار آن را بخوانید . تعیین این پارامتر اجباری است .	property

## ( ) prop : تغییر مقدار یک خاصیت عنصر HTML به مقدار دلخواه.

( ) prop گفتیم که این متد می تواند برای تغییر مقدار یک خاصیت یک عنصر

HTML به کار رود .

شکل کلی استفاده از متد ( ) prop در حالت دوم به شرح زیر است :

Syntax	\$ ( selector ).prop ( property , value ) ;
--------	---

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ای است که می خواهید مقدار یک خاصیت CSS را به یک مقدار جدید تغییر دهید . تعیین این پارامتر اجباری است .	selector
این پارامتر تعیین کننده نام خاصیتی است که می خواهید مقدار آن را به یک مقدار جدید تغییر دهید . تعیین این پارامتر اجباری است .	property
این پارامتر تعیین کننده مقدار جدیدی است که می خواهید برای خاصیت CSS HTML قرار دهید . تعیین این پارامتر اجباری است .	value

## ( ) prop : تغییر مقدار چند خاصیت یک عنصر HTML به مقادیر دلخواه.

( ) prop ، می توان مقدار چندین خاصیت CSS یک عنصر HTML را به مقادیر دلخواه تغییر

دهید . فرق این حالت با حالت دوم در این است که در حالت سوم ، شما بایستی نام خاصیت ها و مقدارهایشان را به صورت

جفت بنویسید . این مسئله در syntax

شکل کلی استفاده از م ( ) prop در حالت دوم به شرح زیر است :

Syntax	\$ ( selector ).prop ( { property 1 : value 1 , property 2 : value 2 , ... } ) ;
--------	--



## کاربرد jQuery HTML - remove ( )

### jQuery ( ) remove

`remove ( )` HTML تعیین شده برای آن را به همراه تمام متن و اشیای فرزندش از صفحه حذف می کند . این متد همچنین تمام داده ها و رویدادهای مرتبط با آن عنصر را نیز پاک می کند .  
برای مثال فرض کنید که یک پاراگراف در صفحه دارید . می خواهید کدی طراحی نمایید تا در صورت کلیک کاربر بر روی یک دکمه فرمان ، عنصر پاراگراف از روی صفحه حذف شود .

#### نکات :

**نکته :** برای حذف عنصر از صفحه بدون پاک شدن اطلاعات و رویدادهای مرتبط به آن از متد [detach \( \)](#) استفاده نمایید .

**نکته :** برای اینکه متن و یا محتویات یک عنصر را پاک کرده ، ولی عنصر همچنان بر روی صفحه وجود داشته باشد از [empty \( \)](#) استفاده نمایید .

شکل کلی استفاده از متد `remove ( )` به شرح زیر است :

Syntax	<code>\$( selector ).remove ( ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ای است که می خواهید آن را از صفحه حذف نمایید .	selector

Example	
<pre>&lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" /&gt; &lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.position.3/jquery.min.js"&gt; &lt;/script&gt;</pre>	کد

<pre> &lt;script type="text/javascript"&gt;   \$(document).ready(function () {     \$("button").click(function () {       \$("#ex").remove( );     });   }); &lt;/script&gt; &lt;p id="ex"&gt; This is a paragraph .you can remove it . &lt;/p&gt; &lt; button &gt; برای حذف پاراگراف کلیک نمایید &lt; /button &gt; </pre>	
<p>This is a paragraph .you can remove it .</p> <p>برای حذف پاراگراف کلیک نمایید</p>	<p>خروجی</p>

## کاربرد jQuery HTML - removeAttr ( )

### jQuery ( ) removeAttr

( ) removeAttr ، یک یا چندین خاصیت CSS تعیین شده برای عنصر HTML مورد نظرتان را حذف می کند . خاصیت [CSS](#) ، اثر آن بر عنصر از بین می .  
برای مثال فرض کنید که یک برای یک عنصر پاراگراف ( < p > ) در صفحه ، خاصیت رنگ نوشته یا color تعیین کرده اید . با حذف این خاصیت و مقدارش توسط متد ( ) removeAttr ، رنگ داده شده به آن از بین رفته و به رنگ پیش فرض بر می گردد .

شکل کلی استفاده از متد ( ) removeAttr به شرح زیر است :

Syntax	<code>\$( selector ).removeAttr ( attribute ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id HTML ای است که می خواهید یک یا چند خاصیت CSS آن را حذف نمایید . تعیین این پارامتر اجباری است .	selector
این پارامتر تعیین کننده خاصیت یا خواص CSS ای است که می خواهید آنها را حذف نمایید . تعیین این پارامتر اجباری است .	attribute

## کاربرد jQuery - HTML - replaceAll ( )

jQuery : prop ( )

replaceAll ( ) HTML تعیین شده برای آن را با تمام HTML مورد نظر شما در صفحه ، عوض می کند .

فرض کنید در یک صفحه چندین پاراگراف ( < p > ) دارید . می خواهید کدی طراحی نمایید تا در هنگام اجرای آن ، تمامی پاراگراف های صفحه یا مثلا پاراگراف های عضو یک کلاس خاص ، با یک تگ < span > در این ت باید از متد replaceAll ( ) استفاده نمایید .

Syntax	\$ ( content ).replaceAll ( selector ) ;
--------	--

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده عنصر HTML یا متنی است که می خواهید آن را به جای عنصر HTML مورد نظرتان در صفحه قرار دهید . تعیین این پارامتر اجباری است .	content
این پارامتر تعیین کننده id HTML ای است که می خواهید آن را با یک عنصر جدید مورد نظرتان عوض کنید . تعیین این پارامتر اجباری است .	selector

# کاربرد jQuery - HTML - text ( )

## jQuery ( ) text

text ( ) کار انجام می دهد :

- می تواند متن موجود در یک عنصر HTML مثل یک تگ را خوانده و به صفحه برگرداند.
- می تواند متن یک عنصر HTML مثل تگ پاراگراف را به متن دلخواه و تعیین شده تغییر دهد.

هنگامی که این متد برای بازگرداندن متن استفاده می شود متن تمام عناصری که با مشخصات داده شده مطابقت دارند را برمی گرداند ( این متد علامت های زبان نوشتن تگ های HTML را در نظر نمی گیرد ) .  
هنگامی که این متد برای تغییر متن عنصر به کار می رود متن تمام عناصری که با مشخصات داده شده مطابقت دارند را تغییر می دهد .

شکل کلی استفاده از متد text ( ) به شرح زیر است :

### ( خواندن و بازگرداندن متن یک عنصر HTML :

به وسیله متد text ( ) می توانید متن یک عنصر مثل تگ پاراگراف را خوانده و در خروجی نمایش دهید :

Syntax	<code>\$( selector ).text( ) ;</code>
--------	---------------------------------------

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم متن درون آن را خوانده و به خروجی بر گردانیم .	selector

: در مثال زیر ، از متد text ( ) برای خواندن و نمایش متن پاراگراف هایی که عضو کلاس هستند ، استفاده شده است .  
برای مشاهده خروجی مثال بر روی دکمه فرمان کلیک نمایید :

Example	
<code>&lt; html &gt;</code> <code>&lt; head &gt;</code>	کد

<pre> &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt; \$ (document). ready ( function( ) {     \$("#btn1").click(function(){         alert ( \$ ( "p.1" ).text( ) );     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;button id="btn1"&gt; Return the text content of all p elements&lt;/button&gt; &lt;p class="1"&gt; This is a paragraph.&lt;/p&gt; &lt;p class ="1"&gt; This is &lt;b&gt;another&lt;/b&gt; paragraph.&lt;/p&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>Return the text content of all p elements</p> <p>This is a paragraph.</p> <p>This is <b>another</b> paragraph.</p>	<p>خروجی</p>

( تغییر متن یک عنصر به متن دلخواه :

همچنین به وسیله این متد می توانید متن یک عنصر دلخواه مثل یک تگ پاراگراف را به متن دلخواه خود تغییر دهید:

Syntax	\$ ( selector ).text( content ) ;
--------	-----------------------------------

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم متن درون آن را به متن مورد نظرمان تغییر دهیم .	selector
این پارامتر تعیین کننده متنی است که می خواهیم ، متن عنصر مورد نظرمان را با آن جایگزین نماییم .	content

: در مثال زیر ، از متد ( ) text برای تغییر متن پاراگراف هایی که عضو کلاس هستند ، به یک مقدار جدید استفاده . برای مشاهده خروجی مثال بر روی دکمه فرمان کلیک نمایید :

Example	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$("#btn2").click(function(){         alert ( \$ ( "p.2" ).text( "hello World !" ) ) ;       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; button id="btn2" &gt; Return the text content of all p elements&lt; /button &gt;   &lt; p class="2" &gt; This is a paragraph.&lt; /p &gt;   &lt; p class ="2" &gt; This is &lt; b &gt;another&lt; /b &gt; paragraph.&lt; /p &gt; &lt; /body &gt; &lt; /html &gt; </pre>	کد
<p>change the paragraphs content .</p> <p>This is a paragraph.</p> <p>This is <b>another</b> paragraph.</p>	خروجی

## jQuery < کاربرد jQuery < HTML toggleClass ( )

### jQuery ( ) toggleClass

toggleClass ( ) ، می تواند به یک عنصر یک یا چند کلاس CSS را اضافه کرده و یا کلاس CSS آن را حذف نماید

خاصیت به کار بردن این متد این است که می توانید مجموعه خواص یک کلاس را به عنصر مورد نظرتان به یکباره اعمال کرده و یا اینکه مجموعه خواص را از حذف نمایید .

همچنین به وسیله پارامتر switch در این متد ، می توانید تعیین کنید که آیا کلاس فقط اضافه شده و یا فقط حذف شود . برای مشاهده نحوه کار افکت ( ) toggleClass به مثال عملی دقت نمایید . هنگامی که کلاسی را به یک عنصر اضافه می کنید ، آن عنصر به یکباره تمام خواص آن را به ارث برده و در خود اعمال می کند . شکل کلی استفاده از این متد به شرح زیر است :

Syntax	<code>\$( selector ).toggleClass( classname , function ( ) , switch ) ;</code>
--------	--

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم کلاس مورد نظر را به آن اضافه کرده و یا از آن حذف نماییم . استفاده از این پارامتر اجباری است .	selector
این پارامتر تعیین کننده یک یا چند کلاس است که می خواهید آن را به عنصر مورد نظرتان اضافه کرده و یا از آن حذف نمایید . برای تعیین بیش از یک نام کلاس ، نام کلاس ها را با فاصله از هم جدا نمایید . استفاده از این پارامتر اجباری است .	classname
این پارامتر تعیین کننده نام تابعی است که نام یک یا چند کلاس را برای استفاده به متد بر می گرداند .	function
این پارامتر یک مقدار Boolean را تعیین می کند که آیا کلاس تعیین شده باید به عنصر اضافه شود ( true ) و یا از ( false ) .	switch



: در مثال زیر چند پاراگراف داریم . برای آنها یک رویداد ( ) toggleClass نوشته ایم ، با کلیک کاربر بر روی دکمه فرمان رویداد ( ) toggleClass اجرا شده و کلاس تگ های p را به کلاس main تغییر می دهد :

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $(document).ready(function () {
      $("#btn1").click(function(){
        $("p").toggleClass("main");
      });
    });
  </script>
  <style type="text/css">
    .main
    {
      font-size: 120%;
      color: red;
    }
  </style>
</head>
<body>
  <button id="btn1"> Toggle class "main" for p elements</button>
  <p> This is a paragraph.</p>
  <p> This is another paragraph.</p>
  <p> <b>Note:</b> Click the button more than once to see the toggle
effect.</p>
</body>
</html>
```

کد

## jQuery < کاربرد jQuery HTML < ( ) val

### jQuery ( ) val

jQuery ( ) val می تواند مقدار خاصیت value یک عنصر را خوانده و یا به یک مقدار دلخواه تنظیم نماید . در هنگام بازگرداندن مقدار یک عنصر ، این متد مقدار value اولین عنصر مطابق با مشخصات داده شده در آن را بر می

اما در تنظیم کردن مقدار value برای عناصر به مقدار دلخواه ، این متد مقدار خاصیت value تمام عناصری که با مشخصات داده شده مطابقت دارند را تنظیم می کند .

**نکته :** این متد معمولاً با المنت ها و کنترل های فرم های HTML کار می کند .

### کار با متد ( ) val :

باز گرداندن مقدار خاصیت value یک عنصر :

Syntax 1	<code>\$(selector).val( ) ;</code>
----------	------------------------------------

تنظیم مقدار خاصیت value یک عنصر به مقدار دلخواه :

Syntax 2	<code>\$(selector).val( value ) ;</code>
----------	--

تنظیم مقدار خاصیت value یک عنصر به مقدار دلخواه توسط یک تابع :

Syntax 3	<code>\$(selector).val( function ) ;</code>
----------	---

**:** خواندن مقدار خاصیت value یک عنصر:

Example	
<pre>&lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" height="70px" /&gt; &lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.9.1/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;</pre>	کد

<pre> \$(document).ready(function () {     \$("#btn1").click(function ()         { alert ( \$ ( #fname" ).val ( ));     }); }); &lt;/script&gt; </pre> <p> <b>firstname:</b> <input id="fname" type="text" value="Ali"/>   <input type="button" value="Value"/> </p>	
<p> <b>firstname:</b> <input type="text" value="Ali"/>   <input type="button" value="Value"/> </p>	<p>خروجی</p>

: تنظیم مقدار خاصیت value یک عنصر به مقدار دلخواه:

Example	
<pre> &lt; html &gt; &lt; head &gt; &lt;script type="text/javascript" src="jquery.js" &gt; &lt;/script &gt; &lt;script type="text/javascript" &gt;     \$(document).ready(function () {         \$("#btn2").click(function(){             \$("#name").val("Developer 1");         });     }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     <b>firstname:</b> <input &gt;="" <input="" id="btn2" type="button" value="تنظیم مقدار Value"/> &lt;/body&gt; &lt;/html&gt; </pre>	<p>کد</p>
<p> <b>firstname:</b> <input type="text"/> <input type="button" value="تنظیم مقدار Value"/> </p>	<p>خروجی</p>

## jQuery < کاربرد jQuery HTML > wrap ( )

### jQuery ( ) wrap

wrap ( ) jQuery , المنت های تعیین شده برای آن را به دور عنصر یا عناصر مورد نظر قرار می دهد .  
برای مثال فرض کنید ، پاراگراف یا تگ < p > داریم . به وسیله این متد می توان مثلا یک تگ < div > را به دور ، آنها  
قرار دادن این تگ div دور پاراگراف ها ، باعث می شود تا خواص تگ div بر روی پاراگراف ها اثر ب .  
شکل کلی استفاده از این متد به صورت زیر است :

Syntax 1	<code>\$(selector).wrap( wrappingElement ,function( ) ) ;</code>
----------	--

توضیح موارد syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم آن را به دور سایر عناصر قرار دهیم . استفاده از این پارامتر اجباری است . انواع مقادیر ممکن عبارتند از : <ul style="list-style-type: none"><li>HTML elements - المنت های HTML</li><li>jQuery objects - اشیای jQuery</li><li>DOM elements - المنت های DOM</li></ul>	wrappingElement
این پارامتر تعیین کننده یک تابع است که می تواند عنصر مورد نظر را برای انجام عملیات wrapping استفاده از این پارامتر اختیاری است .	function

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ ( document ). ready ( function ( ) {</pre>	کد

<pre> \$("#btn1").click(function(){     \$("p").wrap("&lt;div&gt; &lt;/div&gt;"); }); }); &lt;/script&gt; &lt;style type="text/css"&gt; div{ background-color:yellow; } &lt;/style&gt; &lt;/head&gt; &lt;body&gt; &lt;p&gt; This is a paragraph.&lt;/p&gt; &lt;p&gt; This is another paragraph.&lt;/p&gt; &lt;button id="btn1"&gt; Wrap a div element around each p element&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>This is a paragraph.</p> <p>This is another paragraph.</p> <p>Wrap a div element around each p element</p>	<p>خروجی</p>

## کاربرد jQuery HTML - width ( )

### متد width ( ) jQuery

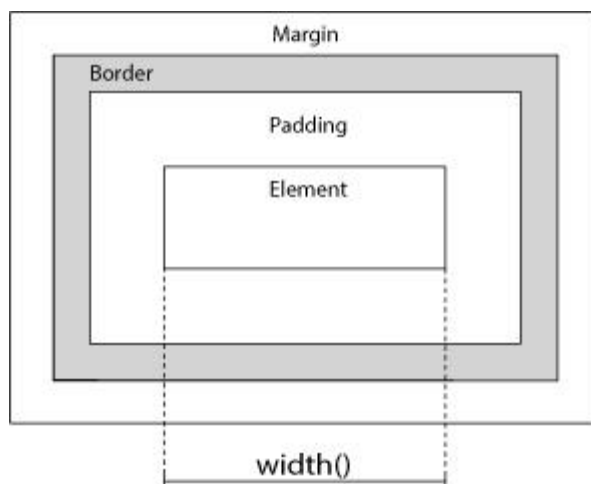
jQuery width ( ) می تواند مقدار عرض یک عنصر HTML را خوانده و بر گرداند و یا به وسیله آن مقدار عرض عنصر را به یک مقدار دلخواه تغییر دهید .

برای مثال اگر یک عنصر div داشته باشیم که عرض آن 300 px width ( ) می توان دو کار را :

مقدار عرض آن را که 300 px است ، به صفحه برگدانیم .

مقدار جدیدی را برای عرض عنصر تعیین کرده و آن را به مقدار جدید تغییر دهید .

**نکته :** همانطور که در تصویر زیر نمایش داده ایم ، متد width ( ) شامل حاشیه بیرونی ( margin ) ، حاشیه درونی ( padding ) و خطوط حاشیه عنصر ( border ) نمی شود و فقط عرض خود عنصر را بر می گرداند .



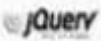
### width ( ) - خواندن عرض یک عنصر :

گفتیم که در حالت اول متد width ( ) ، از این متد برای خواندن مقدار عرض یک عنصر و بازگرداندن آن مقدار به صفحه استفاده می شود . شکل کلی استفاده از متد width ( ) در حالت اول به شرح زیر است :

Syntax	<code>\$ ( " selector " ).width ( ) ;</code>
--------	--

در مثال زیر یک تصویر در مثال داریم . width ، عرض این عنصر را خوانده و در صفحه نمایش

داده ایم :

Example	
<pre>&lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" width="70px" /&gt; &lt;script type="text/javascript" src="//ajax.googleapis.com/ajax/libs/jquery/1.width.3/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;     \$(document).ready(function () {         document.getElementById("EX").innerHTML = "      : " + \$("#Pic_1").width();     }); &lt;/script&gt; &lt;p id="EX"&gt;      &lt;/p&gt;</pre>	کد
 :	خروجی

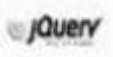
### width ( ) - تنظیم عرض یک عنصر :

گفتیم که در حالت دوم متد width ( ) بین متد برای تنظیم مقدار عرض یک عنصر به یک مقدار جدید استفاده می شود . شکل کلی استفاده از متد width ( ) در حالت دوم به شرح زیر است :

Syntax	<pre>\$( " selector " ).width ( value ) ; * value :</pre>
--------	---

: در مثال زیر یک تصویر در مثال داریم . width ، عرض این عنصر را به یک مقدار جدید تغییر داده ایم :

Example	
<pre>&lt;img src="../../Pic/Icons/JQ1.jpg" id="Img1" width="70px" /&gt; &lt;script type="text/javascript"</pre>	کد

<pre>src="//ajax.googleapis.com/ajax/libs/jquery/1.7.2/jquery.min.js"&gt; &lt;/script&gt;  &lt;script type="text/javascript"&gt;     \$(document).ready(function () {         \$("#Pic_2").width( "200px");     }); &lt;/script&gt;</pre>	
	خروجی



## jQuery < کاربرد jQuery Ajax < لیست متدها

### لیست متدها کاربرد jQuery Ajax

همانطور که در [Ajax](#) اشاره کردیم ، Ajax تکنولوژی تبادل اطلاعات و به روز رسانی صفحات وب ، بدون لود شدن مجدد و کامل صفحه است .

در لیست زیر ، متدهای کار با Ajax با jQuery ر داده شده است ، برای دریافت اطلاعات راجع به هر کدام بر روی نام هر متد کلیک نمایید :

<a href="#">\$.ajax ( )</a>	این متد یک درخواست Ajax را به سرور ارسال می کند .
<a href="#">ajaxComplete ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید ، تا در هنگام اتمام درخواست Ajax .
<a href="#">ajaxError ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید ، تا در هنگام اتمام درخواست Ajax error .
<a href="#">ajaxSend ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید تا Ajax به سرور ، اجرا شود .
<a href="#">ajaxSetup ( )</a>	به وسیله این متد می توانید مقادیر پیش فرضی را برای درخواست های بعدی Ajax تعیین نمایید .
<a href="#">ajaxStart ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید تا همزمان با اولین درخواست Ajax .
<a href="#">ajaxStop ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید ، تا در هنگام پایان یافتن تمامی درخواست های Ajax .
<a href="#">ajaxSuccess ( )</a>	به وسیله این متد می توانید یک تابع را تعیین نمایید تا پس از پایان موفقیت آمیز تمامی درخواست های Ajax به .
<a href="#">\$.get ( )</a>	این متد اطلاعات را با استفاده از یک درخواست Ajax HTTP GET ، از سرور می خواند .
<a href="#">\$.getScript ( )</a>	این متد یک اسکریپت جاوا اسکریپت را به وسیله یک درخواست Ajax HTTP GET

	اجرا می کند .
<a href="#">load ( )</a>	این متد اطلاعات را از سرور خوانده و آن را در عنصر مورد نظر اعمال می کند .
<a href="#">\$.param ( )</a>	این متد لیست اعضای یک آرایه یا مجموعه خواص یک شی را به صورت پشت سر هم و جفت نام-بر می گرداند .
<a href="#">\$.post ( )</a>	این متد اطلاعات را از سرور به وسیله یک متد Ajax HTTP POST ، دریافت می کند .
<a href="#">serialize ( )</a>	این متد می تواند اطلاعات یک فرم را به صورت پشت سر هم و جفت نام-مقدار خوانده و به انتهای آدرس صفحه ، جهت ارسال اضافه نماید .
<a href="#">serializeArray ( )</a>	این متد می تواند اطلاعات یک فرم را به صورت آرایه درآورده و ارسال نماید .

## jQuery < کاربرد jQuery Ajax < Ajax ( )

### متد Ajax ( ) jQuery

( Ajax ) ، برای ارسال یک درخواست Ajax به سرور استفاده می شود .

تمامی متدهای Ajax jQuery ( Ajax ) برای ارسال درخواست خود استفاده می کنند . اما در زمانی که هیچ کدام از آنها را نتوان به کار برد ، باید از متد ( Ajax ) استفاده کرد .

شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$.ajax( { name:value , name:value , ... } );</code>
--------	--

مقادیر پیش فرض را می توانید به صورت جفت های ن - مقدار تعیین نمایید .

در جدول زیر ، انواع مقادیر ممکن برای جفت های نام-مقدار را معرفی کرده ایم :

انواع جفت های نام-	
توضیح	-
این پارامتر یک مقدار Boolean را تعیین می کند که آیا درخواست به صورت تبادل اطلاعاتی به سرور ارسال شود یا خیر . مقدار پیش فرض true .	async
به وسیله این پارامتر می توان یک تابع را تعیین کرد تا قبل از ارسال درخواست به سرور اجرا .	beforeSend ( xhr )
این پارامتر یک مقدار Boolean را تعیین می کند ، که آیا صفحات دریافتی از سرور باید در حافظه Cashe ذخیره شوند یا خیر . مقدار پیش فرض true .	cashe
این پارامتر یک تابع را تعیین می کند تا در هنگام پایان یافتن درخواست اجرا شود . مقدار پیش فرض true .	complete ( xhr , status )
تعیین کننده اطلاعاتی است که می خواهید به سرور ارسال نمایید .	data
این پارامتر تعیین کننده یک تابع است تا در صورتی که درخواست ناموفق بود ، اجرا شود .	error
این پارامتر یک رمز عبور را برای استفاده در درخواست HTTP تعیین می کند .	password

مقدار پیش فرض true .	
این پارامتر یک مقدار Boolean را تعیین می کند که آیا اطلاعاتی که قرار است به سد شوند ، باید به صورت QueryString باشند یا خیر . مقدار پیش فرض true .	processData
این پارامتر یک تابع را تعیین می کند تا در صورت اجرای موفق درخواست اجرا شود .	success
مدت زمانی است که برای اجرای درخواست به میلی ثانیه تعیین می شود .	timeout
این پارامتر نوع درخواست را تعیین می کند ( GET یا POST ) .	type
این پارامتر آدرس صفحه مقصد که باید درخواست به آن ارسال شود را تعیین می کند . مقدار پیش فرض صفحه جاری است .	url
این پارامتر یک نام کاربری را برای استفاده در درخواست HTTP تعیین می کند .	username

## jQuery < کاربرد jQuery Ajax < ajaxComplete ( )

### ajaxComplete ( ) jQuery

ajaxComplete ( ) ، یک تابع را تعیین می کند تا پس از اتمام درخواست Ajax .  
ajaxSuccess ( ) ، این متد تابع تعیین شده برای آن را اجرا می کند ، حتی اگر در خواست Ajax با موفقیت

شکل کلی استفاده از این متد بصورت زیر است :

Syntax	\$( selector ).ajaxComplete( function( event , xhr , options ) ) ;
--------	--

توضیح موارد syntax	
توضیح	Parameter
نام تابعی است که می خواهیم پس از اتمام درخواست های Ajax استفاده از این پارامتر اجباری است . <ul style="list-style-type: none"><li>• <b>event</b> : در بر گیرنده شی event</li><li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XmlHttpRequest</a></li><li>• <b>options</b> : در بر گیرنده تنظیمات ( options ) Ajax</li></ul>	function

: در مثال زیر وقتی که عملیات Ajax شروع می شود ، متد [ajaxStart \( \)](#) یک عکس را بر روی صفحه نمایش می دهد

سپس هنگامی که عملیات Ajax ajaxComplete ( ) ، عکس را از صفحه مخفی می کند . به کد دقت نمایید :

Example	
< html > < head > < script type="text/javascript" src="jquery.js" > < /script > < script type="text/javascript" >	کد

<pre> \$ (document). ready ( function( ) { 2   \$("#txt").ajaxStart(function(){         \$("#wait").css("display","block");     }); 3   \$("#txt").ajaxComplete(function(){         \$("#wait").css("display","none");     }); 1   \$("#btn1").click(function(){         \$("#txt").load("demo_ajax_load.txt");     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt; &lt;div id="txt"&gt;     &lt;h2&gt;         Let AJAX change this text     &lt;/h2&gt; &lt;/div&gt; &lt;button id="btn1"&gt; Change Content &lt;/button&gt; &lt;div id="wait" style="display: none; width: 69px; height: 89px; border: 1px solid black; position: absolute; top: 50%; left: 50%; padding: 2px;"&gt;     &lt;img src='demo_wait.gif' width="64" height="64" /&gt;     Loading.. &lt;/div&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p><b>Let AJAX change this text</b></p> <p>Change Content</p>	<p>خروجی</p>

## jQuery < کاربرد jQuery Ajax < ajaxError ( )

### jQuery ( ) ajaxError

( ajaxError ) ، یک تابع را تعیین می کند تا در صورتی که در خواست Ajax error مواجه شد ، اجرا شود . برای مثال به وسیله این متد می توان پیام هشدار ی را به کاربر مبنی بر انجام ناموفقیت آمیز درخواست Ajax ، به کار برد .  
شکل کلی استفاده از این متد بصورت زیر است :

Syntax	\$( selector ).ajaxError( function( event , xhr , options ) ) ;
--------	---

توضیح موارد syntax	
توضیح	Parameter
نام تابعی است که می خواهیم در صورت بروز error در اجرای درخواست های Ajax فاده از این پارامتر اجباری است . <ul style="list-style-type: none"><li>• <b>event</b> : در بر گیرنده شی event .</li><li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XmlHttpRequest</a> .</li><li>• <b>options</b> : در برگیرنده تنظیمات ( options ) Ajax .</li></ul>	function

: در مثال زیر در هنگام اجرای درخواست Ajax ، به دلیل اینکه برنامه نمی تواند فایل مقصد را پیدا نماید ، یک error رخ داده و پیام هشدار صادر می شود :

Example	
< html > < head > < script type="text/javascript" src="jquery.js" > < /script > < script type="text/javascript" >	کد

<pre> \$ (document). ready ( function ( ) {     \$("div").ajaxError(function () {         alert("An error occured!");     });     \$("#btn1").click(function () {         \$("div").load("wrongfile.txt");     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;div&gt;         &lt;h2&gt; Let AJAX change this text &lt;/h2&gt;     &lt;/div&gt;     &lt;button id="Button1" type="button"&gt; Change Content&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p><b>Let AJAX change this text</b></p> <p>Change Content</p>	<p>خروجی</p>



## jQuery < کاربرد jQuery Ajax < ajaxSend ( )

### jQuery ( ) ajaxSend

به وسیله متد ( ) ajaxSend می توان یک تابع را تعیین کرد تا در زمانی که درخواست Ajax به سرور ارسال می شود ،

اقع از این متد می توان برای تعیین یک دستور یا کد ، برای اجرا همزمان با ارسال درخواست Ajax به سرور استفاده

شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$( selector ).ajaxSend ( function( event , xhr , options ) ) ;</code>
--------	--

#### توضیح موارد syntax

توضیح	Parameter
نام تابعی است که می خواهیم همزمان با ارسال درخواست های Ajax به سرور، اجرا شود . استفاده از این پارامتر اجباری است .	function
<ul style="list-style-type: none"><li>• <b>event</b> : در بر گیرنده شی event .</li><li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XmlHttpRequest</a> .</li><li>• <b>options</b> : در برگیرنده تنظیمات ( options ) Ajax .</li></ul>	

#### Example

<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt;     \$( document). ready ( function( ) {         \$("div").ajaxSend(function(e,xhr,opt){</pre>	کد
---	----

<pre> \$(this).html("Requesting " + opt.url); }); \$("#btn1").click(function(){     \$("#div").load("demo_ajax_load.asp"); }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;div&gt;         &lt;h2&gt; Let AJAX change this text &lt;/h2&gt;     &lt;/div&gt;     &lt;button id="btn1" type="button"&gt; Change Content&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>Let AJAX change this text</p> <p>Change Content</p>	<p>خروجی</p>

## jQuery < کاربرد jQuery Ajax < ajaxStart ( )

### ajaxStart ( ) jQuery

به وسیله متد ( ) ajaxStart ، می توان یک تابع را تعیین نمود تا در زمانی که در خواست Ajax برای ارسال به سرور شروع می شود ، اجرا شد .

می توان کدها یا دستوراتی که می خواهید به محض شروع شدن درخواست Ajax اجرا شوند ، را در این متد تعیین نمایید . به مثال عملی در انتهای صفحه دقت نمایید .

شکل کلی استفاده از این متد بصورت زیر است :

Syntax	\$( selector ).ajaxStart( function ( ) ) ;
--------	--

توضیح موارد syntax	
توضیح	Parameter
نام تابعی است که می خواهیم به محض شروع شدن درخواست های Ajax استفاده از این پارامتر اجباری است .	function

**مثال عملی :** در مثال زیر هنگامی که درخواست Ajax در رویداد load شروع به اجرا می کند ، همزمان با آن یک دستور ajaxStart ( ) ، تصویر را در خروجی مثال نمایش می دهد :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$("#div1").ajaxStart(function(){         \$(this).html("&lt;img src='demo_wait.gif' /&gt;");       });       \$("#btn1").click(function(){         \$("#div1").load("demo_ajax_load.asp");       });     });   &lt; /script &gt; &lt; /html &gt;</pre>	کد

<pre> }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;div&gt;     &lt;h2&gt; Let AJAX change this text &lt;/h2&gt;   &lt;/div&gt;   &lt;button id="btn1" type="button"&gt; Change Content&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>Let AJAX change this text</p> <p>Change Content</p>	<p>خروجی</p>

## jQuery < کاربرد jQuery Ajax < ajaxStop ( )

### ajaxStop ( ) jQuery

به وسیله متد ( ) ajaxStop می توان یک تابع را تعیین کرد تا در زمان پایان یافتن ارسال و انجام تمام درخواست های Ajax به سرور ، اجرا شود .

در واقع از این متد می توان برای تعیین یک دستور یا کد ، برای اجرا پس از پایان یافتن انجام درخواست های Ajax

هنگامی که یک درخواست ajax تمام می شود ، jQuery چک می کند که آیا درخواست دیگری وجود دارد یا خیر .

ajax دیگری وجود نداشته باشد ، آنگاه ajax تابع این متد را اجرا می کند .

شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$( selector ).ajaxStop ( function ( ) ) ;</code>
--------	---

توضیح موارد syntax	
توضیح	Parameter
نام تابعی است که می خواهیم پس از پایان یافتن تمام درخواست های Ajax استفاده از این پارامتر اجباری اس .	function

**مثال عملی :** در مثال زیر یک درخواست ajax `load ( )` انجام می شود . پس از پایان یافتن آن ، برنامه یک پیام هشدار را به وسیله متد ( ) ajaxStop به کاربر ارسال می کند :

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {         \$("div").ajaxStop(function){</pre>	کد

<pre>                 alert("All AJAX requests completed");             });             \$("#btn1").click(function(){                 \$("#div").load("demo_ajax_load.txt");             });         });     &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;div&gt;         &lt;h2&gt; Let AJAX change this text &lt;/h2&gt;     &lt;/div&gt;     &lt;button id="btn1" type="button"&gt; Change Content&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>Let AJAX change this text</p> <p>Change Content</p>	<p>خروجی</p>

## jQuery < کاربرد jQuery Ajax < ajaxSuccess ( )

### jQuery ( ) ajaxSuccess

به وسیله متد ( ) ajaxSuccess می توان یک تابع را تعیین کرد تا در زمانی که درخواست های Ajax به سرور ، به صورت کامل و موفقیت آمیز به پایان رسید، اجرا شود .  
در واقع از این متد می توان برای قرار دادن کدها و دستوراتی که می خواهید پس از پایان یافتن موفقیت آمیز درخواست های Ajax شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$( selector ).ajaxSuccess ( function ( event , xhr , options ) ) ;</code>
--------	--

توضیح موارد syntax	
توضیح	Parameter
نام تابعی است که می خواهیم پس از اتمام موفقیت آمیز درخواست های Ajax استفاده از این پارامتر اجباری است .	function
<ul style="list-style-type: none"><li>• <b>event</b> : در بر گیرنده شی event</li><li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XmlHttpRequest</a></li><li>• <b>options</b> : در برگیرنده تنظیمات ( options ) Ajax</li></ul>	

**مثال عملی :** در مثال زیر یک درخواست ajax `load ( )` انجام می شود . پس از پایان یافتن موفقیت آمیز آن ، برنامه یک پیام هشدار را به وسیله متد ( ) ajaxSuccess به کاربر ارسال می کند :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;</pre>	کد

<pre> \$ (document). ready ( function( ) {     \$("div").ajaxSuccess(function(){         alert("All AJAX requests completed");     });     \$("#btn1").click(function(){         \$("div").load("demo_ajax_load.txt");     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;div&gt;         &lt;h2&gt; Let AJAX change this text &lt;/h2&gt;     &lt;/div&gt;     &lt;button id="btn1" type="button"&gt; Change Content&lt;/button&gt; &lt;/body&gt; &lt;/html&gt; </pre>	
<p>Let AJAX change this text</p> <p>Change Content</p>	<p>خروجی</p>



## jQuery < کاربرد jQuery Ajax < \$.get ( )

### \$.get ( ) jQuery

\$.get ( ) ، با استفاده از یک درخواست HTTP GET ، اطلاعات مورد نظر را از سرور می خواند .  
حالت های مختلفی برای استفاده از این متد وجود دارد که به تشریح هر یک می پردازیم :  
. خواندن اطلاعات از یک فایل " test.php " ، بدون در نظر گرفتن خروجی آن :

```
$.get( " test.php " ) ;
```

. خواندن اطلاعات از یک فایل " test.php " و ارسال همزمان اطلاعات اضافه به همراه درخواست ، بدون در نظر گرفتن خروجی آن :

```
$.get( " test.php " , { name : " Ali " , town : " 1jQuery " } ) ;
```

. خواندن اطلاعات از فایل " test.php " و ارسال همزمان آرایه ای از داده ها به همراه درخواست ، بدون در نظر گرفتن خروجی آن :

```
$.get( " test.php " , { 'colors[]' : [ "Red" , "Green" , "Blue" ] } ) ;
```

. خواندن اطلاعات از فایل " test.php " و نمایش اطلاعات در خروجی :

```
$.get("test.php", function(data){  
    alert("Data: " + data);  
});
```

شکل کلی استفاده از متد \$.get ( ) به صورت زیر است :

Syntax

```
$.get( URL , data , function ( data , status , xhr ) , dataType ) ;
```

توضیح موارد syntax

توضیح	Parameter
تعیین کننده آدرس فایلی است که می خواهید متد آن را از سرور درخواست کرده و بخواند . استفاده از این پارامتر اجباری است .	url
این پارامتر تعیین کننده اطلاعاتی است که می خوئیم همراه درخواست خود به سرور ارسال نماییم. استفاده از این پارامتر اختیاری است .	data
این پارامتر تعیین کننده تابعی است که می خواهید هنگ این تابع خود می تواند چندین پارامتر نیز داشته باشد : استفاده از این پارامتر اختیاری است .  <ul style="list-style-type: none"> <li>• <b>data</b> : در بر گیرنده اطلاعات خروجی از درخواست است .</li> <li>• <b>status</b> : این پارامتر تعیین کننده وضعیت درخواست است که می تواند چندین حالت داشته :  <ul style="list-style-type: none"> <li>○ <b>success</b> : .</li> <li>○ <b>notmodified</b> : هنوز اعمال یا بررسی نشده.</li> <li>○ <b>error</b> : اشکال در ارسال یا پردازش درخواست .</li> <li>○ <b>timeout</b> : مدت زمان کافی برای اجرای درخواست تمام شده است .</li> <li>○ <b>parseerror</b> : خواندن و اجرای درخواست با مشکل مواجه شده است .</li> </ul> </li> <li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XMLHttpRequest</a> .</li> </ul>	function
این پارامتر نوع داده ارسالی از طرف سرور را تعیین می کند . به صورت اتوماتیک jQuery به صورت پیش فرض نوع این داده را حدس می زند و می تواند یکی از موارد زیر باشد : استفاده از این پارامتر اختیاری است .  <ul style="list-style-type: none"> <li>• <b>xml</b> : یک فایل XML.</li> <li>• <b>html</b> : فایل HTML .</li> <li>• <b>text</b> : یک متن ساده String .</li> <li>• <b>Script</b> : درخواست را به صورت یک اسکریپت اجرا کرده و نتیجه را به صورت متن ساده بر می گرداند .</li> </ul>	dataType

## jQuery < کاربرد jQuery Ajax < ( ) ajaxScript

### jQuery ( ) ajaxScript

به وسیله متد ( ) ajaxScript ، می توان یک اسکریپت را با استفاده از روش HTTP GET

سپس از نتیجه حاصل در صفحه استفاده کرد .  
شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$(selector).getScript( url , success ( response , status ) ) ;</code>
--------	--

توضیح موارد syntax	
توضیح	Parameter
تعیین کننده آدرس فایل اسکریپتی است که می خواهید متد آن را از سرور درخواست کرده و بخواند . استفاده از این پارامتر اجباری است .	url
این پارامتر می تواند یک تابع را تعیین نماید تا در صورتی که اجرای درخواست و اسکریپت موفقیت آمیز بود ، اجرا شود . استفاده از این پارامتر اختیاری است . این تابع خود می تواند دارای پارامتر نیز باشد :	success
<ul style="list-style-type: none"><li>• <b>response</b> : شامل جواب برگشتی از سرور است .</li><li>• <b>status</b> : این پارامتر تعیین کننده وضعیت درخواست است که می تواند چندین حالت داشته باشد :</li></ul> <ul style="list-style-type: none"><li>○ <b>success</b> : .</li><li>○ <b>notmodified</b> : هنوز اعمال یا بررسی نشده.</li><li>○ <b>error</b> : اشکال در ارسال یا پردازش درخواست .</li><li>○ <b>timeout</b> : مدت زمان کافی برای اجرای درخواست تمام شده است .</li><li>○ <b>parseerror</b> : رای درخواست با مشکل مواجه شده است .</li></ul>	

:

### Example

```
< html >
< head >
  < script type="text/javascript" src="jquery.js" > < /script >
  < script type="text/javascript" >
    $ (document). ready ( function ( ) {
      $("#btn1").click(function(){
        $.getScript("demo_ajax_script.js");
      });
    });
  < /script >
< /head >
< body >
  < button id="btn1" > خواندن اسکرپت < /button >
< /body >
< /html >
```

کد

خواندن اسکرپت

خروجی

# load ( ) < Ajax jQuery < jQuery

## jQuery ( ) load

**load ( )** ( از یک فایل یا اسکریپت ) خوانده و در عنصر مورد نظر ، بر روی صفحه قرار می دهد ( اطلاعات آن را به روز رسانی می کند ) .  
برای مثال فرض کنید که متنی را در یک فایل بر روی سرور ذخیره کرده اید ، می توانید با قرار دادن یک دکمه فرمان و تعیین **load ( )** در کد آن ، کاری کنید تا در صورت کلیک کاربر بر روی دکمه فرمان ، متن ذخیره شده در فایل ، مثلا در یک **div** بر روی صفحه قرار داده شود .  
شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$(selector).load ( url , data , function( response , status , xhr ) ) ;</code>
--------	---

توضیح موارد syntax	
توضیح	Parameter
تعیین کننده آدرس فایلی است که می خواهیم اطلاعات را از روی آن بخوانیم . استفاده از این پارامتر اجباری است .	url
این پارامتر تعیین کننده اطلاعاتی است که می خواهیم همراه درخواست خود به سرور ارسال نماییم. استفاده از این پارامتر اختیاری است .	data
این پارامتر تعیین کننده تابعی است که می خواهید هنگامی که متد <b>load ( )</b> به پایان رسید ، اجرا . این تابع خود می تواند چندین پارامتر نیز داشته باشد : استفاده از این پارامتر اختیاری است . <ul style="list-style-type: none"><li><b>response</b> : در بر گیرنده اطلاعات خروجی از درخواست است .</li><li><b>status</b> : این پارامتر تعیین کننده وضعیت درخواست است که می تواند چندین حالت داشته باشد :<ul style="list-style-type: none"><li><b>success</b> : .</li><li><b>notmodified</b> : هنوز اعمال یا بررسی نشد .</li></ul></li></ul>	function

- |   |  |
|---|--|
| <ul style="list-style-type: none"><li>○ error : اشکال در ارسال یا پردازش درخواست .</li><li>○ timeout : مدت زمان کافی برای اجرای درخواست تمام شده است .</li><li>○ parseerror : خواندن و اجرای درخواست با مشکل مواجه شده است .</li><li>● xhr : در بر گیرنده شی <a href="#">XmlHttpRequest</a> .</li></ul> |  |
|---|--|

## jQuery < کاربرد jQuery Ajax < post ( )

### متد post ( ) jQuery

- وسیله متد ( ) post ، می توان فایلی را از سرور با استفاده از یک درخواست HTTP POST . نتیجه خروجی بر روی صفحه استفاده نمود . شکل کلی استفاده از این متد بصورت زیر است :

Syntax	<code>\$(selector).post( URL , data , function ( data , status , xhr ) , dataType ) ;</code>
--------	--

توضیح موارد syntax	
توضیح	Parameter
تعیین کننده آدرس فایلی است که می خواهیم اطلاعات را از سرور درخواست کرده و بخوانیم . استفاده از این پارامتر اجباری است .	URL
این پارامتر تعیین کننده اطلاعاتی است که می خواهیم همراه درخواست خود به سرور ارسال نماییم. استفاده از این پارامتر اختیاری است .	data
این پارامتر تعیین کننده تابعی است که می خواهید هنگامی که متد ( ) post به پایان رسید ، اجرا شود . این تابع خود می تواند چندین پارامتر نیز داشته باشد : استفاده از این پارامتر اختیاری است . <ul style="list-style-type: none"><li>• <b>data</b> : در بر گیرنده اطلاعات خروجی از درخواست است .</li><li>• <b>status</b> : این پارامتر تعیین کننده وضعیت درخواست است که می تواند چندین حالت داشته باشد :<ul style="list-style-type: none"><li>○ <b>success</b> : .</li><li>○ <b>notmodified</b> : هنوز اعمال یا بررسی نشده.</li><li>○ <b>error</b> : اشکال در ارسال یا پردازش درخواست .</li><li>○ <b>timeout</b> : مدت زمان کافی برای اجرای درخواست تمام شده است .</li><li>○ <b>parseerror</b> : خواندن و اجرای درخواست با مشکل مواجه شده است .</li></ul></li></ul>	function

<ul style="list-style-type: none"> <li>• <b>xhr</b> : در بر گیرنده شی <a href="#">XmlHttpRequest</a> .</li> </ul>	
<p>این پارامتر نوع داده ارسالی از طرف سرور را تعیین می کند . به صورت اتوماتیک jQuery به صورت پیش فرض نوع این داده را حدس می زند می تواند یکی از موارد زیر باشد : استفاده از این پارامتر اختیاری است .</p> <ul style="list-style-type: none"> <li>• <b>xml</b> : یک فایل XML .</li> <li>• <b>html</b> : فایل HTML .</li> <li>• <b>text</b> : یک متن ساده String .</li> <li>• <b>Script</b> : درخواست را به صورت یک اسکریپت اجرا کرده و نتیجه را به صورت متن ساده بر می گرداند .</li> </ul>	<b>dataType</b>
<b>Example</b>	
<pre> &lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$("#btn1").click(function(){         \$.post("demo_ajax_post.txt",           {             name: "Ali",             city: "Tehran"           },           function (data, status) {             alert("Data: " + data + "\nStatus: " + status);           });       });     }   &lt; /script &gt; &lt; /head &gt; &lt; body &gt; </pre>	<b>کد</b>



<pre>&lt;button id="btn1"&gt; خواندن فایل &lt;/button&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
خواندن فایل	خروجی

# jQuery < جلوگیری از تداخل کدها در jQuery

## جلوگیری از تداخل کدها در jQuery

### jQuery و سایر فریم ورک های جاوا اسکریپت :

همانطور که می دانید و در بخش آموزش jQuery مشاهده کردید این زبان از علامت \$ در کدهای خود استفاده می کند

برخی دیگر از فریم ورک های جاوا اسکریپت نیز از علامت \$ در کدهای خود استفاده می کنند ، مثل فریم ورکهای Backbone یا GoogleWebToolkit . jQuery و یک فریم ورک دیگر در یک صفحه بصورت همزمان از

\$ استفاده نمایند ، ممکن است در اجرای اسکریپت دچار مشکل شوید و کد شما درست کار نکند .

برای این منظور در jQuery ( ) \$.noConflict استفاده می شود . با استفاده از این متد دیگر نیاز به بکار بردن \$ در کدهای jQuery نبوده و می توانید آن را حذف نمایید . سپس از این علامت در کدهای سایر فریم ورک ها به راحتی و بدون اختلال jQuery استفاده نمایید .

شکل کلی استفاده از این متد به صورت زیر است :

Syntax	\$.noConflict ( ) ;
--------	---------------------

: در مثال زیر نحوه استفاده از متد ( ) \$.noConflict را در عمل نمایش داده ایم . هنگامی که شما ، این متد را در ابتدای

کدهای jQuery خود قرار دهید ، دیگر نیازی به استفاده از علامت \$ نخواهید داشت :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$.noConflict();     jQuery(document).ready(function(){       jQuery("#btn1").click(function(){         jQuery("#p1").text("jQuery می کند");       });     });   &lt; /script &gt;</pre>	کد

<pre>&lt;/head&gt; &lt;body&gt;   &lt;p id="p1" &gt; \$          &lt;/p&gt;   &lt;button id="btn1" type="button"&gt; کلیک نمایند &lt;/button&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>\$</p> <p>کلیک نمایند</p>	خروجی

## jQuery < لیست سایر متدها در jQuery

### لیست سایر متدها در jQuery

در این بخش ، لیست سایر متدهای jQuery .  
برای دریافت اطلاعات و مشاهده مثال های عملی روی نام هر متد کلیک نمایید :

نام رویداد	
<a href="#">data ( )</a>	این متد می تواند به یک عنصر در صفحه داده اضافه کرده و یا داده ای موجود در آن را خوانده و برگرداند .
<a href="#">each ( )</a>	این متد یک تابع تعیین شده را برای تمام عناصری که مشخصات آنها با مشخصات داده شده در پارامتر \$selector یکسان است ، اجرا می کند .
<a href="#">index ( )</a>	این متد می تواند به دنبال عنصر مورد نظرتان از میان عناصر با مشخصات یکسان گشته و ایندکس ( ترتیبی ) .
<a href="#">\$.noconflict ( )</a>	این متد کاربر را از نوشتن علامت \$ در کدهای jQuery ، بی نیاز می کند .
<a href="#">\$.param ( )</a>	این متد می تواند اعضای یک آرایه یا مجموعه خواص یک شی ( object ) را به صورت پشت سر هم و جفت -
<a href="#">removeData ( )</a>	این متد می تواند داده ای که قبلا توسط متد ( data ( ) ، به یک عنصر اضافه شده است را حذف نماید .
<a href="#">size ( )</a>	این متد تعداد عناصر Dom ای که با مشخصات داده شده در پارامتر \$selector ، یکسان هستند را بر می

## jQuery < سایر متدهای jQuery > data ( )

### متد data ( ) jQuery

data ( ) ، می تواند اطلاعاتی را به یک عنصر مورد نظر اضافه کرده و یا اینکه اطلاعات آن را خوانده و به صفحه

**نکته :** برای حذف اطلاعات اضافه شده از [removeData \( \)](#) استفاده نمایید .

. خواندن و برگرداندن اطلاعات یک عنصر با استفاده از متد data ( ) :

به وسیله متد data ( ) می توانید اطلاعات موجود در یک عنصر HTML ه صفحه بر گردانید .  
شکل کلی استفاده از متد data ( ) در این حالت به صورت زیر است :

Syntax	<code>\$( selector ).data ( name ) ;</code>
--------	---

توضیح هر یک از موارد syntax :

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم اطلاعات آن را خوانده و به صفحه بر گردانیم .	selector
این پارامتر تعیین کننده نام اطلاعاتی در عنصر مورد نظر است که می خواهید آن را خوانده و به صفحه بر گردانید . استفاده از این پارامتر اختیاری است .	name

. اضافه کردن اطلاعات به یک عنصر توسط متد data ( ) :

به وسیله متد data ( ) می توان اطلاعات مورد نظر خود را به یک عنصر در صفحه اضافه نمایید .  
شکل کلی استفاده از متد data ( ) در این حالت به صورت زیر است :

Syntax	<code>\$( selector ).data ( name , value ) ;</code>
--------	---

توضیح هر یک از موارد syntax

توضیح syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم اطلاعات آن را خوانده و به صفحه بر گردانیم .	selector
این پارامتر تعیین کننده نام اطلاعاتی است که می خواهیم به عنصر مورد نظرمان اضافه نماییم . استفاده از این پارامتر اجباری است .	name
این پارامتر تعیین کننده اطلاعاتی است که می خواهیم به عنصر مورد نظرمان اضافه نماییم . استفاده از این پارامتر اجباری است .	value

: در مثال زیر ابتدا یک اطلاعات با نام greeting Hello World را به عنصر div اضافه کرده ایم . سپس این اطلاعات را خوانده و نمایش داده ایم :

Example	
<pre>&lt; html &gt; &lt; head &gt; &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt; &lt; script type="text/javascript" &gt; \$( document ). ready ( function ( ) {     \$("#btn1").click(function(){         \$("#div1").data("greeting", "Hello World");     });     \$("#btn2").click(function(){         alert(\$("#div1").data("greeting"));     }); });</pre>	کد

<pre>}); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;button id="btn1"&gt; div اضافه کردن اطلاعات به عنصر &lt;/button&gt;&lt;br&gt;   &lt;button id="btn2"&gt; div &lt;/button&gt;   &lt;div id="div1"&gt;   &lt;/div&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>div اضافه کردن اطلاعات به عنصر</p> <p>div</p>	<p>خروجی</p>

## jQuery < سایر متدهای jQuery > each ( )

### jQuery ( ) each

each ( ) ، یک تابع را تعیین کرده و به ازای هر عنصری که با مشخصات داده شده در پارامتر \$selector داشته باشد ، یکبار اجرا می کند .  
شکل کلی استفاده از متد ( ) each به صورت زیر است :

Syntax	\$( selector ).each( function( index,element ) ) ;
--------	--

توضیح هر یک از موارد syntax	
توضیح	Parameter
این پارامتر تعیین کننده کلاس یا تگ عنصرهایی است که می خواهیم تابع مورد نظرمان به ازای هر نمونه از آن ، یکبار اجرا شود .	selector
تعیین کننده اندیس عنصر مورد نظر است .	index
نشانیگر عنصر جاری است . ( "this" هم می تواند استفاده شود ) .	element

: با ارائه یک مثال عملی کار با متد ( ) each را در عمل نمایش خواهیم داد . در مثال زیر با استفاده از یک متد دستور خواندن متن هر آیتم عضو کلاس ex را یکبار اجرا کرده و آن را در خروجی نمایش داده ایم :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$( document ). ready ( function ( ) {       \$("#btn1").click(function () {         \$("li.ex").each(function () {           alert(\$(this).text())         });       });     });   &lt; /script &gt;</pre>	کد



<pre>}); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;   &lt;button id="btn1" &gt; کلیک نمایید &lt;/button&gt;   &lt;ul&gt;     &lt;li class="ex"&gt;Coffee&lt;/li&gt;     &lt;li class="ex"&gt;Milk&lt;/li&gt;     &lt;li class="ex"&gt;Soda&lt;/li&gt;   &lt;/ul&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>کلیک نمایید</p> <ul style="list-style-type: none"><li>• Coffee</li><li>• Milk</li><li>• Soda</li></ul>	خروجی

## jQuery < سایر متدهای jQuery < index ( )

### jQuery : ( ) index

index ( ) ، اندیس عنصر مورد نظر را نسبت به سایر عناصر مجاور آن خوانده و بر می گرداند .

عنصر می تواند توسط انتخاب کننده های تعیین شده و یا براساس اشیای DOM .

**نکته :** اگر عنصر مورد نظر پیدا نشود ، مقدار برگشتی - خواهد بود .

**پیدا کردن اندیس اولین عنصر مطابق با جستجو نسبت به سایر عناصر مشابه :**

به وسیله این متد می توان اندیس اولین عنصر مطابق با اطلاعات داده شده را خوانده و به صفحه برگرداند .

شکل کلی استفاده از متد index ( ) ، در این حالت به صورت زیر است :

Syntax	<code>\$(selector).index( ) ;</code>
--------	--------------------------------------

توضیح syntax	
توضیح	Parameter
تعیین کننده id عنصری است که می خواهیم اندیس آن را پیدا کنیم . استفاده از این پارامتر اجباری است .	selector

**در مثال زیر با استفاده از متد index ( ) ، کدی نوشته ایم که با کلیک بر روی هر یک از آیتم های لیست در خروجی**

**مثال ، می توانید اندیس آن را به دست آورید :**

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$(document).ready ( function( ) {</pre>	کد

<pre>    \$("li.ex").click(function(){         alert(\$(this).index());     }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;p&gt;برای مشاهده اندیس هر عنصر ، روی آن کلیک نمایید&lt;/p&gt;     &lt;ul&gt;         &lt;li class="ex"&gt;Coffee&lt;/li&gt;         &lt;li class="ex"&gt;Milk&lt;/li&gt;         &lt;li class="ex"&gt;Soda&lt;/li&gt;     &lt;/ul&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>برای مشاهده اندیس هر عنصر ، روی آن کلیک نمایید</p> <ul style="list-style-type: none"><li>• Coffee</li><li>• Milk</li><li>• Soda</li></ul>	<p>خروجی</p>

## jQuery < سایر متدهای jQuery < \$.param ( )

### متد \$.param ( ) jQuery

( \$.param ) ، می تواند اعضای یک آرایه یا مجموعه خواص یک شی را به صورت پشت سر هم و جفت نام-خوانده و به صفحه بر گرداند .  
برای مثال فرض کنید که یک شی خاصیت داشته باشد . خروجی این متد در هنگام استفاده به صورت زیر می باشد :

Syntax	prop1 = value1 & prop2 = value2 & prop3 = value3 ;
--------	--

از این رشته خواص و یا اعضای آرایه می توان برای ارسال اطلاعات در روش `get` استفاده کرد . این اطلاعات در انتهای URL صفحه اضافه شده و به صفحه مقصد ارسال می شوند .  
شکل کلی استفاده از متد ( \$.param ) به صورت زیر است :

Syntax	\$.param( object ) ;
--------	----------------------

توضیح syntax	
توضیح	Parameter
تعیین کننده نام شی یا آرایه ای است که می خواهیم مجموعه اعضا و یا خواص آن را به صورت پشت سر هم نمایش دهیم . استفاده از این پارامتر اجباری است .	obj

: در مثال زیر یک شی به نام `person` را تعریف کرده و خواصی را به آن نسبت داده ایم . سپس به وسیله متد ( \$.param ) ، مجموعه جفت نام-مقدار ، خواص آن را در خروجی نمایش داده ایم :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       personObj = new Object ( );</pre>	کد

<pre>personObj.firstname="John"; personObj.lastname="Doe"; personObj.age=50; \$("#btn1").click(function(){     \$("#div1").text(\$.param( personObj )); }); }); &lt;/script&gt; &lt;/head&gt; &lt;body&gt;     &lt;button id="btn1"&gt; person خواندن اطلاعات شی &lt;/button&gt;&lt;br&gt;     &lt;div id="div1"&gt;     &lt;/div&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
خواندن اطلاعات شی person	خروجی

## jQuery < سایر متدهای jQuery < removeData ( )

### jQuery ( ) removeData

removeData ( ) ، می تواند اطلاعات اضافه شده به یک عنصر را که قبلا توسط [data \( \)](#) اضافه شده است ، را حذف نماید .

**نکته :** اگر اطلاعاتی قبلا توسط متد ( ) data ، به عنصر اضافه نشده باشد ، این متد کار نمی کند .

شکل کلی استفاده از متد ( ) removeData در این حالت به صورت زیر است :

Syntax	<code>\$( selector ).removeData ( name ) ;</code>
--------	---

توضیح موارد syntax	
توضیح	Parameter
این پارامتر تعیین کننده id عنصری است که می خواهیم اطلاعات آن را حذف نماییم .	selector
این پارامتر تعیین کننده نام اطلاعاتی در عنصر مورد نظر است که می خواهید آن را حذف نمایید . استفاده از این پارامتر اختیاری است . اگر این پارامتر تعیین نشود ، کلیه اطلاعات موجود در عنصر حذف خواهد شد .	name

## jQuery < سایر متدهای jQuery < size ( )

### متد size ( ) jQuery

متد size ( ) [DOM](#) ای که با مشخصات تعیین شده برای آن در پارامتر \$selector ، مطابقت دارند را بر می

به عبارت دیگر ، این متد برای شمارش تعداد یک عنصر یا تگ مورد نظر ، بر روی صفحه استفاده می شود .  
شکل کلی استفاده از متد size ( ) به صورت زیر است :

Syntax	\$(selector).size( ) ;
--------	------------------------

: در مثال زیر متد size ( ) ، تعداد عضوهای کلاس ex را شمارش کرده و آن را در خروجی بر می گرداند :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$("#btn1").click(function () {         alert(\$("#li.ex").size());       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; button id ="btn1" &gt; کلیک نمایید &lt; /button &gt;   &lt; ul &gt;     &lt; li class="ex"&gt;Coffee&lt; /li &gt;     &lt; li class="ex"&gt;Milk&lt; /li &gt;     &lt; li class="ex"&gt;Soda&lt; /li &gt;   &lt; /ul &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد

<pre>&lt;/ul&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
<p>کلیک نمایه</p> <ul style="list-style-type: none"><li>• Coffee</li><li>• Milk</li><li>• Soda</li></ul>	خروجی



## jQuery < سایر متدهای jQuery < toArray ( )

jQuery ( ) toArray :

toArray ( ) ، کلیه عناصری که با مشخصات داده شده در پارامتر \$selector آن مطابقت دارند ، را در به صورت یک آرایه در خروجی بر می گرداند .  
شکل کلی استفاده از متد ( ) toArray به صورت زیر است :

Syntax	\$(selector).toArray( ) ;
--------	---------------------------

: در مثال زیر متد ( ) toArray ، به ازای هر عضو کلاس ex یک دستور را اجرا کرده و مقدار متن آن را در خروجی بر می گرداند :

Example	
<pre>&lt; html &gt; &lt; head &gt;   &lt; script type="text/javascript" src="jquery.js" &gt; &lt; /script &gt;   &lt; script type="text/javascript" &gt;     \$ (document). ready ( function ( ) {       \$("#btn1").click(function ( ) {         x = \$("li.ex").toArray()         for (i = 0; i &lt; x.length; i++) {           alert(x[i].innerHTML);         }       });     });   &lt; /script &gt; &lt; /head &gt; &lt; body &gt;   &lt; button id ="btn1" &gt; کلیک نمایید &lt; /button &gt;   &lt; ul &gt;     &lt; li class="ex"&gt;Coffee&lt; /li &gt;   &lt; /ul &gt; &lt; /body &gt; &lt; /html &gt;</pre>	کد

<pre>&lt;li class="ex"&gt;Milk&lt;/li&gt; &lt;li class="ex"&gt;Soda&lt;/li&gt; &lt;/ul&gt; &lt;/body&gt; &lt;/html&gt;</pre>	
کلیک نمایید <ul style="list-style-type: none"><li>• Coffee</li><li>• Milk</li><li>• Soda</li></ul>	خروجی