

به نام خدا

برنامه نویسی مبتنی بر وب



فصل چهارم

jQuery

jQuery چیست؟

- **jQuery** یک کتابخانه سبک جاوا اسکریپت است که امکان انجام کارهای بیشتر با کد نویسی کمتر را می دهد. هدف jQuery استفاده آسانتر از JavaScript در وب سایت است.



Lightweight Footprint

Only 30kB minified and gzipped.
Can also be included as an AMD module



CSS3 Compliant

Supports CSS3 selectors to find elements as well as in style property manipulation



Cross-Browser

Chrome, Edge, Firefox, IE, Safari,
Android, iOS, and more

اضافه کردن jQuery به صفحه وب

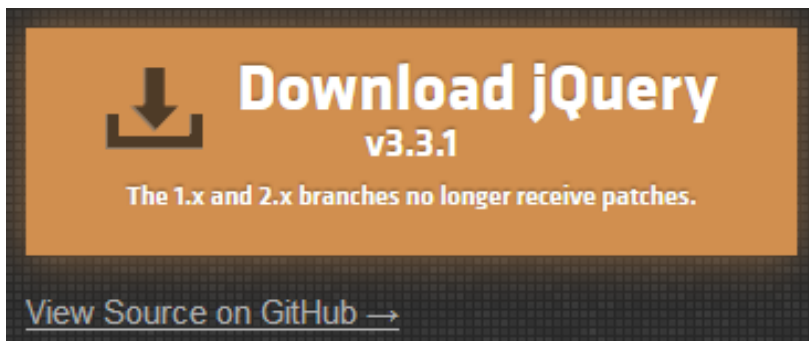
- چندین راه برای استفاده از jQuery وجود دارد:

✓دانلود کتابخانه jQuery از سایت <http://jquery.com/> و لینک کردن آن به

صفحه وب توسط تگ `<script>`

✓استفاده از یک CDN مانند Google و اضافه کردن URL آن به صفحه وب

توسط تگ `<script>`



دانلود jQuery

❖ دو نسخه از jQuery برای دانلود در دسترس است:

- **نسخه Production**: این نسخه، یک فایل فشرده از jQuery است و برای وب سایت های در حال کار مناسب است.
- **نسخه Development**: این نسخه، یک فایل بدون فشرده سازی و قابل خواندن است و برای زمانی که در حال آماده سازی وب سایت هستید مناسب است.

[Download the compressed, production jQuery 3.3.1](#)

[Download the uncompressed, development jQuery 3.3.1](#)

نحوه ارتباط دادن با فایل html

- کتابخانه jQuery، تنها یک فایل JavaScript است و همانطور که گفته شد می توانید توسط تگ `<script>` آن را به صفحه وب لینک کنید.
- ✓ خصوصیت `src` باید با آدرس فایل دانلود شده تنظیم شود، در مثال بالا، فایل jQuery دقیقاً در همان مسیر صفحه جاری قرار دارد. اگر فایل jQuery در آدرسی غیر از آدرس صفحه جاری وجود دارد باید آنرا در نظر بگیرید.

```
<head>
```

```
<script src="jquery-3.3.1.min.js"></script>
```

```
</head>
```

jQuery CDN جایگزینی برای دانلود

- اگر نمی خواهید فایل jQuery را در بین فایل های پروژه تان قرار دهید، می توانید با استفاده از یک **CDN** آن را به صفحه وب لینک کنید.
- Google و Microsoft هر دو میزبان jQuery هستند.

Google CDN:

```
<head>  
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js"></script>  
</head>
```

Microsoft CDN:

```
<head>  
<script src="https://ajax.aspnetcdn.com/ajax/jquery/jquery-3.3.1.min.js"></script>  
</head>
```

یک مزیت بزرگ استفاده از CDN

- زمانی که بازدیدکنندگان سایت شما، قبلاً در سایت های دیگر، فایل jQuery را از CDN های Google یا Microsoft دریافت کرده اند، در دفعات بعدی، فایل jQuery از Cache فراخوانی می شود که باعث افزایش سرعت می شود.

jQuery Syntax

- اصلی ترین دستور در jQuery به صورت زیر است:
- Basic syntax is: `$(selector).action()`
 - ✓ A `$` sign to define/access jQuery
 - ✓ A (*selector*) to "query (or find)" HTML elements
 - ✓ A jQuery *action()* to be performed on the element(s)

مثال

Examples:

`$(this).hide()` - hides the current element.

`$("p").hide()` - hides all `<p>` elements.

`$(".test").hide()` - hides all elements with `class="test"`.

`$("#test").hide()` - hides the element with `id="test"`.

```
$(this).hide() //عنصر جاری را پنهان می کند
```

```
$("p").hide() //تمام پاراگراف ها را پنهان می کند
```

```
$(".test").hide() //تمام عناصری که شامل کلاس مشخص شده در این کد هستند را پنهان می کند
```

```
$("#test").hide() //عنصری که شامل شناسه مشخص شده در این کد هستند را پنهان می کند
```

رویداد ready (Document Ready Event)

- تمام متدهای jQuery داخل رویداد **ready** آورده می شوند. این امر باعث می شود تا کدهای **jQuery** بعد از بارگذاری کامل صفحه، اجرا شوند. همچنین با این تکنیک، قادر خواهید بود کدهای JavaScript را در قسمت `<head>` صفحه قرار دهید.

```
$(document).ready(function){  
  
    // jQuery methods go here...  
  
});
```

نکته

- تیم jQuery یک متد کوتاه تر برای رویداد ready ایجاد کرده است:

```
$(function(){  
  
    // jQuery methods go here...  
  
});
```

- می توانید هر کدام از روش های بالا را که ترجیح می دهید استفاده کنید. اما توجه داشته باشید که روش اول قابل فهم تر و خواناتر است (easier to understand when reading the code).

گزینشگرهای (jQuery Selectors)

- گزینشگرهای jQuery به شما این اجازه را می دهند تا عناصر HTML را انتخاب و دستکاری کنید. با گزینشگرهای jQuery می توانید عناصر HTML را براساس **شناسه**، **کلاس**، **نوع**، **نام خاصیت**، **مقدار خاصیت** و ... پیدا کنید. این تکنیک براساس گزینشگرهای CSS است و علاوه بر این، jQuery تعدادی گزینشگر سفارشی نیز دارد.
- تمام گزینشگرها در jQuery با علامت "\$" آغاز و با پرانتز محصور می شوند. **\$()**

گزینشگر نام تگ element Selector

- گزینشگر نام تگ، به شما اجازه می دهد تا عناصر را براساس نام تگ، انتخاب کنید.
- ❖ به عنوان مثال می توانید تمام تگ های `<p>` را در صفحه مانند زیر انتخاب کنید:

```
$("p")
```

Example

```
$(document).ready(function(){  
    $("button").click(function(){  
        $("p").hide();  
    });  
});
```

This is a heading

This is a paragraph.

This is another paragraph.

Click me to hide paragraphs

```
<body>  
  
<h2>This is a heading</h2>  
  
<p>This is a paragraph.</p>  
<p>This is another paragraph.</p>  
  
<button>Click me to hide paragraphs</button>  
  
</body>
```



Code 1.txt

انتخابگر شناسه #id Selector

- گزینه‌گر شناسه، از مقدار خصوصیت id تگ HTML برای پیدا کردن عنصر استفاده می‌کند. شناسه در یک صفحه، باید یکتا باشد، بنابراین زمانی که بخواهید به یک عنصر منحصر به فرد در صفحه دسترسی داشته باشید از این روش استفاده نمایید. برای پیدا کردن یک عنصر با شناسه مشخص، از کاراکتر "#" همراه با شناسه مورد نظر استفاده نمایید.

```
$("#test")
```

Example

```
$(document).ready(function(){
    $("button").click(function(){
        $("#test").hide();
    });
});
```

This is a heading

This is a paragraph.

This is another paragraph.

```
<body>

<h2>This is a heading</h2>

<p>This is a paragraph.</p>
<p id="test">This is another paragraph.</p>

<button>Click me</button>

</body>
```



Code 2.txt

گزینشگر نام کلاس class Selector.

- گزینشگر نام کلاس، عناصر با کلاس مشخص را پیدا می کند. برای پیدا کردن عناصر با نام کلاس مشخص، از کاراکتر "." همراه با نام کلاس مورد نظر استفاده کنید.

```
$(".test")
```

Example

```
$(document).ready(function(){
    $("button").click(function(){
        $(".test").hide();
    });
});
```

```
<body>
```

```
<h2 class="test">This is a heading</h2>
```

```
<p class="test">This is a paragraph.</p>
```

```
<p>This is another paragraph.</p>
```

```
<button>Click me</button>
```

```
</body>
```

This is a heading

This is a paragraph.

This is another paragraph.

Click me



Code 3.txt

مثال های بیشتر از گزینشگر های jQuery



Code 4.txt



Code 5.txt

Syntax	Description
<code>\$("*")</code>	Selects all elements
<code>\$(this)</code>	Selects the current HTML element

<code>\$("*")</code>	تمام عناصر را انتخاب می کند.
<code>\$(this)</code>	عنصر جاری را انتخاب می کند.

مثال های بیشتر از گزینشگر های jQuery

`$("p.intro")` Selects all `<p>` elements with `class="intro"`



Code 6.txt

`$("p:first")` Selects the first `<p>` element



Code 7.txt

`$("ul li:first")` Selects the first `` element of the first ``



Code 8.txt

تمام عناصر `<p>` با کلاس `"intro"` را انتخاب می کند.

`s("p.intro")`

اولین عنصر `<p>` را انتخاب می کند.

`s("p:first")`

اولین آیتم از اولین عنصر `` را انتخاب می کند.

`s("ul li:first")`

توابع پرکاربرد در یک فایل مجزا

- اگر وب سایت شما شامل تعداد زیادی صفحه است، و می خواهید توابع jQuery یکسانی را در آنها استفاده کنید، می توانید این توابع را در یک فایل مجزا با پسوند .js قرار دهید.
- همانطور که در این آموزش دیده اید، توابع به طور مستقیم در قسمت <head> صفحه قرار داده شده اند. اما بعضی مواقع مانند بالا، ترجیح داده می شود که آنها را در یک فایل JavaScript جداگانه با پسوند .js ذخیره کنیم و مانند زیر، توسط تگ <script> به صفحه لینک کرد.

Example

```
<head>
<script src="https://ajax.googleapis.com/ajax/libs/jquery/3.3.1/jquery.min.js">
</script>
<script src="my_jquery_functions.js"></script>
</head>
```

رویدادها jQuery Event Methods

- تمام کارهایی که در یک صفحه وب اتفاق می افتد و می توانیم به آنها پاسخ دهیم، رویداد (Event) گفته می شود.
- مثال:
 - ✓ حرکت موس بر روی یک عنصر
 - ✓ انتخاب یک radio button
 - ✓ کلیک کردن بر روی یک عنصر
- واژه "اتفاق افتادن" اغلب همراه با واژه ی "رویداد" استفاده می شود. مثال: رویداد keypress زمانی اتفاق می افتد که یکی از دکمه های صفحه کلید را فشار دهید.

عناوین از رویدادهای رایج در مدل DOM

Mouse Events	Keyboard Events	Form Events	Document/Window Events
click	keypress	submit	load
dblclick	keydown	change	resize
mouseenter	keyup	focus	scroll
mouseleave		blur	unload

نحوه استفاده از رویدادها در jQuery

- بیشتر رویدادهای مدل DOM، یک متد معادل در jQuery دارند.

- مثال

✓ برای اختصاص دادن رویداد onclick به تمام تگ های <p>، می توانید مانند زیر عمل کنید:

```
$("#p").click();
```

✓ قدم بعدی، مشخص کردن کاری است که باید به ازای رویداد onclick اتفاق بیافتد. باید یک تابع را به عنوان پارامتر به متد click ارسال کنید.

```
$("#p").click(function(){  
    // action goes here!!  
});
```


رویدادهای رایج در jQuery

- `$(document).ready()`
- `click()`
- `dblclick()`
- `mouseenter()`
- `mouseleave()`
- `mousedown()`
- `mouseup()`
- `hover()`
- `focus()`
- `on()`

متد click()

- این متد، یک تابع را برای پاسخ دادن به رویداد `onclick` یک عنصر، به آن اختصاص می دهد. زمانی که کاربر روی یک عنصر `HTML` کلیک کند، رویداد `onclick` اتفاق می افتد.

Example

```
$("#p").click(function(){  
    $(this).hide();  
});
```

```
<body>  
  
<p>If you click on me, I will disappear.</p>  
<p>Click me away!</p>  
<p>Click me too!</p>  
  
</body>
```



Code 9.txt

رویداد `dblclick()`

- این متد، یک تابع را برای پاسخ دادن به رویداد `double-clicks` یک عنصر، به آن اختصاص می دهد. زمانی که کاربر روی یک عنصر HTML دوبار کلیک کند، رویداد `dblclick` اتفاق می افتد.

Example

```
$("#p").dblclick(function(){  
    $(this).hide();  
});
```



Code 10.txt

رویداد mouseenter()

- این متد، یک تابع را برای زمانی که موس کاربر روی یک عنصر HTML قرار می گیرد، اجرا می کند.

Example

```
$("#p1").mouseenter(function(){  
    alert("You entered p1!");  
});
```



Code 11.txt

```
<body>  
  
<p id="p1">Enter this paragraph.</p>  
  
</body>
```

رویداد mouseleave()

- زمانی که موس از روی یک عنصر HTML خارج می شود، تابع نوشته شده در این رویداد اجرا می شود.

Example

```
$("#p1").mouseleave(function(){  
    alert("Bye! You now leave p1!");  
});
```

```
<body>
```

```
<p id="p1">This is a paragraph.</p>
```

```
</body>
```



Code 12.txt

رویداد mousedown()

- زمانی که موس روی یک عنصر HTML قرار دارد و کلید چپ، وسط یا راست (left, middle or right mouse button is pressed down) فشرده شود این رویداد پاسخ می دهد.

Example

```
$("#p1").mousedown(function(){  
    alert("Mouse down over p1!");  
});
```



Code 13.txt

```
<body>  
  
<p id="p1">This is a paragraph.</p>  
  
</body>
```

رویداد mouseup()

- زمانی که موس کاربر روی یک عنصر HTML قرار دارد و کلید چپ، وسط یا راست (left, middle or right mouse button is released) را رها کند، این رویداد اجرا می شود.

Example

```
$("#p1").mouseup(function(){  
    alert("Mouse up over p1!");  
});
```



Code 14.txt

رویداد hover()

- این متد، همزمان دو تابع را برای پاسخ دادن به رویداد mouseenter() و mouseleave() یک عنصر، به آن اختصاص می دهد. زمانی که موس کاربر روی عنصر HTML قرار گیرد، تابع اول اجرا می شود و زمانی که موس از روی آن خارج می شود، تابع دوم اجرا خواهد شد.

Example

```
$("#p1").hover(function(){  
    alert("You entered p1!");  
},  
function(){  
    alert("Bye! You now leave p1!");  
});
```



Code 15.txt

رویداد focus()

- این متد، یک تابع را برای پاسخ دادن به رویداد onfocus یک عنصر، به آن اختصاص می دهد. زمانی که علامت چشمک زن مکان نما روی یکی از اجزای فرم، مانند: Text Box قرار گیرد، این رویداد پاسخ می دهد.

Example

```
$("#input").focus(function(){  
    $(this).css("background-color", "#cccccc");  
});
```

Name:

Email:



Code 16.txt

رویداد blur()

- زمانی که تمرکز از روی یکی از اجزای فرم خارج شود، این رویداد پاسخ خواهد داد.

Example

```
$("#input").blur(function(){  
    $(this).css("background-color", "#ffffff");  
});
```



Code 17.txt

on() **متد**

- The on() method attaches one or more event handlers for the selected elements.
- Attach a click event to a <p> element:

Example

```
$("p").on("click", function(){  
    $(this).hide();  
});
```

Attach multiple event handlers to a <p> element:

Example

```
$("#p").on({  
  mouseenter: function(){  
    $(this).css("background-color", "lightgray");  
  },  
  mouseleave: function(){  
    $(this).css("background-color", "lightblue");  
  },  
  click: function(){  
    $(this).css("background-color", "yellow");  
  }  
});
```



Code 18.txt

افکت ها jQuery Effects

- Hide and Show
- fade

Hide and Show

- در jQuery، با متدهای `hide()` و `show()` می توانید عناصر HTML را پنهان و نمایان کنید.

Syntax:

```
$(selector).hide(speed,callback);
```

```
$(selector).show(speed,callback);
```

Example

```
$("#button").click(function(){  
    $("#p").hide(1000);  
});
```

Example

```
$("#hide").click(function(){  
    $("p").hide();  
});
```

```
$("#show").click(function(){  
    $("p").show();  
});
```



Code 19.txt

<body>

<p>If you click on the "Hide" button, I will disappear.</p>

<button id="hide">Hide</button>

<button id="show">Show</button>

</body>

متد toggle()

- با متد toggle () می توانید بین متدهای hide () و show () حرکت کنید. این متد عناصر نمایان را پنهان و عناصر پنهان را نمایان می کند

Syntax:

```
$(selector).toggle(speed,callback);
```

Example

```
$("#button").click(function(){  
    $("#p").toggle();  
});
```


محو کردن عناصر Fading

- با jQuery می توان عناصر را کم کم ناپدید و یا نمایان کرد.

- fadeIn()

fadeIn()

این مثال، چگونگی نمایان کردن عناصر را نشان می دهد.

- fadeOut()

fadeOut()

این مثال، چگونگی محو کردن عناصر را نشان می دهد.

- fadeToggle()

fadeToggle()

این مثال، چگونگی محو یا نمایان کردن عناصر را نشان می دهد.

- fadeTo()

fadeTo()

این مثال، چگونگی محو کردن عناصر تا یک مقدار مشخص را نشان می دهد.

Syntax:

```
$(selector).fadeIn(speed, callback);
```

Syntax:

```
$(selector).fadeToggle(speed, callback);
```

Syntax:

```
$(selector).fadeOut(speed, callback);
```

Syntax:

```
$(selector).fadeTo(speed, opacity, callback);
```

Example

```
$("#button").click(function(){  
    $("#div1").fadeIn();  
    $("#div2").fadeIn("slow");  
    $("#div3").fadeIn(3000);  
});
```



Code 20.txt

Example

```
$("#button").click(function(){  
    $("#div1").fadeOut();  
    $("#div2").fadeOut("slow");  
    $("#div3").fadeOut(3000);  
});
```



Code 21.txt

پنهان کردن عناصر بصورت اسلایدی Sliding

- با jQuery می توان عناصر را بصورت اسلایدی یا کشویی مخفی و یا نمایان کرد.

jQuery has the following slide methods:

- `slideDown()`
- `slideUp()`
- `slideToggle()`

`slideDown()`

در این مثال، چگونگی نمایان کردن یک عنصر پنهان را بصورت اسلایدی نشان می دهد.

`slideUp()`

در این مثال، چگونگی پنهان کردن یک عنصر را بصورت اسلایدی نشان می دهد.

`slideToggle()`

در این مثال، چگونگی نمایان یا پنهان کردن یک عنصر را بصورت اسلایدی نشان می دهد.

Syntax:

```
$(selector).slideDown(speed, callback);
```

Syntax:

```
$(selector).slideUp(speed, callback);
```

متحرک سازی Animation

- متد `animate()` برای متحرک سازی در jQuery استفاده می شود.

Syntax:

```
$(selector).animate({params}, speed, callback);
```

Example

```
$("#button").click(function(){  
    $("#div").animate({left: '250px'});  
});
```

پارامترها:

پارامتر	توضیح
<i>params</i>	الزامی است، یک یا چند خصوصیت دلخواه CSS برای متحرک سازی است.
<i>speed</i>	اختیاری است، سرعت متحرک سازی را مشخص می کند و می تواند با مقادیر "fast"، "slow" و یا میلی ثانیه مقدار دهی شود.
<i>callback</i>	اختیاری است، تابعی است که بعد از اجرای کامل متد <code>animate()</code> اجرا می شود.

Example

```
$("#button").click(function(){  
    $("#div").animate({  
        left: '250px',  
        opacity: '0.5',  
        height: '150px',  
        width: '150px'  
    });  
});
```



Code 22.txt

By default, all HTML elements
remember to first set the



By default, all HTML elements have a static position, and cannot
remember to first set the CSS position property of the element to



متوقف کردن متحرک سازی

- متد `stop()` برای متوقف کردن متحرک سازی استفاده می شود.

Syntax:

```
$(selector).stop(stopAll, goToEnd);
```

پارامترها:

پارامتر	توضیح
<i>stopAll</i>	اختیاری است، مشخص می کند که آیا تمام انیمیشن هایی که در صف قرار دارند متوقف شوند یا نه. مقدار پیش فرض "false" است و معنی آن این است که فقط انیمیشنی که در حال اجراست متوقف خواهد شد و اجازه می دهد که بقیه انیمیشن هایی که در صف قرار دارند اجرا شوند.
<i>goToEnd</i>	اختیاری است، مشخص می کند که آیا انیمیشن جاری فوراً به انتهای اجرا برود یا در همان جا متوقف شود. مقدار پیش فرض "false" است.

Example

```
<script>
$(document).ready(function(){
    $("#flip").click(function(){
        $("#panel").slideDown(5000);
    });
    $("#stop").click(function(){
        $("#panel").stop();
    });
});
</script>
```

```
<body>
```

```
<button id="stop">Stop sliding</button>
```

```
<div id="flip">Click to slide down panel</div>
```

```
<div id="panel">Hello world!</div>
```

```
</body>
```



code 23.txt

زنجیره ای کردن متدها در jQuery

- با jQuery می توانید متدها را به یکدیگر زنجیر کنید. زنجیره ای کردن به ما اجازه می دهد تا تنها با یک دستور، چندین متد را روی یک عنصر مشخص، اجرا کنیم. به این ترتیب، برای اعمال کارهای مختلف، مرورگرها مجبور نیستند که عنصرهای یکسان را بیش از یک مرتبه شناسایی کنند.

Example

```
$("#p1").css("color", "red").slideUp(2000).slideDown(2000);
```


نکته

- زمانی که در حال زنجیره ای کردن متدها هستید، می توانید هر متد را در یک خط مانند زیر قرار دهید:

Example

```
$("#p1").css("color", "red").slideUp(2000).slideDown(2000);
```

Example

```
$("#p1").css("color", "red")  
    .slideUp(2000)  
    .slideDown(2000);
```