

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

JqueryUI - Autocomplete

مدرس : مهندس افشین رفوآ

JqueryUI - Autocomplete

Autocompletion مکانیزمی است که به طور مداوم در وب سایت های مدرن برای ارائه ی لیستی از پیشنهادات به کاربرها برای اولین حرفی که در تکست باکس تایپ کرده، استفاده می شود. سپس کاربر می تواند یک آیتم را از لیست انتخاب کند که در فیلد ورودی نمایش داده خواهد شد. این ویژگی باعث می شود کاربر مجبور به وارد کردن کامل یک لغت یا مجموعه ای لغات نباشد.

JQuery یک ویجت **autocomplete** را ارائه می دهد – کنترلی که بسیار مشابه یک **<select>** **dropdown** کار می کند، اما انتخاب ها را فیلتر می کند تا تنها مواردی را نمایش دهد که کاربر در حال تایپ در یک کنترل می باشد. JQuery متد **autocomplete()** برای ایجاد لیست پیشنهادات در زیر فیلد ورودی، ارائه می دهد و گروه های جدید CSS را به عناصر اضافه می کند تا به آنها استایل جدید را بدهد. هر فیلدی که می تواند یک ورودی دریافت کند، می تواند به یک **Autocomplete** تبدیل شود، یعنی عناصر **<input>**، عناصر **<textarea>**، و عناصری با ویژگی **contenteditable**.

متد **autocomplete()** به دو فرم زیر استفاده می شود:

`$(selector, context).autocomplete (options) Method`

`$(selector, context).autocomplete ("action", params) Method`

متد **\$(selector, context).autocomplete (options)**

متد **autocomplete (options)** اعلام می کند که یک عنصر **HTML <input>** باید به عنوان یک فیلد ورودی تنظیم شود که در بالای لیستی از پیشنهادات نمایش داده خواهد شد. پارامتر **options** آجکتی است که رفتار لیست پیشنهاد ها را در هنگام تایپ کردن کاربر در فیلد ورودی، تعیین می کند.

```
$(selector, context).autocomplete (options);
```

با استفاده از آجکت **Javascript** می توانید یک یا چند گزینه را همزمان ارائه داده و سپس آنها را به وسیله ی کاما از یکدیگر مجزا کنید:

```
$(selector, context).autocomplete({option1: value1, option2: value2..... });
```

جدول زیر گزینه های مختلفی را ارائه می دهد که با این متد استفاده می شوند.

Option	Description
appendTo	این گزینه برای ضمیمه کردن یک عنصر به منو استفاده می شود. مقدار پیش فرض آن null است.
autoFocus	اگر این گزینه روی true تنظیم شده باشد، وقتی منو نمایش داده شود، به طور خودکار اولین آیتم منو متمرکز می شود. مقدار پیش فرض آن false است.
delay	این گزینه یک عدد صحیح می باشد که زمان انتظار را قبل از هماهنگ کردن مقادیر، به هزارم ثانیه نشان می دهد ( همانطور که توسط گزینه ی source تعیین شده است). وقتی یک داده ی غیرداخلی با دادن وقت به کاربر برای وارد کردن کاراکترهای بیشتر قبل از شروع جستجو، مورد استفاده قرار می گیرد، این امر trashing را کاهش می دهد. مقدار پیش فرض آن 300 است.
disabled	اگر این گزینه روی true تنظیم شده باشد، autocomplete widget از ابتدا غیرفعال خواهد بود. مقدار پیش فرض آن false است.

minLength	تعداد کاراکترهایی که باید قبل از هماهنگ سازی مقادیر (همانطور که توسط گزینه ی SOURCE تعیین شد)، وارد شوند. می تواند از تعیین مقادیر بیش از حد بزرگ جلوگیری کند وقتی که تعداد کاراکترها برای تنظیم عدد کافی نیستند. مقدار پیش فرض آن 1 است.
position	این گزینه موقعیت پیشنهادهای منو را در ارتباط با عنصر ورودی مرتبط تشخیص می دهد. گزینه ی Of پیش فرض عنصر ورودی است، اما می توانید عنصر دیگری را در مقابل این موقعیت تعیین کنید. مقدار پیش فرض آن { my: "left top", at: "left bottom", collision: "none" } است.
source	این گزینه روشی را تعیین می کند که در آن داده ی هماهنگ با داده ی ورودی، تنظیم می شود. یک مقدار باید ارائه شود، در غیر این صورت autocomplete widget ایجاد نخواهد شد. مقدار پیش فرض آن none; must be specified is است.

بخش بعدی مثال های کاربردی در مورد قابلیت ویجت **autocomplete** را ارائه می دهد.

### قابلیت پیش فرض:

مثال زیر به طور ساده قابلیت ویجت **autocomplete** را توضیح میدهد که هیچ پارامتری را به متد

**autocomplete()** انتقال نمی دهد.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Autocomplete functionality</title>
  <link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
  <script src="http://code.jquery.com/jquery-1.10.2.js"></script>
  <script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
  <!-- Javascript -->
  <script>
    $(function () {
      var availableTutorials = [
        "ActionScript",
        "Bootstrap",
        "C",

```

```

        "C++",
    ];
    $("#autocomplete-1").autocomplete({
        source: availableTutorials
    });
});
</script>
</head>
<body>
<!-- HTML -->
<div class="ui-widget">
<p>Type "a" or "s"</p>
<label for="autocomplete-1">Tags: </label>
<input id="autocomplete-1">
</div>
</body>
</html>

```

استفاده از autofocus:

مثال زیر استفاده از گزینه ی **autoFocus** را در ویجت **autocomplete** از **jQuery** توضیح می دهد:

```

<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>jQuery UI Autocomplete functionality</title>
<link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<!-- Javascript -->
<script>
$(function () {
    var availableTutorials = [
        "ActionScript",
        "Bootstrap",
        "C",
        "C++",
    ];
    $("#autocomplete-2").autocomplete({
        source: availableTutorials,
        autoFocus: true
    });
});
</script>
</head>
<body>
<!-- HTML -->
<div class="ui-widget">
<p>Type "a" or "s"</p>
<label for="autocomplete-2">Tags: </label>
<input id="autocomplete-2">

```

```
</div>
</body>
</html>
```

## استفاده از delay و minLength:

مثال زیر استفاده از گزینه های **delay** و **minLength** را در ویجت **autocomplete** از **jQuery** توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Autocomplete functionality</title>
  <link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
  <script src="http://code.jquery.com/jquery-1.10.2.js"></script>
  <script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
  <!-- Javascript -->
  <script>
    $(function () {
      var availableTutorials = [
        "ActionScript",
        "Bootstrap",
        "C",
        "C++",
        "Ecommerce",
        "Jquery",
        "Groovy",
        "Java",
        "JavaScript",
        "Lua",
        "Perl",
        "Ruby",
        "Scala",
        "Swing",
        "XHTML"
      ];
      $("#autocomplete-3").autocomplete({
        minLength: 2,
        delay: 500,
        source: availableTutorials
      });
    });
  </script>
</head>
<body>
  <!-- HTML -->
  <div class="ui-widget">
    <p>Type two letter for e.g:ja,sc etc</p>
```

```
<label for="autocomplete-3">Tags: </label>
<input id="autocomplete-3">
</div>
</body>
</html>
```

## استفاده از label:

مثال زیر استفاده از گزینه ی label را در autocomplete widget از JQueryUI توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>jQuery UI Autocomplete functionality</title>
<link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<!-- Javascript -->
<script>
$(function () {
  $("#autocomplete-4").autocomplete({
    source: [
      { label: "India", value: "IND" },
      { label: "Australia", value: "AUS" }
    ]
  });
});
</script>
</head>
<body>
<!-- HTML -->
<div class="ui-widget">
  <p>Type I OR A</p>
  <input id="autocomplete-4">
</div>
</body>
</html>
```

## استفاده از source خارجی

مثال زیر استفاده از فایل خارجی را برای گزینه ی source در autocomplete widget از JQueryUI

توضیح می دهد.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Autocomplete functionality</title>
  <link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
  <script src="http://code.jquery.com/jquery-1.10.2.js"></script>
  <script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
  <!-- Javascript -->
  <script>
    $( "#autocomplete-5" ).autocomplete({
      source: "jqueryui/search.php",
      minLength: 2
    });
  </script>
</head>
<body>
  <input id="autocomplete-5">
</body>
</html>

```

فایل **search.php** مانند فایل بالا، در همان موقعیت جایگزین می شود

(**autocompleteexample.html**). محتوای **search.php** در زیر نمایش داده می شود.

```

<?
$term = $_GET[ "term" ];
$companies = array(
  array( "label" => "JAVA", "value" => "1" ),
  array( "label" => "DATA IMAGE PROCESSING", "value" => "2" ),
  array( "label" => "JAVASCRIPT", "value" => "3" ),
  array( "label" => "DATA MANAGEMENT SYSTEM", "value" => "4" ),
  array( "label" => "COMPUTER PROGRAMMING", "value" => "5" ),
  array( "label" => "SOFTWARE DEVELOPMENT LIFE CYCLE", "value" => "6" ),
  array( "label" => "LEARN COMPUTER FUNDAMENTALS", "value" => "7" ),
  array( "label" => "IMAGE PROCESSING USING JAVA", "value" => "8" ),
  array( "label" => "CLOUD COMPUTING", "value" => "9" ),
  array( "label" => "DATA MINING", "value" => "10" ),
  array( "label" => "DATA WAREHOUSE", "value" => "11" ),
  array( "label" => "E-COMMERCE", "value" => "12" ),
  array( "label" => "DBMS", "value" => "13" ),
  array( "label" => "HTTP", "value" => "14" )
);
$result = array();
foreach ( $companies as $company ) {
  $companyLabel = $company[ "label" ];
  if ( strpos( strtoupper( $companyLabel ), strtoupper( $term )
    !== false ) {
    array_push( $result, $company );
  }
}
echo json_encode( $result );
?>

```

متد `$(selector, context).autocomplete ("action", params)`:

متد `autocomplete ("action", params)` می تواند فعالیت را روی لیست پیشنهادات انجام دهد، به عنوان مثال پنهان کردن و یا نمایش دادن. فعالیت به عنوان یک **String** در اولین برهان نمایش داده می شود ( به عنوان مثال `close` برای پنهان کردن لیست). اکشن هایی را که قابل انتقال هستند، در جدول زیر مشاهده می کنید.

`$(selector, context).autocomplete ("action", params);`

جدول زیر اکشن هایی را ارائه می دهد که با این متد استفاده می شوند:

Action	Description
close	این اکشن لیست پیشنهادات را در <code>Autocomplete</code> پنهان می کند. این متد هیچ آرگیومنتی نمی پذیرد.
destroy	این اکشن قابلیت <code>Autocomplete</code> را حذف می کند. لیست پیشنهادات پاک می شود و این متد هیچ آرگیومنتی نمی پذیرد.
disable	این اکشن مکانیزم <code>autocomplete</code> را غیرفعال می سازد. لیست پیشنهادات دیگر ظاهر نمی شود. این متد هیچ آرگیومنتی نمی پذیرد.
enable	این اکشن مکانیزم <code>autocomplete</code> را مجددا فعالسازی می کند. لیست پیشنهادات نیز دوباره نمایش داده خواهند شد. این متد هیچ آرگیومنتی نمی پذیرد.

مثال:



option( optionName )	این اکشن مقدار تعیین شده برای پارامتر optionName را بازیابی می کند. این گزینه با یکی از گزینه های به کار برده شده در autocomplete مطابقت دارد.
option	این اکشن یک آبجکت حاوی جفت های key/value را دریافت می کند که گزینه های hash مربوط به autocomplete را نشان می دهد.
option( optionName, value )	این اکشن مقدار گزینه ی autocomplete را مربوط به optionName تعیین شده، تنظیم می کند. میحث optionName نام گزینه ای است که باید تنظیم شود و value مقداری است که باید برای گزینه تنظیم شود.
option( options )	این اکشن یک یا چند گزینه را برای autocomplete تنظیم می کند. میحث options نقشه ای است از جفت های option-value که قرار است تنظیم شوند.
search( [value ] )	این اکشن به دنبال نوعی سازگاری بین مقدار رشته و منبع داده است ( در options.source تعیین شده است). حداقل تعداد کارکترها باید به مقدار برسند، در غیر این صورت جستجو اجرا نمی شود.
widget	عنصر DOM <ul> را بازمی گرداند که با لیست پیشنهادها مطابقت دارد. این اکشن، آبجکتی است از گروه jQuery که امکان دسترسی آسان به لیست را بدون استفاده از انتخابگرهای jQuery فراهم می کند.

اکنون اجازه بدهید با استفاده از اکشن های بالا، مثالی را بررسی کنیم. مثال زیر استفاده از متد **option()**

**optionName, value )** را توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>jQuery UI Autocomplete functionality</title>
<link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<!-- Javascript -->
<script>
```

```

$(function () {
var availableTutorials = [
"ActionScript",
"Bootstrap",
"C",
"C++",
"Ecommerce",
"Jquery",
"Groovy",
"Java",
"JavaScript",
"Lua",
"Perl",
"Ruby",
"Scala",
"Swing",
"XHTML"
];
$("#autocomplete-6").autocomplete({
source: availableTutorials
});
$("#autocomplete-6").autocomplete("option", "position", { my: "right-10 top+10", at: "right top" });
});
</script>
</head>
<body>
<!-- HTML -->
<div class="ui-widget">
<p>Type "a" or "s"</p>
<label for="autocomplete-6">Tags: </label>
<input id="autocomplete-6">
</div>
</body>
</html>

```

نکات گسترش:

**autocomplete widget** می تواند همانطور که با ساخته شد، گسترش یابد. وقتی ویجت ها را گسترش می دهید، قادرید که رفتارهایی را به متدهای موجود اضافه کرده و یا حذف کنید. جدول زیر متدهایی را ارائه می دهد که در نکات گسترش با همان ثبات **API** رفتار می کنند که متدهای پلاگین در بالا داشتند.

Method	Description
--------	-------------

<code>_renderItem( ul, item )</code>	این متد ایجاد هر گزینه را در منوی ویجت کنترل می کند. این متد یک عنصر <code>&lt;li&gt;</code> جدید ایجاد می کند، آن را به منو ضمیمه کرده و بازمی گرداند.
<code>_renderMenu(ul, items)</code>	این متد ساخت منوی ویجت را کنترل می کند.
<code>_resizeMenu()</code>	این متد اندازه ی منو را قبل از نمایش داده شدن آن کنترل می کند. عنصر منو در <code>menu.element</code> موجود می باشد. این متد هیچ برهانی را نمی پذیرد.

مدیریت رویداد روی عنصر `autocomplete`:

علاوه بر متد `autocomplete (options)` که در بخشهای قبل مشاهده کردیم، **JqueryUI** متدهای دیگری ارائه می دهد که برای یک رویداد خاص آغاز می شوند. این متدها در زیر ارائه شده اند:

Event Method	Description
<code>change(event, ui)</code>	این رویداد زمانی آغاز می شود که مقدار عنصر <code>&lt;input&gt;</code> براساس یک انتخاب تغییر می کند. در زمان آغاز، این رویداد همیشه پس از آغاز رویداد <code>close</code> اتفاق می افتد.
<code>close(event, ui)</code>	هر زمانی که منوی <code>autocomplete</code> بسته شود، این رویداد آغاز می شود.
<code>create(event, ui)</code>	وقتی <code>autocomplete</code> ایجاد می شود، این رویداد آغاز می شود.

<code>focus(event, ui)</code>	هرزمان که تمرکز بر روی یکی از انتخاب های منو باشد، این رویداد آغاز می شود. مقدار تمرکز بر روی <code>&lt;input&gt;</code> تنظیم می شود، مگر اینکه کنسل شود ( به عنوان مثال با گزارش دادن <code>false</code> ).
<code>open(event, ui)</code>	پس از اینکه داده آماده شده است و منو در آستانه ی باز شدن می باشد، این رویداد آغاز می شود.
<code>response(event, ui)</code>	پس از اینکه جستجو کامل شده و منو نمایش داده شود، این رویداد آغاز می شود. این رویداد همیشه با تکمیل شدن جستجو آغاز می شود، حتی اگر منو نمایش داده نشود، زیرا هیچ نتیجه ای وجود ندارد، یا <code>Autocomplete</code> غیرفعال شده است.
<code>search(event, ui)</code>	این رویداد پس از اینکه ضوابط <code>delay</code> و <code>minLength</code> اجرا شده اند، آغاز می شود، درست قبل از اینکه مکانیزم تعیین شده توسط منبع، فعال شود. اگر کنسل شود، عملکرد جستجو هم متوقف می شود.
<code>select(event, ui)</code>	این رویداد زمانی آغاز می شود که یک مقدار از منوی <code>autocomplete</code> انتخاب شده باشد. کنسل کردن این رویداد از تنظیم مقدار برای عنصر <code>&lt;input&gt;</code> جلوگیری می کند ( اما از بسته شدن منو جلوگیری نمی کند ).

مثال:

مثال زیر استفاده از متد رویداد را در `autocomplete widgets` توضیح می دهد. این مثال استفاده از رویدادهای `focus` و `select` را توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
<meta charset="utf-8">
<title>jQuery UI Autocomplete functionality</title>
```

```

<link href="http://code.jquery.com/ui/1.10.4/themes/ui-lightness/jquery-ui.css" rel="stylesheet">
<script src="http://code.jquery.com/jquery-1.10.2.js"></script>
<script src="http://code.jquery.com/ui/1.10.4/jquery-ui.js"></script>
<style>
#project-label {
display: block;
font-weight: bold;
margin-bottom: 1em;
}
#project-icon {
float: left;
height: 32px;
width: 32px;
}
#project-description {
margin: 0;
padding: 0;
}
</style>
<!-- Javascript -->
<script>
$(function () {
var projects = [
{
value: "java",
label: "Java",
desc: "write once run anywhere",
},
{
value: "jquery-ui",
label: "jQuery UI",
desc: "the official user interface library for jQuery",
},
{
value: "Bootstrap",
label: "Twitter Bootstrap",
desc: "popular front end frameworks ",
}
];
$("#project").autocomplete({
minLength: 0,
source: projects,
focus: function (event, ui) {
$("#project").val(ui.item.label);
return false;
},
select: function (event, ui) {
$("#project").val(ui.item.label);
$("#project-id").val(ui.item.value);
$("#project-description").html(ui.item.desc);
return false;
}
})
.data("ui-autocomplete")._renderItem = function (ul, item) {
return $("- ")
.append("<a>" + item.label + "<br>" + item.desc + "</a>")

```

```
        .appendTo(ul);
    };
});
</script>
</head>
<body>
<div id="project-label">Select a project (type "a" for a start):</div>
<input id="project">
<input type="hidden" id="project-id">
<p id="project-description"></p>
</body>
</html>
```

www.tahlildadeh.com