

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

JqueryUI - Droppable

مدرس : مهندس افشین رفوآ

JqueryUI - Droppable

JQueryUI متد `droppable()` را برای ساخت هر گونه عنصر **DOM** از نوع **droppable** در یک هدف تعیین شده، ارائه می دهد.

ترکیب:

متد `droppable()` به دو روش مورد استفاده قرار می گیرد:

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

`$(selector, context).droppable ("action", params)`

متد `$(selector, context).droppable (options)`

متد `droppable (options)` بیان می کند که یک عنصر **HTML** می تواند به عنوان عنصری مورد استفاده قرار بگیرد که می توانید در آن عناصر دیگری قرار دهید. پارامتر **options** آبجکتی است که رفتار عناصر شامل را تعیین می کند.

ترکیب

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

می توانید با استفاده از آبجکت **Javascript**، یک یا چند گزینه هم زمان ارائه دهید. اگر بیشتر از یک گزینه برای ارائه وجود داشته باشد، می توانید با استفاده از ویرگور آنها را از یکدیگر مجزا کنید، مانند ترکیب زیر:

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

جدول زیر گزینه های مختلفی را ارائه می دهد که با این متد مورد استفاده قرار می گیرند:

Option	Description
accept	این گزینه زمانی استفاده می شود که لازم است عنصرهای جابجایی را که قرار است برای دراپ کردن پذیرفته شوند، کنترل می کند. مقدار آن به طور پیش فرض * می باشد.
activeClass	این گزینه یک String می باشد که یک یا چند گروه CSS را نمایش می دهد که هنگام جابجایی یک عنصر پذیرفته شده، به عنصر droppable اضافه می شوند. مقدار پیش فرض آن false می باشد.
addClasses	وقتی این گزینه روی false تنظیم شده باشد، از اضافه شدن گروه <i>ui-droppable</i> به عناصر droppable جلوگیری می کند. مقدار پیش فرض آن true می باشد.
disabled	وقتی این گزینه روی true تنظیم شده باشد، قابلیت دراپ شدن (droppable) را غیرفعال خواهد کرد. مقدار پیش فرض آن false می باشد.
greedy	این گزینه زمانی استفاده می شود که لازم است عناصر قابل جابه جایی را کنترل کنید که قرار است برای قرار گرفتن در droppable های تو در تو پذیرفته شوند. مقدار پیش فرض آن false می باشد. اگر این گزینه روی true تنظیم شده باشد، هر منبعی از

	droppable عنصر را دریافت نخواهد کرد.
hoverClass	این گزینه یک String می باشد که یک یا چند گروه CSS را نمایش می دهد که به عنصر droppable اضافه می شوند و یک عنصر پذیرفته شده وارد آن می شود. مقدار پیش فرض آن false می باشد.
scope	این گزینه برای محدود کردن عمل droppable از عناصر قابل جابه جایی فقط برای آیتم هایی که دارای <i>options.scope</i> مشابه هستند، استفاده می شود. مقدار پیش فرض آن "default" می باشد.
tolerance	این گزینه یک String می باشد که تعیین می کند کدام وضعیت برای امتحان کردن اینکه آیا یک draggable روی یک droppable معلق است، استفاده شود. مقدار پیش فرض آن "intersect" می باشد.

بخش زیر چند مثال کاربردی از قابلیت دراپ کردن نشان خواهد داد.

قابلیت پیش فرض:

مثال زیر یک قابلیت ساده ی **droppable** را توضیح می دهد که هیچ پارامتری به **droppable()** انتقال نمی دهد.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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>
    #draggable-1 {
      width: 100px;
      height: 50px;
      padding: 0.5em;
      float: left;
      margin: 22px 5px 10px 0;
    }
  </style>

```

```

#droppable-1 {
  width: 120px;
  height: 90px;
  padding: 0.5em;
  float: left;
  margin: 10px;
}
</style>
<script>
$(function () {
  $("#draggable-1").draggable();
  $("#droppable-1").droppable();
});
</script>
</head>
<body>
<div id="draggable-1" class="ui-widget-content">
  <p>Drag me to my target</p>
</div>
<div id="droppable-1" class="ui-widget-header">
  <p>Drop here</p>
</div>
</body>
</html>

```

استفاده از `addClass`، `disabled` و `tolerance`:

مثال زیر استفاده از سه گزینه ی **(a) disabled**، **(b) addClass** و **(c) tolerance** را عملکرد دراپ کردن

**JqueryUI** نشان می دهد.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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>
</head>
<body>
  <div id="draggable-2" style="width: 100px; height: 50px; padding: 0.5em; margin: 0px 5px 10px 0;">
    Drag me to my target
  </div>
  <div id="droppable-2" style="width: 120px; height: 90px; padding: 0.5em; margin: 10px 0 0 0;">
    Drop here
  </div>
  <div id="droppable-3" style="width: 120px; height: 90px; padding: 0.5em; margin: 10px 0 0 0;">
    Drop here
  </div>
  <div id="droppable-4" style="width: 120px; height: 90px; padding: 0.5em; margin: 10px 0 0 0;">
    Drop here
  </div>
  <div id="droppable-5" style="width: 120px; height: 90px; padding: 0.5em; margin: 10px 0 0 0;">
    Drop here
  </div>
</body>
</html>

```

```

    float: left;
    margin: 10px;
  }
</style>
<script>
$(function () {
  $("#draggable-2").draggable();
  $("#droppable-2").droppable({
    drop: function (event, ui) {
      $(this)
        .addClass("ui-state-highlight")
        .find("p")
        .html("Dropped!");
    }
  });
  $("#droppable-3").droppable({
    disabled: "true",
    drop: function (event, ui) {
      $(this)
        .addClass("ui-state-highlight")
        .find("p")
        .html("Dropped!");
    }
  });
  $("#droppable-4").droppable({
    tolerance: 'touch',
    drop: function (event, ui) {
      $(this)
        .addClass("ui-state-highlight")
        .find("p")
        .html("Dropped with a touch!");
    }
  });
  $("#droppable-5").droppable({
    tolerance: 'fit',
    drop: function (event, ui) {
      $(this)
        .addClass("ui-state-highlight")
        .find("p")
        .html("Dropped only when fully fit on the me!");
    }
  });
});
</script>
</head>
<body>
<div id="draggable-2" class="ui-widget-content">
  <p>Drag me to my target</p>
</div>
<div id="droppable-2" class="ui-widget-header">
  <p>Drop here</p>
</div>
<div id="droppable-3" class="ui-widget-header">
  <p>I'm disabled, you can't drop here!</p>
</div>
<div id="droppable-4" class="ui-widget-header">
  <p>Tolerance Touch!</p>

```

```

</div>
<div id="droppable-5" class="ui-widget-header">
  <p>Tolerance Fit!</p>
</div>
</body>
</html>

```

اکنون عنصر را روی باکس **"Tolerance Touch!"** قرار دهید ( فقط در کناره ی باکس حرکت کنید) و تغییرات عنصر هدف را مشاهده کنید. اکنون برای قرار دادن عنصر روی هدف **"Tolerance Fit!"**، عنصر قابل جابجایی باید به طور کامل عنصر هدف را تحت پوشش قرار دهد، به عنوان مثال **"Tolerance Fit!"**.

عناصر را انتخاب کنید تا دراپ شوند:

مثال زیر استفاده از گزینه های **accept** و **scope** را در عملکرد درگ کردن **jQueryUI** نشان می دهد.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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>
  .wrap {
    display: table-row-group;
  }

  #japanpeople, #indiapeople, #javatutorial, #springtutorial {
    width: 120px;
    height: 70px;
    padding: 0.5em;
    float: left;
    margin: 0px 5px 10px 0;
  }

  #japan, #india, #java, #spring {
    width: 140px;
    height: 100px;
    padding: 0.5em;
    float: left;
    margin: 10px;
  }
</style>
<script>
$(function () {
  $("#japanpeople").draggable();
  $("#indiapeople").draggable();

```

```

$( "#japan" ).droppable({
  accept: "#japanpeople",
  drop: function (event, ui) {
    $(this)
      .addClass("ui-state-highlight")
      .find("p")
      .html("Dropped!");
  }
});
$( "#india" ).droppable({
  accept: "#indiapeople",
  drop: function (event, ui) {
    $(this)
      .addClass("ui-state-highlight")
      .find("p")
      .html("Dropped!");
  }
});

$( "#javatutorial" ).draggable({ scope: "java" });
$( "#springtutorial" ).draggable({ scope: "spring" });
$( "#java" ).droppable({
  scope: "java",
  drop: function (event, ui) {
    $(this)
      .addClass("ui-state-highlight")
      .find("p")
      .html("Dropped!");
  }
});
$( "#spring" ).droppable({
  scope: "spring",
  drop: function (event, ui) {
    $(this)
      .addClass("ui-state-highlight")
      .find("p")
      .html("Dropped!");
  }
});
});
</script>
</head>
<body>
<div class="wrap">
<div id="japanpeople" class="ui-widget-content">
<p>People to be dropped to Japan</p>
</div>

<div id="indiapeople" class="ui-widget-content">
<p>People to be dropped to India</p>
</div>

<div id="japan" class="ui-widget-header">
<p>Japan</p>
</div>

<div id="india" class="ui-widget-header">

```

```

    <p>India</p>
  </div>
</div>
<hr />
<div class="wrap">
  <div id="javatutorial" class="ui-widget-content">
    <p>People who want to learn Java</p>
  </div>
  <div id="springtutorial" class="ui-widget-content">
    <p>People who want to learn Spring</p>
  </div>
  <div id="java" class="ui-widget-header">
    <p>Java</p>
  </div>

  <div id="spring" class="ui-widget-header">
    <p>Spring</p>
  </div>
</div>
</body>
</html>

```

در اینجا مشاهده می کنید که می توانید عنصر "People from Japan" را فقط روی هدف "Japan" و عنصر "People from India" را روی هدف "India" قرار دهید. به طور مشابه محدوده ی مربوط به "People who want to learn Java" روی هدف "Java" و "People who want to learn Spring" روی هدف "Spring" target تنظیم می شود.

### مدیریت ظاهر

مثال زیر استفاده از گزینه های **activeClass** و **hoverClass** از گروه **jQueryUI** را توضیح می دهد، که در تنظیم ظاهر به ما کمک می کند.

```

<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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 type="text/css">
    #draggable-3 {
      width: 100px;
      height: 50px;
      padding: 0.5em;
      float: left;
      margin: 21px 5px 10px 0;
    }
  </style>

```

```

}

#droppable-6 {
  width: 120px;
  height: 90px;
  padding: 0.5em;
  float: left;
  margin: 10px;
}

.active {
  border-color: blue;
  background: grey;
}

.hover {
  border-color: red;
  background: green;
}
</style>
<script>
$(function () {
  $("#draggable-3").draggable();
  $("#droppable-6").droppable({
    activeClass: "active",
    hoverClass: "hover",
    drop: function (event, ui) {
      $(this)
        .addClass("ui-state-highlight")
        .find("p")
        .html("Dropped!");
    }
  });
});
</script>
</head>
<body>

<div id="draggable-3" class="ui-widget-content">
  <p>Drag me to my target</p>
</div>

<div id="droppable-6" class="ui-widget-header">
  <p>Drop here</p>
</div>
</body>
</html>

```

شما متوجه می شوید که در جابه جایی و یا تعلیق عنصر "Drag me to my target" رنگ عنصر "Drop here" را تغییر می دهد.

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

متد `droppable ("action", params)` می تواند عملی را روی عناصر `droppable` اجرا کند، مانند جلوگیری از قابلیت `droppable`. این عمل به عنوان یک رشته در اولین برهان تعیین شده است ( به عنوان مثال "disable" مانع دراپ شدن می شود). در جدول زیر فعالیت هایی که منتقل می شوند را چک کنید.

ترکیب

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

جدول زیر فعالیت های مختلفی را لیست می کند که در با این متد مورد استفاده قرار می گیرند:

Action	Description
destroy	این عملکرد قابلیت droppable مربوط به یک عنصر را کاملا از بین می برد. این عناصر به وضعیت pre-init خود باز می گردند.
disable	این عملیات droppable را لغو می کند. عناصر دیگر عناصر droppable نیستند. این متد هیچ برهانی نمی پذیرد.
enable	این عملیات مجددا عملکرد droppable را فعال سازی می کند و عناصر مجددا می توانند عنصر droppable دریافت کنند. این متد هیچ برهانی را نمی پذیرد.
option	این فعالیت یک آبجکت حاوی جفت های key/value را دریافت می کند که نشان دهنده ی گزینه های hash فعلی دارای قابلیت droppable می باشد. این متد

	هیچ برهانی را نمی پذیرد.
<code>option( optionName )</code>	این فعالیت مقداری که معمولا عنصر <code>droppable</code> را با <code>optionName</code> تعیین شده مرتبط می سازد، دریافت می کند و مقدار <code>String</code> را به عنوان یک برهان می گیرد.
<code>option( optionName, value )</code>	این فعالیت مقدار گزینه ی <code>droppable</code> مرتبط با <code>optionName</code> تعیین شده را تنظیم می کند.
<code>option( options )</code>	این عمل یک یا چند گزینه برای <code>droppable</code> تنظیم می کند. گزینه های برهان نقشه ای از جفت های <code>option-value</code> می باشد که باید تنظیم شوند.
<code>widget</code>	این عمل یک آبجکت <code>jQuery</code> حاوی عنصر <code>droppable</code> یازمی گرداند. این متد هیچ برهانی را نمی پذیرد.

## مثال

اکنون اجازه بدهید با استفاده از عملکردهای ارائه شده در جدول بالا، مثالی را به شما توضیح دهیم. مثال زیر استفاده از متد `destroy()` را توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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>
.droppable-4 {
  width: 90px;
  height: 50px;
  padding: 0.5em;
  float: left;
  margin: 0px 5px 10px 0;
  border: 1px solid red;
}
```

```

    background-color: #B9CD6D;
}

.droppable-7 {
    width: 100px;
    height: 90px;
    padding: 0.5em;
    float: left;
    margin: 10px;
    border: 1px solid black;
    background-color: #A39480;
}

.droppable.active {
    background-color: red;
}
</style>
<script>
$(function () {
    $('.draggable-4').draggable({ revert: true });
    $('.droppable-7').droppable({
        hoverClass: 'active',
        drop: function (e, ui) {
            $(this).html(ui.draggable.remove().html());
            $(this).droppable('destroy');
            $(this)
                .addClass("ui-state-highlight")
                .find("p")
                .html("i'm destroyed!");
        }
    });
});
</script>
</head>
<body>
<div class="draggable-4"><p>drag 1</p></div>
<div class="draggable-4"><p>drag 2</p></div>
<div class="draggable-4"><p>drag 3</p></div>

<div style="clear: both;padding:10px"></div>

<div class="droppable-7">drop here</div>
<div class="droppable-7">drop here</div>
<div class="droppable-7">drop here</div>
</body>
</html>

```

اگر "drag1" را روی رکدام از عناصر به نام "drop here" قرار دهید، متوجه خواهید شد که این عنصر دراپ شده و این عمل قابلیت یک عنصر را به طور کامل از بین می برد. شما می توانید "drag2" و "drag3" را مجدداً روی این عنصر قرار دهید.

## مدیریت رویداد روی عناصر `droppable`:

علاوه بر متد `droppable (options)` که در بخش قبل مشاهده کردیم، **jQueryUI** متدهای رویدادی ارائه می دهد که برای یک رویداد خاص آغاز می شوند. این متدها در جدول زیر لیست شده اند:

Event Method	Description
<code>activate(event, ui)</code>	این رویداد وقتی آغاز می شود که عنصر قابل جابه جایی پذیرفته شده جابه جایی را شروع کند. اگر بخواهید <code>droppable</code> را دارای ویژگی <code>light up</code> کنید، این متد می تواند مفید باشد.
<code>create(event, ui)</code>	این رویداد وقتی آغاز می شود که یک عنصر <code>droppable</code> ایجاد شده باشد، که رویداد از نوع <code>Event</code> است و <code>ui</code> از نوع <code>object</code> .
<code>deactivate(event, ui)</code>	این رویداد وقتی آغاز می شود که یک عنصر قابل جابه جایی، جابه جایی را متوقف کند که رویداد از نوع <code>Event</code> است و <code>ui</code> از نوع <code>object</code> .
<code>drop(event, ui)</code>	این عمل وقتی آغاز می شود که یک المنت روی <code>droppable</code> دراپ شود که براساس گزینه <code>tolerance</code> می باشد که رویداد از نوع <code>Event</code> است و <code>ui</code> از نوع <code>object</code> .
<code>out(event, ui)</code>	این رویداد وقتی آغاز می شود که یک المنت قابل جابه جایی پذیرفته شده از <code>droppable</code> خارج شود و براساس گزینه <code>tolerance</code> می باشد و رویداد از نوع <code>Event</code> است و <code>ui</code> از نوع <code>object</code> .
<code>over(event, ui)</code>	این رویداد وقتی آغاز می شود که یک عنصر قابل جابه جایی پذیرفته شده روی <code>droppable</code> درگ می شود. این متد براساس گزینه <code>tolerance</code> می باشد و

## رویداد از نوع Event است و ui از نوع object

مثال

مثال زیر استفاده از متد **event** را در هنگام عملکرد دراپ توضیح می دهد. این مثال استفاده از رویدادهای **drop**، **over** و **out** را توضیح می دهد.

```
<!doctype html>
<html lang="en">
<head>
  <meta charset="utf-8">
  <title>jQuery UI Droppable - Default 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>
    #draggable-5 {
      width: 100px;
      height: 50px;
      padding: 0.5em;
      float: left;
      margin: 22px 5px 10px 0;
    }

    #droppable-8 {
      width: 120px;
      height: 90px;
      padding: 0.5em;
      float: left;
      margin: 10px;
    }
  </style>
  <script>
    $(function () {
      $("#draggable-5").draggable();
      $("#droppable-8").droppable({
        drop: function (event, ui) {
          $(this)
            .addClass("ui-state-highlight")
            .find("p")
            .html("Dropped!");
        },
        over: function (event, ui) {
          $(this)
            .addClass("ui-state-highlight")
            .find("p")
            .html("moving in!");
        },
        out: function (event, ui) {

```

```
$(this)
.addClass("ui-state-highlight")
.find("p")
.html("moving out!");
}
});
});
</script>
</head>
<body>
<div id="draggable-5" class="ui-widget-content">
<p>Drag me to my target</p>
</div>
<div id="droppable-8" class="ui-widget-header">
<p>Drop here</p>
</div>
</body>
</html>
```

در اینجا متوجه خواهید شد که هنگامی که عنصر را درگ می کنید، چگونه پیغام در یک عنصر **droppable** تغییر می کند.