

بسم الله الرحمن الرحيم

آموزشگاه تحلیل داده

تخصصی ترین مرکز برنامه نویسی و دیتابیس در ایران

قابل جابجایی

مدرس : مهندس افشین رفوآ

قابل جابجایی

jQueryUI متد **draggable()** را برای ساخت عنصر قابل جابجایی **DOM** ارائه می دهد. زمانیکه عنصر قابل جابجایی می باشد، می توانید با کلیک کردن روی بخش و انتقال آن در هر جایی در محدوده ی بخش ویو، آن را جابجا کنید.

ترکیب:

متد **draggable()** به دو فرم زیر مورد استفاده قرار می گیرد:

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

`$(selector, context).draggable ("action", [params]) Method`

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

متد **draggable (options)** اعلام می کند که یک عنصر **HTML** می تواند در صفحه ی **HTML** جابجا شود. پارامتر **options** در واقع آبجکتی می باشد که رفتار عناصر را تعیین می کند.

ترکیب:

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

شما می توانید با استفاده از آبجکت **Javascript** یک یا چند گزینه را همزمان ارائه دهید. اگر بیشتر از یک گزینه برای ارائه وجود دارد، می توانید آنها را با استفاده از کاما (ویرگول) از یکدیگر مجزا کنید، مانند زیر:

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

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

Option	Description
addClasses	اگر این گزینه روی <code>false</code> تنظیم شده است، مانع اضافه شدن گروه <code>ui-draggable</code> به لیست عناصر انتخاب شده ی <code>DOM</code> می شود. به طور پیش فرض این گزینه روی <code>true</code> تنظیم شده است.
appendTo	عنصری را مشخص می کند که در آن کمک کننده ی قابل جابجایی در هنگام درگ کردن، باید ضمیمه شود. مقدار این گزینه به طور پیش فرض <code>"parent"</code> می باشد.
axis	این گزینه درگ کردن به صورت افقی (<code>X</code>) و یا عمودی (<code>Y</code>) را تعیین می کند. مقادیر ممکن <code>"x", "y"</code> : می باشند.
cancel	شما می توانید از این گزینه برای جلوگیری از درگ کردن بر روی برخی عناصر استفاده کنید. به طور پیش فرض مقدار مربوطه آن گزینه ی <code>"input,textarea, button,select,option"</code> می باشد.
connectToSortable	شما می توانید از این گزینه برای تعیین لیستی استفاده کنید که عناصر آنها قابل تعویض می باشند. در انتهای جایگزینی، عنصر بخشی از لیست می باشد. مقدار آن به طور پیش فرض <code>"false"</code> می باشد.
containment	جابجایی را به داخل محدوده ی عنصر یا ناحیه ی تعیین شده محدود می کند. مقدار آن به طور پیش فرض <code>"false"</code> می باشد.
cursor	زمانی که عنصر جابجا می شود، پراپرتی <code>CSS</code> مکان نما را تعیین می کند. این گزینه در واقع شکل نشانگر ماوس را نمایش می دهد. مقدار آن به طور پیش فرض <code>"auto"</code> می باشد.

cursorAt	افست مربوط به درگ کمکی مر بوط به نشانگر ماوس تنظیم می کند. با استفاده از ترکیبی از یک یا دو کلید، مختصات می تواند به عنوان یک hash ارائه شود.
delay	پس از اینکه اولین جابجایی ماوس انجام می شود، تاخیری در حدود هزارم ثانیه دارد. تغییر مکان ممکن است پس از این زمان آغاز شود. مقدار آن به طور پیش فرض "0" می باشد.
disabled	این گزینه وقتی روی true تنظیم باشد، توانایی جابجایی آیتم ها را غیرفعال می سازد. آیتم ها نمی توانند حرکت کنند تا زمانیکه عملکرد فعال شده باشد (با استفاده از عملکرد ("enable" draggable) مقدار آن به طور پیش فرض "false" می باشد.
distance	قبل از جابجایی، تعداد پیکسل هایی که ماوس باید جابجا شود، در نظر گرفته می شوند. مقدار آن به طور پیش فرض "1" می باشد.
grid	جابجایی کمکی را در هر پیکسل X و y به یک شبکه آغاز می کند. ردیف مربوطه باید به شکل [X, y] باشد. مقدار آن به طور پیش فرض "false" می باشد.
handle	اگر مشخص شده باشد، جابجایی از آغاز را محدود می کند، مگر اینکه mousedown روی عناصر خاص اتفاق بیفتد. مقدار آن به طور پیش فرض "false" می باشد.
helper	به یک عنصر کمکی اجازه می دهد تا برای درگ کردن صفحه نمایش استفاده شود. مقدار آن به طور پیش فرض "original" می باشد.
iframeFix	در طول یک درگ، مانع iframes ها در گرفتن رویدادهای mousemove می شود. مقدار آن به طور پیش فرض "false" می باشد.
opacity	در هنگام جابجایی حالت محوی از عنصر حرکت می کند. مقدار آن به طور پیش فرض "false" می باشد.

refreshPositions	اگر روی true تنظیم شده باشد، همه ی موقعیت ها با هر حرکت ماوس محاسبه می شوند. به طور پیش فرض مقدار آن "false" می باشد.
revert	نشان می دهد که آیا در انتهای جابه جایی عنصر به عقب و موقعیت اصلی خود بازمی گردد. مقدار آن به طور پیش فرض "false" می باشد.
revertDuration	طول دوره ی جابه جایی (به هزارم ثانیه) پس از اینکه عنصر به موقعیت اصلی خود بازگشت. مقدار آن به طور پیش فرض "500" می باشد.
scope	علاوه بر پذیرش گزینه ی droppable، برای گروه بندی آیتم های draggable و droppable استفاده می شود. به طور پیش فرض مقدار آن "default" می باشد.
scroll	وقتی روی true تنظیم باشد، اگر آیتم از محدوده ی قابل مشاهده ی پنجره خارج شود، صفحه اسکرول خواهد شد. مقدار پیش فرض آن "true" می باشد.
scrollSensitivity	تعداد پیکسل هایی را نشان می دهد که ماوس باید از پنجره خارج شود تا باعث اسکرول شدن صفحه شود. مقدار آن به طور پیش فرض "20" می باشد.
scrollSpeed	Indicates the scrolling speed of the display once scrolling begins. By default its value is "20". سرعت اسکرول کردن صفحه نمایش را هنگام آغاز آن نشان می دهد. مقدار آن به طور پیش فرض "20" می باشد.
snap	نمایش آیتمی را که روی دیگر عناصر جابجا شده، تنظیم می کند. مقدار پیش فرض آن "false" می باشد.
snapMode	تعیین می کند که چگونه بین عنصر جابجا شده و موارد نمایش داده شده در <i>options.snap</i> تنظیم برقرار شود. مقدار پیش فرض آن "both" می باشد.

snapTolerance	تعداد پیکسل ها در موقعیت های مختلف را که برای انتشار تنظیمات لازم هستند، به حداکثر می رساند. مقدار پیش فرض آن "20" می باشد.
stack	Z-index مربوط به مجموعه عناصر را کنترل می کند که با انتخابگر مطابقت دارد، همیشه آخرین آیتم درگ شده را جلو می آورد. در برنامه هایی مانند window managers بسیار موثر است. مقدار پیش فرض آن "false" می باشد.
zIndex	Z-index برای کمکی در هنگام درگ شدن. مقدار پیش فرض آن "false" می باشد.

بخش زیر چند مثال کارآمد از عملکرد درگ نشان می دهد.

عملکرد پیش فرض:

در مثالی را مشاهده می کنید که یک عملکرد قابل جابجایی را توضیح می دهد که هیچ پارامتری به متد

`draggable()` انتقال نمی دهد.

```
<!DOCTYPE html>
<head>
  <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 {
      width: 150px;
      height: 150px;
      padding: 0.5em;
      background: #eee;
    }
  </style>
  <script>
    $(function () {
      $("#draggable").draggable();
    });
  </script>
</head>
<body>
  <div id="draggable" class="ui-widget-content">
    <p>Drag me !!!</p>
  </div>
</body>
```

</html>

اجازه بدهید کد بالا را در یک فایل HTML به نام `dragexample.htm` ذخیره کنیم و آن را در یک مرورگر استاندارد که `Javascript` را پشتیبانی می کند، باز کنیم.

استفاده از `disable`، `distance` و `delay`:

مثال زیر استفاده از سه گزینه ی مهم `delay (b) disabled (a)` و `distance (c)` را در عملکرد درگ از `JqueryUI` نشان می دهد.

```
<!DOCTYPE html>
<head>
  <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="div1" style="border:solid 1px;background-color:gainsboro;">
    <span>You can't move me!</span><br /><br />
  </div>
  <div id="div2" style="border:solid 1px;background-color:grey;">
    <span>
      Dragging will start only after you drag me for 50px
    </span>
    <br /><br />
  </div>
  <div id="div3" style="border:solid 1px;background-color:gainsboro;">
    <span>
      You have to wait for 500ms for dragging to start!
    </span>
    <br /><br />
  </div>
  <script>
    $("#div1 span").draggable(
      { disabled: true }
    );
    $("#div2 span").draggable(
      { distance: 50 }
    );
    $("#div3 span").draggable(
      { delay: 500 }
    );
  </script>
</body>
</html>
```

اجزه بدهید کد بالا را در یک فایل **HTML** به نام **dragexample.htm** ذخیره کرده و آن را در یک مرورگر استاندارد که **javascript** را پشتیبانی می کند، باز کنیم.

جنبش محدود

مثال های زیر چگونگی محدود کردن محیط عناصر را روی صفحه با استفاده از گزینه ی **containment** در عملکرد درگ کردن **JqueryUI** نشان می دهد.

```
<!DOCTYPE html>
<head>
  <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="div4" style="border:solid 1px;background-color:gainsboro;">
    <span>You can drag me only within this div.</span><br /><br />
  </div>
  <div id="div5" style="border:solid 1px;background-color:grey;">
    <span>You can drag me only along x axis.</span><br /><br />
  </div>
  <script>
    $("#div4 span").draggable({
      containment: "#div4"
    });
    $("#div5 span").draggable({
      axis: "x"
    });
  </script>
</body>
</html>
```

اجازه بدهید کد بالا را در یک فایل **HTML** به نام **dragexample.htm** ذخیره کرده و آن را در یک مرورگر استاندارد که **javascript** را پشتیبانی می کند، باز کنیم

در اینجا از خارج شدن عناصر **** از یک **<div>** که ID آن **div4** می باشد، جلوگیری می شود. همچنین می توانید با استفاده از گزینه های محور، با ارزش **x** یا **y**، که توضیح داده شده اند، محدودیت هایی را اعمال کنید.

جابجایی محتوا با استفاده از تکثیر

مثال زیر چگونگی جابجایی یک آیتم را که کلون (**clone**) مربوط به آیتم انتخاب شده می باشد، نشان می دهد. این کار با استفاده از گزینه ی **helper** با مقدار **clone** انجام می شود.

```
<!DOCTYPE html>
<head>
  <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="div6" style="border:solid 1px;background:#eee;height:50px;">
    <span>You can duplicate me....</span>
  </div>
  <script>
    $("#div6 span").draggable({
      helper: "clone"
    });
  </script>
</body>
</html>
```

اجازه بدهید کد بالا را در یک فایل **HTML** به نام **dragexample.htm** ذخیره کرده و آن را در یک مرورگر استاندارد که **javascript** را پشتیبانی می کند، باز کنیم.

همانطور که مشاهده می کنید اولین عنصر که درگ می شود – تنها عنصر کلون منتقل شده است، در حالیکه آیتم اصلی در موقعیت اصلی خود باقی می ماند. اگر ماوس را رها کنید، عنصر کلون ناپدید شده و آیتم اصلی هنوز به جای خود باقیست.

دریافت مقدار گزینه ی فعلی

مثال زیر چگونگی دریافت مقدار مربوط به هر گزینه را در هر زمانی، در طول دوره ی اجرای نسخه ی شما توضیح می دهد. در اینجا مقدار گزینه های **cursor** و **cursorAt** را که در زمان اجرا تنظیم شده اند، خواهیم خواند. به روش مشابه می توانید مقدار گزینه های موجود دیگر را دریافت کنید.

```
<!DOCTYPE html>
<head>
  <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="divX" style="border:solid 1px;background:#eee;height:50px;">
    <span>Click anywhere on me to see cursor type...</span>
  </div>
```



```

<script>
  /* First make the item draggable */
  $("#divX span").draggable();

  $("#divX span").bind('click', function (event) {
    var cursor = $("#divX span").draggable("option", "cursor");
    var cursorAt = $("#divX span").draggable("option", "cursorAt");
    alert("Cursor type - " + cursor + ", cursorAt - " + cursorAt);
  });
</script>
</body>
</html>

```

اجازه بدهید کد بالا را در یک فایل **HTML** به نام **dragexample.htm** ذخیره کرده و آن را در یک مرورگر استاندارد که **javascript** را پشتیبانی می کند، باز کنیم.

متد [\\$\(selector, context\).draggable \("action", \[params\]\)](#)

متد **draggable (action, params)** می تواند عملکردی را روی عناصر قابل حرکت اجرا کند، به عنوان مثال جلوگیری از جابجایی. **Action** به عنوان یک رشته در اولین استدلال تعیین شده و یک یا چند **params** می توانند بر اساس عملکرد ارائه شده، تعیین شوند.

اساسا در اینجا عملکردها چیزی به جز متدهای **jQuery** نیستند که می توانیم از آنها به فرم رشته استفاده کنیم.

ترکیب

```
$(selector, context).draggable("action", [params]);
```

جدول زیر عملکردهایی را برای این متد لیست می کند:

Action	Description
--------	-------------

destroy()	قابلیت جابجایی را کاملا حذف می کند. عناصر قابل جابه جایی نیستند. این امر عنصر را به وضعیت pre-init خود بازمی گرداند.
disable()	قابلیت جابجایی را غیرفعال می سازد. عناصر نمی توانند جابجا شوند تا زمانیکه فراخوانی draggable("enable") به متد انجام شود.
enable()	مدیریت جابه جایی را دوباره فعال می سازد. عناصر می توانند مجددا جابجا شوند.
option(optionName)	مقدار کنونی مربوط به <i>optionName</i> تعیین شده را دریافت می کند. در اینجا <i>optionName</i> نام گزینه ای برای دریافت است و از نوع String می باشد.
option()	آبجکتی حاوی جفت های key/value را دریافت می کند که گزینه های قابل جابجایی کنونی را نشان می دهد.
option(optionName, value)	مقدار گزینه ی draggable را تنظیم می کند که با <i>optionName</i> تعیین شده در ارتباط می باشد. در اینجا <i>optionName</i> نام گزینه ای برای تنظیم می باشد و <i>value</i> نام مقداری است که برای این گزینه تنظیم می شود.
option(options)	یک یا چند گزینه را برای draggable تنظیم می کند. در اینجا options در واقع نقشه ای از جفت های option-value برای تنظیم می باشد.
widget()	یک آبجکت jQuery را که حاوی عنصر draggable می باشد، باز می گرداند.

مثال ها

اکنون اجازه بدهید مثالی را مشاهده کنیم که از عملکردهای موجود در جدول بالا استفاده می کند. مثال زیر استفاده از عملکردهای **enable** و **disable** را توضیح می دهد:

```
<!DOCTYPE html>
<head>
```

آدرس آموزشگاه : تهران - خیابان شریعتی - بالا تر از خیابان ملک - جنب بانک صادرات - پلاک 561 - واحد 7
88146323 - 88446780 - 88146330

```

<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="div7" style="border:solid 1px;background-color:gainsboro;">
  <span>You can't move me. Dragging is disabled.</span><br><br>
</div>
<div id="div8" style="border:solid 1px;background-color:grey;">
  <span>You can move me. Dragging is enabled.</span><br><br>
</div>
<script>
$( "#div7 span" ).draggable();
$( "#div7 span" ).draggable('disable');
$( "#div8 span" ).draggable();
$( "#div8 span" ).draggable('enable');
</script>
</body>
</html>

```

اجازه بدهید کد بالا را در یک فایل **HTML** به نام **dragexample.htm** ذخیره کرده و آن را در یک مرورگر استاندارد که **javascript** را پشتیبانی می کند، باز کنیم.

همانطور که مشاهده می کنید، جابجایی اولین عنصر غیر فعال و جابجایی دومین عنصر فعال می باشد که می توانید جابجایی آن را امتحان کنید.

مدیریت رویداد روی عناصر جابجا شده:

علاوه بر متد **(options) draggable**، در بخش های قبل مشاهده کردیم که **JqueryUI** متدهایی ارائه می دهد که برای یک رویداد خاص فعال می شدند. این متدها در جدول زیر لیست شده اند:

Event Method	Description
create(event, ui)	زمانی که draggable ایجاد می شود، آغاز به کار می کند. در اینجا رویداد از نوع Event بوده و آبجکت نیز از نوع ui می باشد.
drag(event, ui)	هنگام حرکت ماوس در طول جابه جایی، آغاز به کار می کند، در اینجا رویداد از نوع Event بوده و آبجکت نیز از نوع ui می باشد، مانند helper ، position و object .

start(event, ui)	در هنگام شروع جابه جایی آغاز به کار می کند. در اینجا رویداد از نوع Event و ui از نوع object می باشد، مانند helper، position و object.
stop(event, ui)	در هنگام توقف جابه جایی آغاز به کار می کند. در اینجا رویداد از نوع Event و ui از نوع object می باشد، مانند helper، position و object.

مثال ها

مثال زیر استفاده از متدی را در طول درگ کردن توضیح می دهد. این مثال استفاده از رویداد **drag** را بیان می کند.

```
<!DOCTYPE html>
<head>
  <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="div9" style="border:solid 1px;background-color:gainsboro;">
    <span>Drag me to check the event method firing</span><br /><br />
  </div>
  <script>
    $("#div9 span").draggable({
      cursor: "move",
      axis: "x",
      drag: function (event, ui) {
        alert("hi..");
      }
    });
  </script>
</body>
</html>
```

اجازه بدهید کد بالا را در فایل **HTML** به نام **dragexample.htm** ذخیره کنیم و آن را در مرورگر استاندارد که **Javascript** را پشتیبانی می کند، باز کنیم.

اکنون سعی کنید که محتوای نوشته شده را درگ کنید، مشاهده خواهید کرد که شروع یک رویداد درگ منجر به نمایش یک دیالوگ باکس می شود و مکان نما برای جابجایی آیکن، تغییر خواهد کرد و متن هم فقط در محور **X** حرکت خواهد کرد.

www.tahlildadeh.com