**Neo Rays Software Solutions Pvt. Ltd**



jQuery UI

|  |  |
| --- | --- |
| Created By | Meheboob Nadaf |
| Prepared For | Neo Rays Software Solutions Pvt. Ltd. |
| Version | 0.1 |
| Created On | 29 December, 2015 |
| Last Updated On |  |

Contents

# 

# Version Control

Completing the following table makes it easy to come back later and track what changes were made to the requirements at each point in the project, who made them, and why they were made. This is a way of implementing change control on the BRD.

## Revision History

|  |  |  |  |
| --- | --- | --- | --- |
| **Version #** | **Date** | **Responsibility (Author)** | **Description** |
| 0.1 | December 15th , 2015 | Meheboob Nadaf |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

# Acronyms and Glossary

The following table includes definitions for any unique symbols or notations that are used in the document.

|  |  |
| --- | --- |
| Term | Definition |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |

Table 1: Acronyms and Glossary

# jQuery UI

## What is jQuery UI

* Good for highly interactive web applications
* Open source and free to use
* Powerful theme mechanism
* Stable and maintenance friendly
* Extensive browser support

## jQuery UI Example

<!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: 100px; height: 100px; padding: 0.5em; background:#7fffd4;}

</style>

<script>

$(function() {

$( "#draggable" ).draggable();

});

</script>

</head>

<body>

<div id="draggable" class="ui-widget-content">

<p>I am draggable!</p>

</div>

</body>

</html>

# jQuery UI Introduction

* jQueryUI stands for jQuery User Interface. It is a collection of animated visual effects, GUI widgets, and themes implemented with jQuery, CSS, HTML and JavaScript. These new plug-ins add a lot of new functionalities in the jQuery core library.
* It is a very popular and powerful mobile first front-end framework used for faster and easier web development. According to a survey it is used on over 176000 websites, making it the second most popular JavaScript library on the Web.

### jQuery UI History

* jQueryUI is a free and open source software first published and released in September 2007. It is distributed by jQuery foundation under the MIT license.

### jQuery UI Release History

* jQuery was first launched on Sep, 17, 2007. After that, there have been many versions released with additional functionalities.

# jQuery UI Features

* jQueryUI facilitates you to make highly interactive web applications.
* It is open source and free to use.
* It provides a powerful theme mechanism.
* It is very stable and maintenance friendly.
* It provides an extensive browser support
* Good documentation

## jQuery UI Categorization

We can categorize the jQueryUI into four groups.

1. Interactions
2. Widgets
3. Effects
4. Utilities
5. **Interactions:** Interactions are the set of plug-ins which facilitates users to interact with DOM elements. These are the mostly used interactions:

* Draggable
* Droppable
* Resizable
* Selectable
* Sortable

1. **Widgets:** Widgets are the jQuery plug-ins which makes you able to create user interface elements like date picker, progress bar etc. These are the mostly used widgets:

* Accordion
* Autocomplete
* Dialog
* Button
* Date Picker
* Menu
* Progress Bar
* Tabs
* Tooltip
* Slider
* Spinner

1. **Effects:**The internal jQuery effects contain a full suite of custom animation and transition for DOM elements.

* Hide
* Show
* Add Class
* Remove Class
* Switch Class
* Toggle Class
* Color Animation
* Effect
* Toggle

1. **Utilities:**Utilities are the modular tools, used by jQuery library internally.

* **Position:**It is used to set the position of the element according to the other element's alignment (position).

# jQuery UI Draggable

* jQuery UI draggable() method is used to make any DOM element draggable. Once the element is made draggable, you can move it by clicking on it with the mouse and drag it anywhere within the viewport.

**Syntax:**

You can use the draggable () method in two forms:

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

2.  $(selector, context).draggable ("action", params) Method

## First Method

The draggable (option) method specifies that an HTML element can be moved in the HTML page. Here, the option parameter specifies the behavior of the elements involved.

**Syntax:**

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

You can use one or more options at a time using JavaScript object. In the case of multiple options, you should use commas to separate them. For example:

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

Following is a list of different options that can be used with this method:

|  |  |  |
| --- | --- | --- |
| **Option** | | **Description** |
| addclasses | If this option is set to false, it will prevent the UI-draggable class from being added in the list of selected DOM elements. By default its value is true. | |
| appendto | It specifies the element in which the draggable helper should be appended to while dragging. By default its value is "parent". | |
| axis | This option constrains dragging to either the horizontal (x) or vertical (y) axis. Its possible values are:"x", "y". | |
| cancel | This option is used to prevent dragging from starting on specified elements. By default its value is "input, textarea, button, select, option". | |
| connecttosortable | This option is used to specify a list whose elements are interchangeable. At the end of placement, the element is part of the list. By default its value is "false". | |
| containment | Constrains dragging to within the bounds of the specified element or region. By default its value is "false". | |
| cursor | It is used to specify the CSS property of the cursor when the element moves. It represents the shape of the mouse pointer. By default its value is "auto". | |
| cursorat | It sets the offset of the dragging helper relative to the mouse cursor. Coordinates can be given as a hash using a combination of one or two keys: { top, left, right, bottom }. By default its value is "false". | |
| delay | It specifies the delay in milliseconds, after which the first movement of the mouse is taken into account. The displacement may begin after that time. By default its value is "0". | |
| disabled | It disables the ability to move items when set to true. Items cannot be moved until this function is enabled (using the draggable ("enable") instruction). By default its value is "false". | |
| distance | The number of pixels that the mouse must be moved before the displacement is taken into account. By default its value is "1". | |
| grid | It snaps the dragging helper to a grid, every x and y pixels. The array must be of the form [ x, y ]. By default its value is "false". | |
| handle | If specified, restricts dragging from starting unless the mousedown occurs on the specified element(s). By default its value is "false". | |
| helper | It allows for a helper element to be used for dragging display. By default its value is "original". | |
| iframefix | It prevents iframes from capturing the mousemove events during a drag. By default its value is "false". | |
| opacity | Opacity of the element moved when moving. By default its value is "false". | |
| refreshpositions | If set to true, all droppable positions are calculated on every mousemove. By default its value is "false". | |
| revert | It indicates whether the element is moved back to its original position at the end of the move. By default its value is "false". | |
| revertduration | It indicates the duration of displacement (in milliseconds) after which the element returns to its original position (see options.revert). By default its value is "500". | |
| scope | It is used to group sets of draggable and droppable items, in addition to droppable's accept option. By default its value is "default". | |
| scroll | when set to true (the default), the display will scroll if the item is moved outside the viewable area of the window. by default its value is "true". | |
| scrollsenstivity | It indicates how many pixels the mouse must exit the window to cause scrolling of the display. By default its value is "20". | |
| scrollspeed | It indicates the scrolling speed of the display once scrolling begins. By default its value is "20". | |
| snap | It adjusts the display of the item being moved on other elements (which are flown). By default its value is "false". | |
| snapmode | It specifies how the adjustment should be made between the moved element and those indicated in options.snap. By default its value is "both". | |
| snaptolerance | It specifies the maximum number of pixels in the difference in position necessary to establish the adjustment. By default its value is "20". | |
| stack | It controls the z-index of the set of elements that match the selector, always brings the currently dragged item to the front. It is very useful in things like window managers. By default its value is "false". | |
| zindex | z-index for the helper while being dragged. By default its value is "false". | |

## jQuery UI Draggable() method example 1

Let's take an example to demonstrate draggable functionality passing no parameters to the draggable() method.

<!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: 100px; height: 100px; padding: 0.5em; background:#7fffd4;}

</style>

<script>

$(function() {

$( "#draggable" ).draggable();

});

</script>

</head>

<body>

<div id="draggable" class="ui-widget-content">

<p>I am draggable!</p>

</div>

</body>

</html>

## Second Method

The draggable (action, params) method is used to perform an action like prevent displacement. Here action is specified as a string and one or more params can be provided based on the given action.

**Syntax:**

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

Following is a list of actions used with this method:

|  |  |
| --- | --- |
| **Action** | **Description** |
| destroy() | It is used to remove drag functionality completely. The elements are no longer movable. This will return the element back to its pre-init state. |
| disable() | It is used to disable drag functionality. Elements cannot be moved until the next call to the draggable("enable") method. |
| enable() | It is used to reactivates drag management. The elements can be moved again. |
| option(optionname) | It gets the value currently associated with the specified optionname. Here optionname is name of the option to get and is of type string. |
| option() | It gets an object containing key/value pairs representing the current draggable options hash. |
| option(optionname, value) | It sets the value of the draggable option associated with the specified optionname. Here optionname is the name of the option to set and value is the value to set for the option. |
| option(options) | It sets one or more options for the draggable. Here options is a map of option-value pairs to set. |
| widget() | It returns a jQuery object containing the draggable element. |

## jQuery UI Draggable() method example 5

Let's take an example to see the use of actions from the above table. The following example demonstrates the use of "disable" and "enable" action.

<!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="div7" style="border:solid 1px;background-color:pink;">

<span>Dragging is disabled. You can't move me!</span><br><br>

</div>

<div id="div8" style="border:solid 1px;background-color:yellow;">

<span>Dragging is enabled. Move Me!</span><br><br>

</div>

<script>

$("#div7 span").draggable ();

$("#div7 span").draggable ('disable');

$("#div8 span").draggable ();

$("#div8 span").draggable ('enable');

</script>

</body>

</html>

# jQuery UI Droppable

* jQuery UI facilitates you to use droppable() method to make any DOM element droppable at a specified target.

**Syntax:**

You can use the droppable () method in two forms:

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

2.  $(selector, context).droppable ("action", params) Method

## First Method

* The droppable (options) method specifies that you can use an HTML element as an element in which you can drop other elements. Here, the option parameter specifies the behavior of the elements involved.

**Syntax:**

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

You can use one or more options at a time using JavaScript object. If you are using more than one option then will have to separate them by using a comma. For example:

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

Following is a list of the different options that can be used with this method:

|  |  |  |
| --- | --- | --- |
| **Option** | | **Description** |
| accept | This option is used when you need to control which draggable elements are to be accepted for dropping. by default its value is \*. | |
| activeclass | This option is a string representing one or more css classes to be added to the droppable element when an accepted element (one of those indicated in options.accept) is being dragged. by default its value is false. | |
| addclasses | This option when set to false will prevent the ui-droppable class from being added to the droppable elements. by default its value is true. | |
| diasabled | This option when set to true disables the droppable. by default its value is false. | |
| greedy | This option is used when you need to control which draggable elements are to be accepted for dropping on nested droppables. by default its value is false. if this option is set to true, any parent droppables will not receive the element. | |
| hoverclass | This option is string representing one or more css classes to be added to the element of droppable when an accepted element (an element indicated in options.accept) moves into it. by default its value is false. | |
| scope | This option is used to restrict the droppable action of draggable elements only to items that have the same options.scope (defined in draggable (options)). by default its value is "default". | |
| tolerence | This option is a string that specifies which mode to use for testing whether a draggable is hovering over a droppable. By default its value is "intersect". | |

## jQuery UI Droppable() method example 1

Let's take an example to demonstrate the jQuery UI Droppable() method with no parameter.

<!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: 100px; padding: 0.5em; float: left;

margin: 22px 5px 10px 0;

}

#droppable-1 {

width: 150px; height: 120px;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>I am SSSIT.org<br/> Drag me to NeoRays </p>

</div>

<div id="droppable-1" class="ui-widget-header">

<p>Welcome to NeoRays</p>

</div>

</body>

</html>

## Second Method

The droppable ("action", params) method is used to perform an action on droppable elements, like preventing droppable functionality. You have to pass an action specified as a string in the first argument. i.e. "disable" to prevent the drop.

**Syntax:**

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

Following is the list of actions that can be used with this method:

|  |  |  |
| --- | --- | --- |
| **Action** | | **Description** |
| accept | This option is used when you need to control which draggable elements are to be accepted for dropping. by default its value is \*. | |
| activeclass | This option is a string representing one or more css classes to be added to the droppable element when an accepted element (one of those indicated in options.accept) is being dragged. by default its value is false. | |
| addclasses | This option will prevent the ui-droppable class from being added to the droppable elements,when set to false.by default its value is true. | |
| disabled | This option disables the droppable,when set to true. by default its value is false. | |
| greedy | This option is used when you need to control which draggable elements are to be accepted for dropping on nested droppables. by default its value is false. if this option is set to true, any parent droppables will not receive the element. | |
| hoverclass | This option is string representing one or more css classes to be added to the element of droppable when an accepted element (an element indicated in options.accept) moves into it. by default its value is false. | |
| scope | This option is used to restrict the droppable action of draggable elements only to items that have the same options.scope (defined in draggable (options)). by default its value is "default". | |
| tolerence | This option is a string that specifies which mode to use for testing whether a draggable is hovering over a droppable. by default its value is "intersect". | |

## jQuery UI Droppable() method example 5

Let's take an example to demonstrate the use of destroy () method:

<!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-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>

There are some event methods in jQueryUI which get triggered for a particular event. Following are some event methods.

|  |  |
| --- | --- |
| **Event Method** | **Description** |
| activate(event, ui) | This event is triggered when the accepted draggable element starts dragging. this can be useful if you want to make the droppable "light up" when it can be dropped on. |
| create(event,ui) | This event is triggered when a droppable element is created. ?where? event is of type event, and "ui" is of type object. |
| deactivate(event,ui) | This event is triggered when an accepted draggable stops dragging. "Where" event is of type event, and ui is of type object. |
| drop(event,ui) | This action is triggered when an element is dropped on the droppable. this is based on the tolerance option. "Where" event is of type event, and "ui" is of type object. |
| out(event,ui) | This event is triggered when an accepted draggable element is dragged out of the droppable. this is based on the tolerance option. ?Where? event is of type event, and "ui" is of type object. |
| over(event,ui) | This event is triggered when an accepted draggable element is dragged over the droppable. this is based on the tolerance option. "Where" event is of type event, and "ui" is of type object. |

## jQuery UI Droppable() method example 6

Let's take an example to demonstrate the "drop", "over" and "out" events during drop functionality.

<!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>I am the target. </p>

</div>

</body>

</html>