

JqueryUI Tutorial

- ▣ [JqueryUI - Home](#)
- ▣ [JqueryUI - Overview](#)
- ▣ [JqueryUI - Environment Setup](#)

JqueryUI Interactions

- ▣ [JqueryUI - Draggable](#)
- ▣ [JqueryUI - Droppable](#)
- ▣ [JqueryUI - Resizable](#)
- ▣ [JqueryUI - Selectable](#)
- ▣ [JqueryUI - Sortable](#)

JqueryUI Widgets

- ▣ [JqueryUI - Accordion](#)
- ▣ [JqueryUI - Autocomplete](#)
- ▣ [JqueryUI - Button](#)
- ▣ [JqueryUI - Datepicker](#)
- ▣ [JqueryUI - Dialog](#)
- ▣ [JqueryUI - Menu](#)
- ▣ [JqueryUI - Progressbar](#)
- ▣ [JqueryUI - Slider](#)
- ▣ [JqueryUI - Spinner](#)
- ▣ [JqueryUI - Tabs](#)
- ▣ [JqueryUI - Tooltip](#)

JqueryUI Effects

- ▣ [JqueryUI - Add Class](#)
- ▣ [JqueryUI - Color Animation](#)
- ▣ [JqueryUI - Effect](#)
- ▣ [JqueryUI - Hide](#)
- ▣ [JqueryUI - Remove Class](#)
- ▣ [JqueryUI - Show](#)
- ▣ [JqueryUI - Switch Class](#)
- ▣ [JqueryUI - Toggle](#)
- ▣ [JqueryUI - Toggle Class](#)

JqueryUI Utilities

- ▣ [JqueryUI - Position](#)
- ▣ [JqueryUI - Widget Factory](#)

JqueryUI Useful Resources

jQueryUI - Datepicker

[Previous Page](#)[Next Page](#)

Datepickers in jQueryUI allow users to enter dates easily and visually. You can customize the date format and language, restrict the selectable date ranges and add in buttons and other navigation options easily.

jQueryUI provides **datepicker()** method that creates a datepicker and changes the appearance of HTML elements on a page by adding new CSS classes. Transforms the `<input>`, `<div>`, and `` elements in the wrapped set into a datepicker control.

By default, for `<input>` elements, the datepicker calendar opens in a small overlay when the associated text field gains focus. For an inline calendar, simply attach the datepicker to a `<div>`, or `` element.

Syntax

The **datepicker()** method can be used in two forms:

- `$(selector, context).datepicker (options)` [Method](#)
- `$(selector, context).datepicker ("action", [params])` [Method](#)

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

The *datepicker (options)* method declares that an `<input>` element (or `<div>`, or ``, depending on how you choose to display the calendar) should be managed as a datepicker. The *options* parameter is an object that specifies the behavior and appearance of the datepicker elements.

Syntax
















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

You can provide one or more options at a time using Javascript object. If there are more than one options to be provided then you will separate them using a comma as follows:















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



Following table lists the different *options* that can be used with this method:

Option	Description
altField Method	This option specifies a jQuery selector for a field that is also updated with any date selections. The <i>altFormat</i> option can be used to set the format for this value. This is quite useful for setting date values into a hidden input element to be submitted to the server, while displaying a more user-friendly format to the user. By default its value is "".
altFormat Method	This option is used when an <i>altField</i> option is specified. It provides the format for the value to be written to the alternate element. By default its value is "".

appendText 	This option is a String value to be placed after the <input> element, intended to show instructions to the user. By default its value is "".
autoSize 	This option when set to <i>true</i> resizes the <input> element to accommodate the datepicker's date format as set with the <i>dateFormat</i> option. By default its value is false .
beforeShow 	This option is a callback function that's invoked just before a datepicker is displayed, with the <input> element and datepicker instance passed as parameters. This function can return an options hash used to modify the datepicker. By default its value is "".
beforeShowDay 	This option is a callback function which takes a date as parameter, that's invoked for each day in the datepicker just before it's displayed, with the date passed as the only parameter. This can be used to override some of the default behavior of the day elements. This function must return a three-element array. By default its value is null .
buttonImage 	This option specifies the path to an image to be displayed on the button enabled by setting the <i>showOn</i> option to one of <i>button</i> or <i>both</i> . If <i>buttonText</i> is also provided, the button text becomes the <i>alt</i> attribute of the button. By default its value is "".
buttonImageOnly 	This option if set to <i>true</i> , specifies that the image specified by <i>buttonImage</i> is to appear standalone (not on a button). The <i>showOn</i> option must still be set to one of <i>button</i> or <i>both</i> for the image to appear. By default its value is false .
buttonText 	This option specifies the caption for the button enabled by setting the <i>showOn</i> option to one of <i>button</i> or <i>both</i> . By default its value is "...".
calculateWeek 	This option is a custom function to calculate and return the week number for a date passed as the lone parameter. The default implementation is that provided by the <i>\$.datepicker.iso8601Week()</i> utility function.
changeMonth 	This option if set to <i>true</i> , a month dropdown is displayed, allowing the user to directly change the month without using the arrow buttons to step through them. By default its value is false .
changeYear 	This option if set to <i>true</i> , a year dropdown is displayed, allowing the user to directly change the year without using the arrow buttons to step through them. Option <i>yearRange</i> can be used to control which years are made available for selection. By default its value is false .
closeText 	This option specifies the text to replace the default caption of Done for the close button. It is used when the button panel is displayed via the <i>showButtonPanel</i> option. By default its value is "Done" .
constrainInput 	This option if set <i>true</i> (the default), text entry into the <input> element is constrained to characters allowed by the current <i>dateFormat</i> option. By default its value is true .
currentText 	This option specifies the text to replace the default caption of Today for the current button. This is used if the button panel is displayed via the <i>showButtonPanel</i> option. By default its value is Today .
dateFormat 	This option specifies the date format to be used. By default its value is mm/dd/yy .
dayNames 	This option is a 7-element array providing the full day names with the 0th element representing Sunday. Can be used to localize the control. By default its value is ["Sunday", "Monday", "Tuesday", "Wednesday", "Thursday", "Friday", "Saturday"].

dayNamesMin 	This option is a 7-element array providing the minimal day names with the 0th element representing Sunday, used as column headers. Can be used to localize the control. By default its value is ["Su", "Mo", "Tu", "We", "Th", "Fr", "Sa"].
dayNamesShort 	This option specifies a 7-element array providing the short day names with the 0th element representing Sunday. Can be used to localize the control. By default its value is ["Sun", "Mon", "Tue", "Wed", "Thu", "Fri", "Sat"].
defaultDate 	This option sets the initial date for the control, overriding the default value of today, if the <code><input></code> element has no value. This can be a <i>Date</i> instance, the <i>number</i> of days from today, or a <i>string</i> specifying an absolute or relative date. By default its value is null .
duration 	This option specifies the speed of the animation that makes the datepicker appear. Can be one of <i>slow</i> , <i>normal</i> , or <i>fast</i> , or the number of milliseconds for the animation to span. By default its value is normal .
firstDay 	This option specifies which day is considered the first day of the week, and will be displayed as the left-most column. By default its value is 0 .
gotoCurrent 	This option when set to <i>true</i> , the current day link is set to the selected date, overriding the default of today. By default its value is false .
hideIfNoPrevNext 	This option if set to <i>true</i> , hides the next and previous links (as opposed to merely disabling them) when they aren't applicable, as determined by the settings of the <i>minDate</i> and <i>maxDate</i> options. By default its value is false .
isRTL 	This option when set to <i>true</i> , specifies the localizations as a right-to-left language. By default its value is false .
maxDate 	This option sets the maximum selectable date for the control. This can be a <i>Date</i> instance, the number of days from today, or a <i>string</i> specifying an absolute or relative date. By default its value is null .
minDate 	This option sets the minimum selectable date for the control. This can be a <i>Date</i> instance, the <i>number</i> of days from today, or a <i>string</i> specifying an absolute or relative date. By default its value is null .
monthNames 	This option is a 12-element array providing the full month names with the 0th element representing January. Can be used to localize the control. By default its value is ["January", "February", "March", "April", "May", "June", "July", "August", "September", "October", "November", "December"].
monthNamesShort 	This option specifies a 12-element array providing the short month names with the 0th element representing January. Can be used to localize the control. By default its value is ["Jan", "Feb", "Mar", "Apr", "May", "Jun", "Jul", "Aug", "Sep", "Oct", "Nov", "Dec"].
navigationAsDateFormat 	This option if set to <i>true</i> , the navigation links for <i>nextText</i> , <i>prevText</i> , and <i>currentText</i> are passed through the <i>\$.datepicker.formatDate()</i> function prior to display. This allows date formats to be supplied for those options that get replaced with the relevant values. By default its value is false .
nextText 	This option specifies the text to replace the default caption of Next for the next month navigation link. ThemeRoller replaces this text with an icon. By default its value is Next .
numberOfMonths 	This option specifies the number of months to show in the datepicker. By default its value is 1 .
onChangeMonthYear 	This option is a callback that's invoked when the datepicker moves to a new month or year, with the selected year, month (1-based), and datepicker instance passed as parameters, and the function context is set to the input field element. By default its value is null .

onClose 	This option is a callback invoked whenever a datepicker is closed, passed the selected date as text (the empty string if there is no selection), and the datepicker instance, and the function context is set to the input field element. By default its value is null .
onSelect 	This option is a callback invoked whenever a date is selected, passed the selected date as text (the empty string if there is no selection), and the datepicker instance, and the function context is set to the input field element. By default its value is null .
prevText 	This option specifies the text to replace the default caption of <i>Prev</i> for the previous month navigation link. (Note that the ThemeRoller replaces this text with an icon). By default its value is PrevdefaultDatedayNamesMin .
selectOtherMonths 	This option if set to <i>true</i> , days shown before or after the displayed month(s) are selectable. Such days aren't displayed unless the <i>showOtherMonths</i> option is true. By default its value is false .
shortYearCutoff 	This option if its a number, specifies a value between 0 and 99 years before which any 2-digit year values will be considered to belong to the previous century. If this option is a string, the value undergoes a numeric conversion and is added to the current year. The default is +10 which represents 10 years from the current year.
showAnim 	This option specifies sets the name of the animation to be used to show and hide the datepicker. If specified, must be one of <i>show</i> (the default), <i>fadeIn</i> , <i>slideDown</i> , or any of the jQuery UI show/hide animations. By default its value is show .
showButtonPanel 	This option if set to <i>true</i> , a button panel at the bottom of the datepicker is displayed, containing current and close buttons. The caption of these buttons can be provided via the <i>currentText</i> and <i>closeText</i> options. By default its value is false .
showCurrentAtPos 	This option specifies the 0-based index, starting at the upper left, of where the month containing the current date should be placed within a multi-month display. By default its value is 0 .
showMonthAfterYear 	This option specifies if set to <i>true</i> , the positions of the month and year are reversed in the header of the datepicker. By default its value is false .
showOn 	This option specifies when the datepicker should appear. The possible values are <i>focus</i> , <i>button</i> or <i>both</i> . By default its value is focus .
showOptions 	This option provides a hash to be passed to the animation when a jQuery UI animation is specified for the <i>showAnim</i> option. By default its value is {} .
showOtherMonths 	This option if set to <i>true</i> , dates before or after the first and last days of the current month are displayed. These dates aren't selectable unless the <i>selectOtherMonths</i> option is also set to true. By default its value is false .
showWeek 	This option if set to <i>true</i> , the week number is displayed in a column to the left of the month display. The <i>calculateWeek</i> option can be used to alter the manner in which this value is determined. By default its value is false .
stepMonths 	This option specifies specifies how many months to move when one of the month navigation controls is clicked. By default its value is 1 .
weekHeader 	This option specifies the text to display for the week number column, overriding the default value of Wk , when <i>showWeek</i> is true. By default its value is Wk .

yearRange 	This option specifies limits on which years are displayed in the dropdown in the form <i>from:to</i> when <i>changeYear</i> is <i>true</i> . The values can be absolute or relative (for example: 2005:+2, for 2005 through 2 years from now). The prefix <i>c</i> can be used to make relative values offset from the selected year rather than the current year (example: c-2:c+3). By default its value is c-10:c+10 .
yearSuffix 	This option displays additional text after the year in the datepicker header. By default its value is "".

Following section will show you few working examples of datepicker functionality.

Default functionality

The following example demonstrates a simple example of datepicker functionality passing no parameters to the **datepicker()** method .



```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-1" ).datepicker();
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-1"></p>
  </body>
</html>
```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript,you must see the following output. Now you can play with the result:

Enter Date:

Inline Datepicker

The following example demonstrates a simple example of inline datepicker functionality.



```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-2" ).datepicker();
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    Enter Date: <div id="datepicker-2"></div>
  </body>
</html>
```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript,you must see the following output. Now you can play with the result:

Enter Date:

In the above example we use <div> element to get the inline date picker.

Use of appendText, dateFormat, altField and altFormat

The following example shows the usage of three important options **(a) appendText (b) dateFormat** (c) altField and **(d) altFormat** in the datepicker function of JQueryUI.



```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-3" ).datepicker({
          appendText: "(yy-mm-dd)",
          dateFormat: "yy-mm-dd",
          altField: "#datepicker-4",
          altFormat: "DD, d MM, yy"
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-3"></p>
    <p>Alternate Date: <input type="text" id="datepicker-4"></p>
  </body>
</html>
```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Date:

Alternate Date:

In the above example you can see that the date format for first input is set as *yy-mm-dd*. If you select some date from datepicker the same date is reflected in the second input field whose date format is set as "DD, d MM, yy".

Use of beforeShowDay

The following example shows the usage of option **beforeShowDay** in the datepicker function of JQueryUI.



```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-5" ).datepicker({
          beforeShowDay : function (date)
          {
            var dayOfWeek = date.getDay ();
            // 0 : Sunday, 1 : Monday, ...
            if (dayOfWeek == 0 || dayOfWeek == 6) return [false];
            else return [true];
          }
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-5"></p>
  </body>
</html>
```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Date:

In the above example sunday and saturday are disabled.

Use of showOn, buttonImage, and buttonImageOnly

The following example shows the usage of three important options (a) **showOn** (b) **buttonImage** and (c) **buttonImageOnly** in the datepicker

function of JQueryUI.

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-6" ).datepicker({
          showOn:"button",
          buttonImage: "/jqueryui/images/calendar-icon.png",
          buttonImageOnly: true
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-6"></p>
  </body>
</html>
```



Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Date:

In the above example an icon is displayed which needs to be clicked to open the datepicker.

Use of defaultDate, dayNamesMin, and duration

The following example shows the usage of three important options (a) **defaultDate** (b) **dayNamesMin** and (c) **duration** in the datepicker function of JQueryUI.

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-7" ).datepicker({
          defaultDate:+9,
          dayNamesMin: [ "So", "Mo", "Di", "Mi", "Do", "Fr", "Sa" ],
          duration: "slow"
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-7"></p>
  </body>
</html>
```



Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Date:

In the above example the names of the days are changed using *dayNamesMin*. You can also see a default date is set.

Use of prevText, nextText, showOtherMonths and selectOtherMonths

The following example shows the usage of three important options (a) **prevText** (b) **nextText** (c) **showOtherMonths** and (d) **selectOtherMonths** in the datepicker function of JQueryUI.



```

<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-8" ).datepicker({
          prevText:"click for previous months",
          nextText:"click for next months",
          showOtherMonths:true,
          selectOtherMonths: false
        });
        $( "#datepicker-9" ).datepicker({
          prevText:"click for previous months",
          nextText:"click for next months",
          showOtherMonths:true,
          selectOtherMonths: true
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Start Date: <input type="text" id="datepicker-8"></p>
    <p>Enter End Date: <input type="text" id="datepicker-9"></p>
  </body>
</html>

```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Start Date:

Enter End Date:

In the above example the prev and next links have captions. If you click on the element, the datepicker opens. Now in the first datepicker, the other months dates are disabled as selectOtherMonths is set *false*. In the second date picker for second input type, the selectOtherMonths is set to *true*.

Use of changeMonth, changeYear, and numberOfMonths

The following example shows the usage of three important options (a) **changeMonth** (b) **changeYear** and (c) **numberOfMonths** in the datepicker function of jQueryUI.

```

<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-10" ).datepicker({
          changeMonth:true,
          changeYear:true,
          numberOfMonths:[2,2]
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-10"></p>
  </body>
</html>

```



Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output. Now you can play with the result:

Enter Date:

In the above example you can see dropdown menus for Month and Year fields. And we are displaying 4 months in an array structure of [2,2].

Use of showWeek, yearSuffix, and showAnim

The following example shows the usage of three important options **(a) showWeek** **(b) yearSuffix** and **(c) showAnim** in the datepicker function of JQueryUI.

```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-11" ).datepicker({
          showWeek:true,
          yearSuffix:"-CE",
          showAnim: "slide"
        });
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-11"></p>
  </body>
</html>
```

Try it

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript,you must see the following output. Now you can play with the result:

Enter Date:

In the above example you can see that week numbers are displayed on the left side of datepicker as *showWeek* is set to *true*. The year is have a suffix of "-CE".

\$(selector, context).datepicker ("action", [params]) Method

The *datepicker (action, params)* method can perform an action on the calendar, such as such as selecting a new date. The **action** is specified as a string in the first argument and optionally, one or more **params** can be provided based on the given action.








Basically, here actions are nothing but they are jQuery methods which we can use in the form of string.

Syntax

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

The following table lists the actions for this method:

Action	Description
destroy()	This action removes the datepicker functionality completely. This will return the element back to its pre-init state. This method does not accept any arguments.
dialog(date [, onSelect] [, settings] [, pos])	This action displays datepicker in a jQuery UI dialog box .
getDate()	This action returns the Date corresponding to the selected date. This method does not accept any arguments.
hide()	This action closes the previously opened date picker. This method does not accept any arguments.
isDisabled()	This action checks if the date picker funcitonality is disabled. This method does not accept any arguments.
option(optionName)	This action retrieves the value currently associated with the specified <i>optionName</i> .

option() 	This action gets an object containing key/value pairs representing the current datepicker options hash. This method does not accept any arguments.
option(optionName, value) 	This action sets the value of the datepicker option associated with the specified optionName.
option(options) 	This action sets one or more options for the datepicker.
refresh() 	This action redraws the date picker, after having made some external modifications. This method does not accept any arguments.
setDate(date) 	This action sets the specified date as the current date of the datepicker.
show() 	This action opens the date picker. If the datepicker is attached to an input, the input must be visible for the datepicker to be shown. This method does not accept any arguments.
widget() 	This action returns a jQuery object containing the datepicker.

Following examples show the use of some of the actions listed in the above table.

Use of setDate() action

Now let us see an example using the actions from the above table. The following example demonstrates the use of actions *setDate*.



```
<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-12" ).datepicker();
        $( "#datepicker-12" ).datepicker("setDate", "10w+1");
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-12"></p>
  </body>
</html>
```

Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output:

Enter Date:

Use of show() action

The following example demonstrates the use of action *show*.

```

<!doctype html>
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>jQuery UI Datepicker 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() {
        $( "#datepicker-13" ).datepicker();
        $( "#datepicker-13" ).datepicker("show");
      });
    </script>
  </head>
  <body>
    <!-- HTML -->
    <p>Enter Date: <input type="text" id="datepicker-13"></p>
  </body>
</html>

```



Let's save above code in an HTML file **datepickerexample.htm** and open it in a standard browser which supports javascript, you must see the following output:

Enter Date:

Management on datepicker elements

There are no datepicker event methods as of now!

[< Previous Page](#)

[Next Page >](#)

Advertisements

Scalp MED®

"I'm feeling more myself and confident now that my hair is back."
- Eric*

Before 3 MONTHS 6 MONTHS ERIC

Try for ONLY \$29⁹⁵*
*Eric used Scalp Med® 2 times Daily for 6 months.
**Intro offer for 2-mo supply + Autoship Enroll + S&H

ORDER NOW



© Copyright 2015. All Rights Reserved. [About us](#) | [Write for us](#) | [Contact](#)