

PHP - Calendar. DatePicker Calendar

Demo:

Fixed Display Style

Date 1 :

February

2019

Mo	Tu	We	Th	Fr	Sa	Su
05	28	29	30	31	1	2 3
06	4	5	6	7	8	9 10
07	11	12	13	14	15	16 17
08	18	19	20	21	22	23 24
09	25	26	27	28	1	2 3
10	4	5	6	7	8	9 10

- Set default date to server date (Auto unset if date is disabled)
- Set year navigate from 1890 to 2080
- Allow date selectable from 1st Jan 1890 to 01 May 2045
- Not allow to navigate other dates from above
- Disable date of 10th, 13th, and 23th on each month
- Start week on Monday
- Show calendar weeks
- Set/show tooltips
- Set timezone to Australia/Melbourne

Check the value

Set calendar value to today

Code:

<?php

```
$myCalendar = new tc_calendar("date2");
$myCalendar->setIcon("calendar/images/iconCalendar.gif");
$myCalendar->setDate(date('d'), date('m'), date('Y'));
$myCalendar->setPath("calendar/");
$myCalendar->setYearInterval(1890, 2080);

$myCalendar->dateAllow('1890-01-01', '2045-05-01', false);

$myCalendar->setSpecificDate(array("2039-01-10", "2039-01-13", "2039-01-23"), 0, 'month');

$myCalendar->startMonday(true);
$myCalendar->showWeeks(true);

//Tooltips
$myCalendar->setTooltips(array("2013-07-02", "2013-07-15", "2013-07-25"), 'ŞTÁÍÁ ştáíá şşş ş test!', '');
$myCalendar->setTooltips(array("2013-06-06", "2013-06-01", "2013-06-05"), 'א' אשש לבחור הארץ זה', 'month');
$myCalendar->setTooltips(array("1969-07-06", "2040-07-01", "2013-06-05"), 'Δεν επιτρέπετε η επιλογή αυτής της ημέρας', 'month');
$myCalendar->setTooltips(array("1969-07-06", "2040-07-01", "2013-06-05"), 'بعد 2038 العام للدعم الخاص بي ليس لديها الإصدار', 'month');
$myCalendar->setTooltips(array("1969-07-06", "2040-07-01", "2013-06-05"), 'の間の日付を選択してください', 'month');
$myCalendar->setTooltips(array("1969-07-06", "2040-07-01", "2013-06-05"), '올바르지 않은 값입니다', 'month');
$myCalendar->setTooltips(array("2013-06-06", "2013-06-11", "2013-06-15"), 'और बाद के वर्ष का समर्थन नहीं है!', 'month');
$myCalendar->setTooltips(array("2013-07-06", "2013-01-01", "2013-12-25"), 'ჩვენს თვეში არ არის დღეები', 'year');
$myCalendar->setTooltips(array("2013-07-06", "2013-07-15", "2013-07-25"), '请选择日期%之前一个', '');

$myCalendar->setTimezone("Australia/Melbourne");
$myCalendar->writeScript();
?>
```


DatePicker Style

Date 2 :

11

February

2019



Check the value

- Default date to current server date (Auto unset if date is disabled)
- Set year navigate from 1960 to 2025
- Allow date selectable from 01 January 2010 to 01 March 2025
- Allow to navigate other dates from above
- Disable specific date 1, 13, and 25 of every month
- Custom alert when calendar's value has been changed

Code:

```
<?php

$myCalendar = new tc_calendar("date1", true);
$myCalendar->setIcon("calendar/images/iconCalendar.gif");
$myCalendar->setDate(01, 03, 1960);
$myCalendar->setPath("calendar/");
$myCalendar->setYearInterval(1960, 2017);
$myCalendar->dateAllow('2010-01-01', '2025-03-01');
$myCalendar->setSpecifcDate(array("2011-04-01", "2011-04-01", "2011-04-13", "2011-04-25"), 0, 'month');
$myCalendar->setOnChange("myChanged('test')");
$myCalendar->writeScript();
?>
```

```
<script language="javascript">
<!--
function myChanged(v){
    alert("Hello, value has been changed : "+document.getElementById("date1").value+"["+v+"]");
}
//-->
</script>
```

DatePicker with no input box

Date 3 :

Select Date



Check the value

- No default date
- Set year navigate from 2000 to 2017
- Allow date selectable from 13 May 2008 to 01 March 2025
- Allow to navigate other dates from above
- Date input box set to false
- Set alignment left and bottom
- Disable specific date 1, 4 April, and 25 December of every years
- Disable specific date 10, 14 of every months
- Disable 1 June 2011

Code:

```
<?php

$myCalendar = new tc_calendar("dates", true, false);
$myCalendar->setIcon("calendar/images/iconCalendar.gif");
```

```
$myCalendar->setDate(date('d'), date('m'), date('Y'));
$myCalendar->setPath("calendar/");
$myCalendar->setYearInterval(2000, 2017);
$myCalendar->dateAllow('2008-05-13', '2025-03-01');
$myCalendar->setDateFormat('j F Y');
$myCalendar->setDateFormatt('left', 'bottom');
$myCalendar->setSpecificDate(array("2011-04-01", "2011-04-04", "2011-12-25"), 0, 'year');
$myCalendar->setSpecificDate(array("2011-04-10", "2011-04-14"), 0, 'month');
$myCalendar->setSpecificDate(array("2011-06-01"), 0, '');
$myCalendar->writeScript();
?>
```

Date Pair Example

from:

11-Feb-2019

 to

17-Feb-2019

Check the value

Code:

```
<?php
$date3_default = "2019-02-11";
$date4_default = "2019-02-17";

$myCalendar = new tc_calendar("date3", true, false);
$myCalendar->setIcon("calendar/images/iconCalendar.gif");
$myCalendar->setDate(date('d', strtotime($date3_default))
    , date('m', strtotime($date3_default))
    , date('Y', strtotime($date3_default)));
$myCalendar->setPath("calendar/");
$myCalendar->setYearInterval(1970, 2020);
$myCalendar->setAlignment('left', 'bottom');
$myCalendar->setDatePair('date3', 'date4', $date4_default);
$myCalendar->writeScript();

$myCalendar = new tc_calendar("date4", true, false);
$myCalendar->setIcon("calendar/images/iconCalendar.gif");
$myCalendar->setDate(date('d', strtotime($date4_default))
    , date('m', strtotime($date4_default))
    , date('Y', strtotime($date4_default)));
$myCalendar->setPath("calendar/");
$myCalendar->setYearInterval(1970, 2020);
$myCalendar->setAlignment('left', 'bottom');
$myCalendar->setDatePair('date3', 'date4', $date3_default);
$myCalendar->writeScript();
?>
```

Color Themes (v3.75)

- theme1

26 November 2015

Copyright

This calendar datepicker is totally free to use. You can use it in all of your projects without any costs.

How to setup:

Only 2 steps requires for setup and use this calendar component.

Put the javascript file(.js) in the head section or somewhere else but it **should be include once in a page**.

```
<head>
<script language="javascript" src="calendar.js"></script>
</head>
```

Create form element in the html and put the following code

```
<form action="somewhere.php" method="post">
<?php
//get class into the page
require_once('classes/tc_calendar.php');

//instantiate class and set properties
$myCalendar = new tc_calendar("date1", true);
$myCalendar->setIcon("images/iconCalendar.gif");
$myCalendar->setDate(1, 1, 2000);

//output the calendar
$myCalendar->writeScript();
?>
</form>
```

How to get the value?

To get the date selected in calendar **by php after submit the form**, simple write script as the following:

```
<?php
$date = isset($_REQUEST["date1"]) ? $_REQUEST["date1"] : "";

?>
```

The above script should be on another file that html form point to. The parameter 'date1' is the object name that you set in the code at the time calendar construct. See further in Functions and Constructor below.

To get the date selected **by javascript on the current page**, write script as the following:

```
<form action="somewhere.php" method="post" name="form1">
<?php
//get class into the page
require_once('classes/tc_calendar.php');

//instantiate class and set properties
$myCalendar = new tc_calendar("date1", true);
$myCalendar->setIcon("images/iconCalendar.gif");
$myCalendar->setDate(1, 1, 2000);

//output the calendar
```

```
$myCalendar->writeScript();
?>
</form>

//use javascript to get the value
<script language="javascript">
<!--
function showDateSelected(){
    alert("Date selected is "+document.form1.date1.value);
}
//-->
</script>
//create link to click and check calendar value
<a href="javascript:showDateSelected();">Check calendar value</a>
```

Calendar Properties

rtl - specify the right-to-left direction of calendar

Ex. `$myCalendar->rtl = true;`

zindex - specify the zindex value to calendar

Ex. `$myCalendar->zindex = 99;`

Calendar Functions

Constructor

tc_calendar (string bindObjectName)

tc_calendar (string bindObjectName, boolean date_picker)

tc_calendar (string bindObjectName, boolean date_picker, bool show_input)

tc_calendar (string bindObjectName, boolean date_picker, bool show_input, string timezone)

'date_picker' default value is false.

'show_input' default value is true

'timezone' default value is system timezone (as php settings). You can see the valid timezone on <http://php.net/manual/en/timezones.php>

Methods

autoSubmit (bool flag, string form_name, string target_url)

Specify the calendar to auto-submit the value. Default value of autosubmit is **false**

To set calendar auto submit, specify flag to true and you need to specify either *form_name* or *target_url* to make the calendar to perform autosubmit correctly

Ex 1. `$myCalendar->autoSubmit(true, "myForm");`

//assume that the calendar is in the form named 'myForm', then tell the calendar to auto submit the value (other values in myForm will be submitted together by html post method)

Ex 2. `$myCalendar->autoSubmit(true, "", "anywhere.php");`

//tell the calendar to submit the value to 'anywhere.php'. This method will submit only calendar value via html get method

dateAllow (date from, date to, bool navigate_not_allowed_date)

Specify date range allow to select. Other dates from input will be disabled. The parameter *navigate_not_allowed_date* will handle the user to navigate over the disable date, default is true (means allow)

Specify both date *from* and *to* will set range of date user can select.

Specify only date *from* or *to* will set range from/upto year set by *setYearInterval* method.

Ex 1. `$myCalendar->dateAllow('2008-05-13', '2010-03-01', false);` //allow user select date from 13 May 2008 to 1 Mar 2010

Ex 2. `$myCalendar->dateAllow('2008-05-13', '', false);` //allow user select date from 13 May 2008 upto whatever can navigate

Ex 3. `$myCalendar->dateAllow('', '2010-03-01', false);` //allow user select date from whatever can navigate upto 1 Mar 2010

disabledDay (string day)

Specify day column to be disabled. Possible day value: Mon, Tue, Wed, Thu, Fri, Sat, Sun

Ex. `$myCalendar->disabledDay('Sun');` //disable Sunday column

getDate ()

Get the calendar value in date format YYYY-MM-DD

Ex. `$myCalendar->getDate();` //return 2009-06-19

getScript () - v3.75

Get the calendar output as script (for print/echo later). The method has the same purpose as **writeScript()** but return script as string instead of echo by itself.

Ex. `$myCalendar->getScript();`

setAlignment (string horizontal_align, string vertical_align)

Optional: Set the alignment of calendar datepicker with the calendar container. The value of horizontal alignment can be 'left' or 'right' (default) and the value of vertical alignment can be 'top' or 'bottom' (default).

Ex. `$myCalendar->setAlignment('left', 'top');`

setAutoHide (bool auto, int delay_time)

Optional: Set an auto hide of calendar when not active. The calendar has default value of auto hidden since version 3.65 with default delay time of 1000 (1 second)

Ex 1. `$myCalendar->setAutoHide(true, 3000);` //auto hide in 3 seconds

Ex 2. `$myCalendar->setAutoHide(false);`

setDate (int day, int month, int year)

Optional: Set default selected date to the value input. For month parameter: January=1 and December=12

Ex. `$myCalendar->setDate(15, 6, 2005);` //Set the date to 15 June 2005

setDateFormat (string format)

Optional: Set the format of date display when no input box. Apply with 'showInput' function

Ex. `$myCalendar->setDateFormat('j F Y');` //date will display like '9 September 2009'

setDatePair (string calendar_name1, string calendar_name2, string default_pair_value)

Optional: Set a date pair link to calendar object. This function will work when 2 calendars are setup together.

Ex 1. `$myCalendar->setDatePair('date1', 'date2');` //This means when the value of calendar 'date1' is set, the calendar 'date2' will not allow to select the date prior to 'date1'

Ex 2. `$myCalendar->setDatePair('date1', 'date2', "2011-03-26");` //suppose you are setting this value on date1 object, '2011-03-26' will be assigned as default date of date2 and calculated

setIcon (string url)

Optional: Set icon in date picker mode. If the icon is not set the date picker will show text as link.

Ex. `$myCalendar->setIcon("images/iconCalendar.gif");`

setHeight (int height) - deprecated since v2.9 - auto sizing applied

Optional: Set height of calendar. Default value is 205 pixels

Ex. `$myCalendar->setHeight(205);`

setOnChange (string value)

Optional: Set the onchange javascript function to calendar

Ex. `$myCalendar->setOnChange("javascriptFunctionName()");`

setPath (string path)

Optional: Set the path to the 'calendar_form.php' if it is not in the same directory as your script. The path string is a relative path to the script file.

Ex. `$myCalendar->setPath("folder1/");`

setSpecificDate (array dates, int type, string recursive)

Optional: Set the specific to enable or disable.

dates: the array of date, and date should be in format YYYY-MM-DD

type: enable or disable the date specified. 0=disable, 1=enable

recursive: " (blank)=no recursive, calendar will process only on exact date, 'month'=monthly recursive, or 'year'=yearly recursive

Ex 1. `$myCalendar->setSpecificDate(array("2011-04-22"));`

Ex 2. `$myCalendar->setSpecificDate(array("2011-04-22"), 0, 'month');` //disable date 22 every month

Ex 3. `$myCalendar->setSpecificDate(array("2011-04-22"), 0, 'year');` //disable 22 Apr on every year

setText (string text)

Optional: Set text to display. The text will show in date picker mode when the icon is not set.

Ex. `$myCalendar->setText("Click Me");`

setTheme (string theme_name) - v3.75

Optional: Set calendar color theme. Theme name is represented for css folder name such as 'theme1' and theme folder must exist as {calendar_folder}/css/(theme_name)/calendar.css

Ex. `$myCalendar->setTheme("theme1");`

setTimezone (string timezone)

Optional: Set timezone to calendar. The valid timezone string is as specified on <http://php.net/manual/en/timezones.php>

Ex `$myCalendar->setTimezone("Europe/London");`

setToolTips (array dates, string tooltip, string recursive)

Optional: Set the tooltip info to be displayed on certain dates, recursively or not.

dates: the array of dates, and date should be in format YYYY-MM-DD

tooltip: string/text to be displayed when mouse hover the info.gif icon in the top-right corner of each day

recursive: "" (blank)=no recursive, calendar will process only on exact date, "month"=monthly recursive, or "year"=yearly recursive

hint: the function can be called any number of times, like all 5 examples below, together; multi-tooltips will be displayed on multi-lines, in this main order: non-recurrent, monthly then yearly, and then in the order of calling the function.

Ex 1. `$myCalendar->setToolTips(array("2013-07-06", "2013-07-15", "2013-07-25"), "Tooltip Test");`

Ex 2. `$myCalendar->setToolTips(array("2013-06-06", "2013-06-01", "2013-06-05"), "Monthly Tooltip", "month");`

Ex 3. `$myCalendar->setToolTips(array("2013-06-06", "2013-07-11", "2013-09-15"), "Second Monthly Event", "month");`

Ex 4. `$myCalendar->setToolTips(array("2013-06-06", "2013-01-01", "2013-12-25"), "Yearly Event", "year");`

Ex 5. `$myCalendar->setToolTips(array("2013-07-06", "2013-07-15", "2013-07-25"), "Second Tooltip test");`

setWidth (int width) - deprecated since v2.9 - auto sizing applied

Optional: Set width of calendar. Default value is 150 pixels

Ex. `$myCalendar->setWidth(150);`

setYearInterval (int year_start, int year_end)

Optional: Set the year start and year end display on calendar combo box. Default value is +15 and -15 from current year (30 years)

Ex. `$myCalendar->setYearInterval(1970, 2020);`

showInput (bool flag)

Optional: Set the input box display on/off. If showInput set to false, the date will display in panel as example above '**DatePicker with no input box**'. Default value is true

Ex. `$myCalendar->showInput(false);`

showWeeks (bool flag)

Optional: Display weeks on calendar upon setting. Default value is false.

Ex. `$myCalendar->showWeeks(true);`

startDate (bool date_number)

Optional: You can choose any day as a start day of week. The possible values are 0 (for Sunday) through 6 (for Saturday). Default value is 0 (Sunday)

Ex. `$myCalendar->startDate(6);` //start day of week is Sat.

startMonday (bool flag) - deprecated since v3.61 - use startDate instead

Optional: Set whether the calendar will be start on Sunday or Monday. Set flag to **true** means the calendar will display first date as Monday, otherwise **false** will display first date as Sunday. Default value is false.

Ex. `$myCalendar->startMonday(true);`

writeScript()

Write the output calendar to the screen. For buffer the output and return as string, use **getScript()**

© Triconsole (triconsole.com)