

iranphp articles

عنوان مقاله :
نگارنده :
آدرس پست الکترونیک :
تاریخ نگارش :
تقویم جلالی
حق پرست
haghparast@gmail.com
.....

تقویم جلالی

مدتها بود که در فکر نوشتن یک تقویم فارسی بودم. تا اینکه بالاخره دست به کار شدم و نوشتم. برای این کار از برنامه نوشته شده توسط آقایان پورنادر و طوسی استفاده کردم که جا دارد همینجا از این بزرگواران تشکر نمایم. این تقویم در واقع یک کلاس PHP هست که دو تا ورودی دارد. اولین ورودی مربوط به سال می باشد و دومین ورودی مربوط به ماه سال. طرز استفاده از آن هم بسیار ساده است. کافی است که فایل کلاس را include کنید و سپس یک instance از کلاس بسازید. کد نمونه به صورت زیر است :

```
<?php
include "Calendar.Class.php";
$c = new Calendar;
$c->SetStyle();
$c->ShowJalaliMonth(1384,1);
?>
```

مشاهده تقویم:

<http://www.namanegar.com/MyCodes/Calendar>

دریافت برنامه :

<http://www.namanegar.com/MyCodes/Calendar/Calendar.zip>

```
<?php
//-----
//Class Name : Persian Calendar Class
//Published Date: 20 Apr 2005
//Author Email : haghparast@gmail.com
//License : Freeware
//-----
class Calendar
{
    var $OutputText = "";
    function SetStyle()
    {
        $Style =
            "<style>
            table {
            background-color: #EEEEDD;
            border-width: 0;
            font-size: 80%
            }

            td.day {
            border:1px solid #c6a646; background-color: #EEEEDD;
            text-align: center;
            direction: rtl
            }

            td.month {
            border:1px solid #c6a646; background-color: #FFCCBB;
            font-size : 9px;
            text-align: center;
            direction: rtl
            }

            td.today {
            border:1px solid #c6a646; background-color: #FFCCFF;
            text-align: center;
            direction: rtl
            }
            </style>
```

```
        ";
    print $Style;
}

function ReturnMonthName ($monname)
{
    switch ($monname)
    {
case 1:
        return "فروردین";
        break;
case 2:
        return "اردیبهشت";
        break;
case 3:
        return "خرداد";
        break;
case 4:
        return "تیر";
        break;
case 5:
        return "مرداد";
        break;
case 6:
        return "شهریور";
        break;
case 7:
        return "مهر";
        break;
case 8:
        return "آبان";
        break;
case 9:
        return "آذر";
        break;
case 10:
        return "دی";
        break;
case 11:
        return "بهمن";
        break;
case 12:
        return "اسفند";
        break;
    }
}

function div($a,$b)
{
    return (int) ($a / $b);
}

// Thanks to Roozbeh Pournader and Mohammad Toosi for their Date Conversion program
function gregorian_to_jalali ($g_y, $g_m, $g_d)
{
    $g_days_in_month = array(31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31);
    $j_days_in_month = array(31, 31, 31, 31, 31, 31, 30, 30, 30, 30, 30, 29);

    $gy = $g_y-1600;
    $gm = $g_m-1;
    $gd = $g_d-1;
```

```
$g_day_no = 365*$gy+$this->div($gy+3,4)-$this->div($gy+99,100)+$this->div($gy+399,400);

for ($i=0; $i < $gm; ++$i)
    $g_day_no += $g_days_in_month[$i];
if ($gm>1 && (($gy%4==0 && $gy%100!=0) || ($gy%400==0)))
    /* leap and after Feb */
    $g_day_no++;
$g_day_no += $gd;

$j_day_no = $g_day_no-79;

$j_np = $this->div($j_day_no, 12053); /* 12053 = 365*33 + 32/4 */
$j_day_no = $j_day_no % 12053;

$j_y = 979+33*$j_np+4*$this->div($j_day_no,1461); /* 1461 = 365*4 + 4/4 */

$j_day_no %= 1461;

if ($j_day_no >= 366) {
    $j_y += $this->div($j_day_no-1, 365);
    $j_day_no = ($j_day_no-1)%365;
}

for ($i = 0; $i < 11 && $j_day_no >= $j_days_in_month[$i]; ++$i)
    $j_day_no -= $j_days_in_month[$i];
$j_m = $i+1;
$j_d = $j_day_no+1;

return array($j_y, $j_m, $j_d);
}

function jalali_to_gregorian($j_y, $j_m, $j_d)
{
    $g_days_in_month = array(31, 28, 31, 30, 31, 30, 31, 31, 30, 31, 30, 31);
    $j_days_in_month = array(31, 31, 31, 31, 31, 31, 30, 30, 30, 30, 30, 29);

    $j_y = $j_y-979;
    $j_m = $j_m-1;
    $j_d = $j_d-1;

    $j_day_no = 365*$j_y + $this->div($j_y, 33)*8 + $this->div($j_y%33+3, 4);
    for ($i=0; $i < $j_m; ++$i)
        $j_day_no += $j_days_in_month[$i];

    $j_day_no += $j_d;

    $g_day_no = $j_day_no+79;

    $gy = 1600 + 400*$this->div($g_day_no, 146097); /* 146097 = 365*400 + 400/4 - 400/100 +
400/400 */
    $g_day_no = $g_day_no % 146097;

    $leap = true;
    if ($g_day_no >= 36525) /* 36525 = 365*100 + 100/4 */
    {
        $g_day_no--;
        $gy += 100*$this->div($g_day_no, 36524); /* 36524 = 365*100 + 100/4 - 100/100 */
        $g_day_no = $g_day_no % 36524;

        if ($g_day_no >= 365)
```

```
        $g_day_no++;
    else
        $leap = false;
}

$gy += 4*$this->div($g_day_no, 1461); /* 1461 = 365*4 + 4/4 */
$g_day_no %= 1461;

if ($g_day_no >= 366) {
    $leap = false;

    $g_day_no--;
    $gy += $this->div($g_day_no, 365);
    $g_day_no = $g_day_no % 365;
}

for ($i = 0; $g_day_no >= $g_days_in_month[$i] + ($i == 1 && $leap); $i++)
    $g_day_no -= $g_days_in_month[$i] + ($i == 1 && $leap);
$gm = $i+1;
$gd = $g_day_no+1;

return array($gy, $gm, $gd);
}

function CalculateTotalDays($jmonth)
{
    switch ($jmonth)
    {
        case 1:
        case 2:
        case 3:
        case 4:
        case 5:
        case 6:
            $TotalDays = 31;
            break;
        case 7:
        case 8:
        case 9:
        case 10:
        case 11:
            $TotalDays = 30;
            break;
        case 12:
            $TotalDays = 29;
            break;
    }
    return $TotalDays;
}

function ShowJalaliMonth($jyear, $jmonth)
{
    $todaygyyear = date("Y");
    $todaygmonth = date("m");
    $todaygday = date("d");
    list( $todayjyear, $todayjmonth, $todayjday ) = $this->gregorian_to_jalali($todaygyyear,
    $todaygmonth, $todaygday);
    list( $gyyear, $gmonth, $gday ) = $this->jalali_to_gregorian($jyear, $jmonth, 1);
    $FirstDay = mktime(0,0,0,$gmonth,$gday,$gyyear);
    $FirstDayArray = getdate($FirstDay);
    $DayOfWeek = $FirstDayArray['wday'];
}
```

```
switch ($DayOfWeek)
{
    case 0:
        $Difference = -1;
        break;
    case 1:
        $Difference = -2;
        break;
    case 2:
        $Difference = -3;
        break;
    case 3:
        $Difference = -4;
        break;
    case 4:
        $Difference = -5;
        break;
    case 5:
        $Difference = -6;
        break;
    case 6:
        $Difference = 0;
        break;
}
$this->OutputText = '<table width="252">' . "\n";
$this->OutputText .= ' <tr>' . "\n";
$this->OutputText .= ' <td class="month" colspan="7" width="70%">' . $this-
ReturnMonthName($jmonth) . '&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;' . $jyear . '</td>' . "\n";
$this->OutputText .= ' </tr>' . "\n";
$this->OutputText .= ' <tr>' . "\n";
$this->OutputText .= ' <td class="month" width="13%">جمعه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="13%">پنجشنبه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="15%">چهارشنبه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="20%">سه شنبه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="13%">دوشنبه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="13%">یکشنبه</td>' . "\n";
$this->OutputText .= ' <td class="month" width="13%">شنبه</td>' . "\n";
$this->OutputText .= ' </tr>' . "\n";
for ($i=0;$i<6;$i++)
{
    $Const = 7*$i+$Difference ;
    $this->OutputText .= ' <tr>' . "\n";
    $this->OutputText .= ' <td ';
    if (($Const + 7 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

    $this->OutputText .= "class='day'";
    $this->OutputText .= '>' . "\n";
    $Output = $Const + 7;
    if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;

    $this->OutputText .= ' </td>' . "\n";
    $this->OutputText .= ' <td ';
    if (($Const + 6 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

    $this->OutputText .= "class='day'";
    $this->OutputText .= '>' . "\n";
    $Output = $Const + 6;
```

```
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    <td ' ;
if (($Const + 5 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

$this->OutputText .= "class='day'";
$this->OutputText .= '>'."\n";
$Output = $Const + 5;
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    <td ' ;
if (($Const + 4 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

$this->OutputText .= "class='day'";
$this->OutputText .= '>'."\n";
$Output = $Const + 4;
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    <td ' ;
if (($Const + 3 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

$this->OutputText .= "class='day'";
$this->OutputText .= '>'."\n";
$Output = $Const + 3;
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    <td ' ;
if (($Const + 2 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

$this->OutputText .= "class='day'";
$this->OutputText .= '>'."\n";
$Output = $Const + 2;
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    <td ' ;
if (($Const + 1 == $todayjday) && ($jmonth == $todayjmonth) && ($jyear ==
$todayjyear)) $this->OutputText .= "class='today'"; else

$this->OutputText .= "class='day'";
$this->OutputText .= '>'."\n";
$Output = $Const + 1;
if (($Output>0) && ($Output<=$this->CalculateTotalDays($jmonth))) $this->OutputText .=
$Output;
$this->OutputText .= '    </td>'."\n";
$this->OutputText .= '    </tr>'."\n";
}
$this->OutputText .= '</table>'."\n";
print $this->OutputText;
}
}
```