

**Source File:** ~/public.html/lab04.php  
**Input:** URL  
**Output:** Standard Output  
**Value:** 3

A procedure for determining the date of Easter is as follows:

Divide	by	Quotient	Remainder
the year $x$	19		$a$
the year $x$	100	$b$	$c$
$b$	4	$d$	$e$
$b + 8$	25	$f$	
$b - f + 1$	3	$g$	
$19a + b - d - g + 15$	30		$h$
$c$	4	$i$	$k$
$32 + 2e + 2i - h - k$	7		$l$
$a + 11h + 22l$	451	$m$	
$h + l - 7m + 114$	31	$n$	$p$

Then

$n$  = number of the month (3 = March, 4 = April),

$p + 1$  = day of that month upon which Easter Sunday falls.

This method is valid for all years in the Gregorian calendar; that is, any year later than 1582. Write a program that implements the procedure described above for determining the date of Easter Sunday for a given year.

Your program will accept an unsigned decimal integer string via a URL. Let this value represent the year. You may assume that the year is valid; that is, it will be later than 1582.

Some additional notes for this assignment:

- Insert an HTML comment at the top of the document identifying you as the author, the class, and the assignment number. See the sample code from Lab 01.
- Add an echo statement to the beginning of the script section that will display your name, the course number, and the assignment number.
- When working on this assignment, you will need to do some type casting. The above procedure says to let  $b$  be the quotient from dividing the year  $x$  by 100. If you write a statement like

```
$b = $x/100;
```

you will not get the integer quotient. PHP will treat the quotient as a floating-point number. You will need to use a type cast as in:

```
$b = (int)($x/100);
```

Type casting is discussed on pp. 58-61. A reference to type casting in the PHP manual is

<https://www.php.net/manual/en/language.types.type-juggling.php>

- Since this assignment uses several PHP code blocks, it's always a good idea to check for syntax errors. You can do this by using the `-l` option to the `php` command at the command line as in

```
1 newuser@csunix ~/public_html> php -l lab04.php
2 No syntax errors detected in lab04.php
```

- You should always validate the rendered HTML code. The validator is discussed near the top of p. 6 and in Appendix A on pp. 629–631. By including the following link and image, a user will be able to click the image and receive a report from the validator.

```
1 <?php
2   $location = 'https://' . $_SERVER['HTTP_HOST'] . $_SERVER['REQUEST_URI'];
3   $location = urlencode($location);
4   echo '<a href="https://validator.w3.org/nu/?doc=' . $location . '">';
5   ?>
6   
8   </a>
```

After the document is valid, open it in your Web browser to see how it renders.

Upon completion of this assignment, submit your source file via Blackboard.