Source File:　～/public_html/lab16.php
Input:　　　　HTML/PHP Form or URL
Output:　　　Standard output (HTML Code)
Value:　　　　4

For this assignment write a PHP application that will compute the day number of a given date. The date (month, day, and year) will be provided via input from a form or a URL.

The number N of a day in the year can be computed by means of the following formula:

$$N = \mathrm{int}\left(\frac{275M}{9}\right) - K \times \mathrm{int}\left(\frac{M+9}{12}\right) + D - 30$$

where M is the month number, D the day of the month, and

$$K = \begin{cases} 1 & \text{for a leap year} \\ 2 & \text{for a common year} \end{cases}$$

N takes integer values, from 1 on January 1, to 365 (or 366 in leap years) on December 31.

The year 1752 requires special processing. In this year, the day following Wednesday, September 2, 1752, was Thursday, September 14, 1752. The calendar for the month of September looked like:

```
          Sep 1752
  S   M Tu  W Th  F  S
             1  2 14 15 16
 17 18 19 20 21 22 23
 24 25 26 27 28 29 30
```

A leap year is defined as a year in the Gregorian calendar containing 366 days with February 29th as the extra day. Leap years are those years divisible by 4, except those years that complete centuries (that is, divisible by 100) which must be divisible by 400. Thus, 1998 was not a leap year, 1996 was, 2000 was, 1900 was not, 1600 was, 2100 will not be, 2028 will be, etc.

The program needs to be able to handle the following error cases:

- variable `month` missing
- variable `month` empty
- variable `month` out of range
- variable `month` not numeric
- variable `day` missing
- variable `day` empty
- variable `day` out of range
- variable `day` not numeric
- variable `year` missing
- variable `year` empty
- variable `year` out of range (allow a maximum of four digits)
- variable `year` not in the Gregorian calendar (that is, a `year` earlier than 1583)
- variable `year` not numeric
- variable `day` not valid for variable `month` (for example, suppose variable `day` has a value of `31` and variable `month` has a value of `4`)

The output should be formatted as shown in the instructor's version of this program. Use the same phrasing and make sure you choose the correct verb tense, either "was", "is", or "will be". Look at the leap year example to help with this. The reference for the `getdate()` function is

$$http://php.net/manual/en/function.getdate.php$$

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.

- Add an echo statement to the beginning of the script section that will display your name, the course number, and the assignment number.

- 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 lab16.php
2   No syntax errors detected in lab16.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     <img src="https://www.w3.org/QA/Tools/I_heart_validator"
7      alt="I heart Validator logo" height="31" width="80" />
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. Only submit your PHP source; do **not** submit the form (your program will be invoked via a URL).