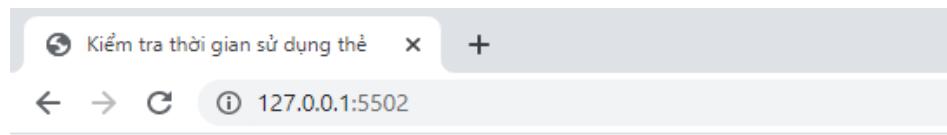


**Bài 1.** Hãy xây dựng giao diện trang web như hình bên dưới. Kiểm tra thời gian sử dụng thẻ và hiển thị thông báo về thời gian thẻ còn hạn sử dụng.



## Kiểm Tra Thời Gian Sử Dụng Thẻ Và Hiển Thị Thông Báo

Chọn ngày đăng ký thẻ:

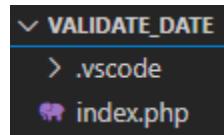
Chọn ngày hết hạn thẻ:

### Kết quả:

**Thẻ của bạn còn thời hạn: 5 năm, 3 tháng, và 28 ngày.**

Gợi ý:

+ B1. Tạo cấu trúc ứng dụng web.



+ B2. Viết mã xử lý kiểm tra ngày sử dụng thẻ tín dụng.

```
<!doctype html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Kiểm tra thời gian sử dụng thẻ</title>
</head>
<body>
    <?php
        $action = filter_input(INPUT_POST, 'action');
        $message = '';
        if(!empty($action) && $action == 'validation')
        {
            $date1 = filter_input(INPUT_POST, 'date1');
            $date2 = filter_input(INPUT_POST, 'date2');
```

```
if($date1 !== false && $date2 !== false)
{
    $now = new DateTime($date1);
    $exp = new DateTime($date2);
    $diff = $now->diff($exp);

    // Display a message
    $result = $diff->format('%y năm, %m tháng, và %d ngày');

    if ($diff->format('%R') == '-') {
        $message .= '<h3 style="color:red">Thẻ của bạn đã hết hạn cách đây: ' . $result . '</h3>';
    } else {
        $message .= '<h3 style="color:red">Thẻ của bạn còn thời hạn: ' . $result . '</h3>';
    }
}

?>

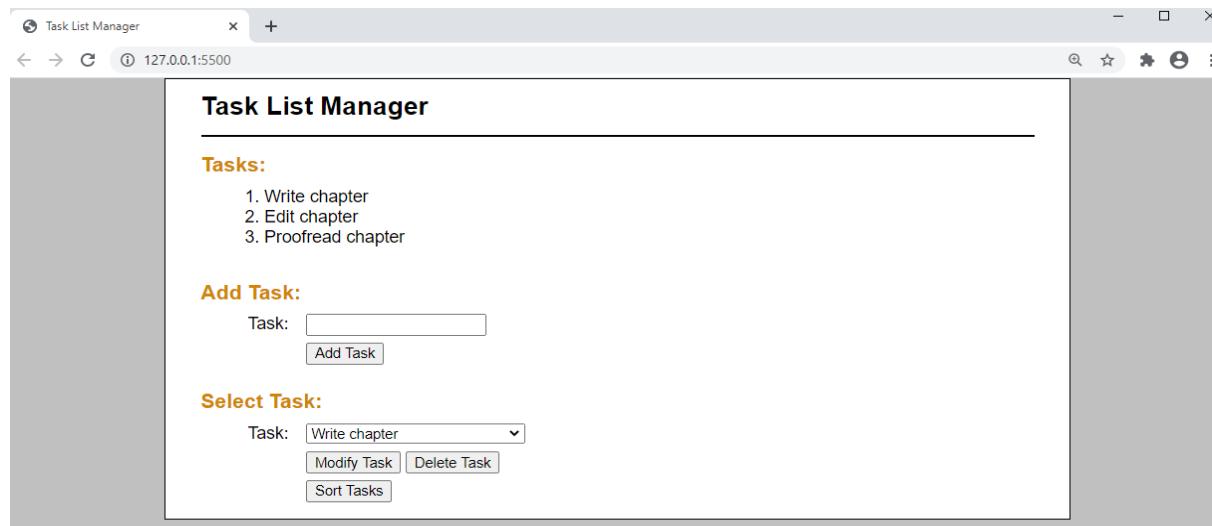
<h3 style="color:blue">Kiểm Tra Thời Gian Sử Dụng Thẻ Và Hiển Thị Thông Báo</h3>
<form action=<?php htmlspecialchars($_SERVER['PHP_SELF']); ?>" method="post">
    Chọn ngày đăng ký thẻ:
    <input type="date" name="date1" value=<?php if(isset($date1)) echo $date1;?>> <br/><br/>
    Chọn ngày hết hạn thẻ:
    <input type="date" name="date2" value=<?php if(isset($date2)) echo $date2;?>> <br/><br/>
    <input type="hidden" name="action" value="validation">
    <input type="submit" value="Kiểm Tra">
    <h2>Kết quả:</h2>
    <?php !empty($message)?print($message):""; ?>
</form>
</body>
</html>
```

**Bài 02.** Viết mã script php, thực hiện các thao tác cơ bản về mảng. Tạo mảng arr gồm 7 phần tử. Hãy thực hiện các thao tác sau trên mảng arr như sau :

- Tạo mảng một chiều arr có 7 phần tử.  
(chú ý: có thể tạo mảng chỉ mục hoặc mảng kết hợp)
- Duyệt mảng arr có 7 phần tử
- Xóa một phần tử đầu mảng và cuối mảng.  
(sử dụng các hàm ví dụ: array\_pop(), array\_shift(), ...)
- Chèn thêm một phần tử vào vị trí đầu mảng và cuối mảng.  
(sử dụng các hàm ví dụ: array\_push(), array\_unshift(), ...)
- Tìm kiếm giá trị x có trong mảng arr hay không.  
(sử dụng các hàm ví dụ: array\_in() ...)
- Đảo ngược các giá trị mảng arr.  
(sử dụng các hàm ví dụ: array\_reverse() ...)
- Sắp xếp mảng tăng dần hoặc giảm dần theo giá trị và khóa (key) của mảng arr.  
(sử dụng các hàm ví dụ: sort(), rsort(), asort(), arsort(), ksort(), krsort() ...)

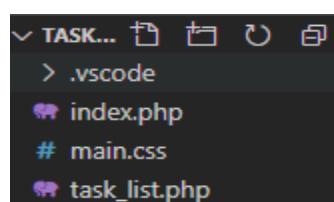
**Bài 03.** Viết mã script php, tạo ma trận số nguyên A có 3 hàng, 5 cột và hiển thị ma trận A.

**Bài 04.** Hãy xây dựng ứng dụng quản lý danh sách các công việc như giao diện bên dưới. Sử dụng lý thuyết về mảng đã học. Thực hiện thao tác thêm, xóa, sửa, sắp xếp các công việc.



**Gợi ý:**

- **Bước 1.** Tạo cấu trúc ứng dụng web.



**- Bước 2.** Tạo trang **main.css**

```
/* the styles for the HTML elements */  
  
html {  
    background-color: rgb(192, 192, 192);  
}  
  
body {  
    font-family: Arial, Helvetica, sans-serif;  
    width: 760px;  
    margin: 0 auto;  
    padding: 0 2em;  
    background-color: white;  
    border: 1px solid black;  
}  
  
header {  
    border-bottom: 2px solid black;  
    margin: .5em 0 1em;  
}  
  
header h1 {  
    color: black;  
}  
  
main {  
    margin-bottom: 1em;  
}  
  
h1 {  
    font-size: 150%;  
    margin: .5em 0;  
}  
  
h2 {  
    font-size: 120%;  
    margin: .25em 0 .5em;  
}  
  
h1, h2 {  
    color: rgb(208, 133, 4);  
}
```

```
ul {  
    margin: 0 0 .5em 0;  
    padding: 0 0 0 2.5em;  
}  
  
li {  
    margin: 0;  
    padding: 0;  
    list-style-type: none;  
}  
  
a {  
    color: rgb(41, 64, 124);  
    font-weight: bold;  
}  
  
a:hover {  
    color: rgb(208, 133, 4);  
}  
  
p {  
    margin: 0;  
    padding-bottom: .5em;  
}  
  
form {  
    margin: 0;  
    padding: 0;  
}  
  
input[text], select {  
    width: 15em;  
}  
  
label {  
    width: 5em;  
    float: left;  
    margin-right: 1em;  
    margin-bottom: .5em;  
    text-align: right;  
}
```

```
br {  
    clear: left;  
}
```

### - **Bước 3.** Xây dựng giao diện trang Task List (**task\_list.php**)

```
<!DOCTYPE html>  
<html>  
<head>  
    <title>Task List Manager</title>  
    <link rel="stylesheet" type="text/css" href="main.css">  
</head>  
<body>  
    <header>  
        <h1>Task List Manager</h1>  
    </header>  
    <main>  
        <!-- part 1: the errors -->  
        <?php if (count($errors) > 0) : ?>  
        <h2>Errors:</h2>  
        <ul>  
            <?php foreach($errors as $error) : ?>  
            <li><?php echo $error; ?></li>  
        <?php endforeach; ?>  
        </ul>  
        <?php endif; ?>  
        <!-- part 2: the tasks -->  
        <h2>Tasks:</h2>  
        <?php if (count($task_list) == 0) : ?>  
        <p>There are no tasks in the task list.</p>  
        <?php else: ?>  
        <ul>  
            <?php foreach( $task_list as $id => $task ) : ?>  
            <li><?php echo $id + 1 . ' ' . $task; ?></li>  
        <?php endforeach; ?>
```

```
</ul>

<?php endif; ?>

<br>

<!-- part 3: the add form --&gt;

&lt;h2&gt;Add Task:&lt;/h2&gt;

&lt;form action="." method="post" &gt;

&lt;?php foreach( $task_list as $task ) : ?&gt;

&lt;input type="hidden" name="tasklist[]" value="&lt;?php echo $task; ?&gt;"&gt;

&lt;?php endforeach; ?&gt;

&lt;label&gt;Task:&lt;/label&gt;

&lt;input type="text" name="newtask" id="newtask"&gt; &lt;br&gt;

&lt;label&gt;&amp;nbsp;&lt;/label&gt;

&lt;input type="submit" name="action" value="Add Task"&gt;

&lt;/form&gt;

&lt;br&gt;

<!-- part 4: the modify/delete form --&gt;

&lt;?php if (count($task_list) &gt; 0 &amp;&amp; empty($task_to_modify)) : ?&gt;

&lt;h2&gt;Select Task:&lt;/h2&gt;

&lt;form action="." method="post" &gt;

&lt;?php foreach( $task_list as $task ) : ?&gt;

&lt;input type="hidden" name="tasklist[]" value="&lt;?php echo $task; ?&gt;"&gt;

&lt;?php endforeach; ?&gt;

&lt;label&gt;Task:&lt;/label&gt;

&lt;select name="taskid"&gt;

&lt;?php foreach( $task_list as $id =&gt; $task ) : ?&gt;

&lt;option value="&lt;?php echo $id; ?&gt;"&gt;

&lt;?php echo $task; ?&gt;

&lt;/option&gt;

&lt;?php endforeach; ?&gt;

&lt;/select&gt;

&lt;br&gt;

&lt;label&gt;&amp;nbsp;&lt;/label&gt;</pre>
```

```

<input type="submit" name="action" value="Modify Task">
<input type="submit" name="action" value="Delete Task">
<br>
<label>&nbsp;</label>
<input type="submit" name="action" value="Sort Tasks">
</form>
<?php endif; ?>
<!-- part 5: the modify form --&gt;
&lt;?php if (!empty($task_to_modify)) : ?&gt;
&lt;h2&gt;Task to Modify:&lt;/h2&gt;
&lt;form action"." method="post" &gt;
&lt;?php foreach( $task_list as $task ) : ?&gt;
    &lt;input type="hidden" name="tasklist[]" value="&lt;?php echo $task; ?&gt;"&gt;
&lt;?php endforeach; ?&gt;
&lt;label&gt;Task:&lt;/label&gt;
&lt;input type="hidden" name="modifiedtaskid" value="&lt;?php echo $task_index; ?&gt;"&gt;
&lt;input type="text" name="modifiedtask" value="&lt;?php echo $task_to_modify; ?&gt;"&gt;&lt;br&gt;
&lt;label&gt;&amp;nbsp;&lt;/label&gt;
&lt;input type="submit" name="action" value="Save Changes"&gt;
&lt;input type="submit" name="action" value="Cancel Changes"&gt;
&lt;/form&gt;
&lt;?php endif; ?&gt;
&lt;/main&gt;
&lt;/body&gt;
&lt;/html&gt;
</pre>

```

- **Bước 4.** Xây dựng trang điều hướng các chức năng của ứng dụng, file **index.php** (controller)

```

<?php
//get tasklist array from POST
$task_list = filter_input(INPUT_POST, 'tasklist', FILTER_DEFAULT,
    FILTER_REQUIRE_ARRAY);
if ($task_list === NULL) {
    $task_list = array();
}

```

```

$task_list[] = 'Write chapter';
$task_list[] = 'Edit chapter';
$task_list[] = 'Proofread chapter';

}

//get action variable from POST
$action = filter_input(INPUT_POST, 'action');

//initialize error messages array
$errors = array();

//process
switch( $action ) {

    case 'Add Task':
        $new_task = filter_input(INPUT_POST, 'newtask');
        if (empty($new_task)) {
            $errors[] = 'The new task cannot be empty.';
        } else {
            // $task_list[] = $new_task;
            array_push($task_list, $new_task);
        }
        break;

    case 'Delete Task':
        $task_index = filter_input(INPUT_POST, 'taskid', FILTER_VALIDATE_INT);
        if ($task_index === NULL || $task_index === FALSE) {
            $errors[] = 'The task cannot be deleted.';
        } else {
            unset($task_list[$task_index]);
            $task_list = array_values($task_list);
        }
        break;

    case 'Modify Task':
        $task_index = filter_input(INPUT_POST, 'taskid', FILTER_VALIDATE_INT);
        if ($task_index === NULL || $task_index === FALSE) {
            $errors[] = 'The task cannot be modified.';
        } else {
            $task_to_modify = $task_list[$task_index];
        }
}

```

```
        }

        break;

case 'Save Changes':
    $i = filter_input(INPUT_POST, 'modifiedtaskid', FILTER_VALIDATE_INT);

    $modified_task = filter_input(INPUT_POST, 'modifiedtask');

    if (empty($modified_task)) {
        $errors[] = 'The modified task cannot be empty.';

    } elseif($i === NULL || $i === FALSE) {
        $errors[] = 'The task cannot be modified.';

    } else {
        $task_list[$i] = $modified_task;
        $modified_task = "";
    }
    break;

case 'Cancel Changes':
    $modified_task = "";
    break;

case 'Sort Tasks':
    sort($task_list);
    break;
}

include('task_list.php');
?>
```