

Lab 4. Working with Arrays

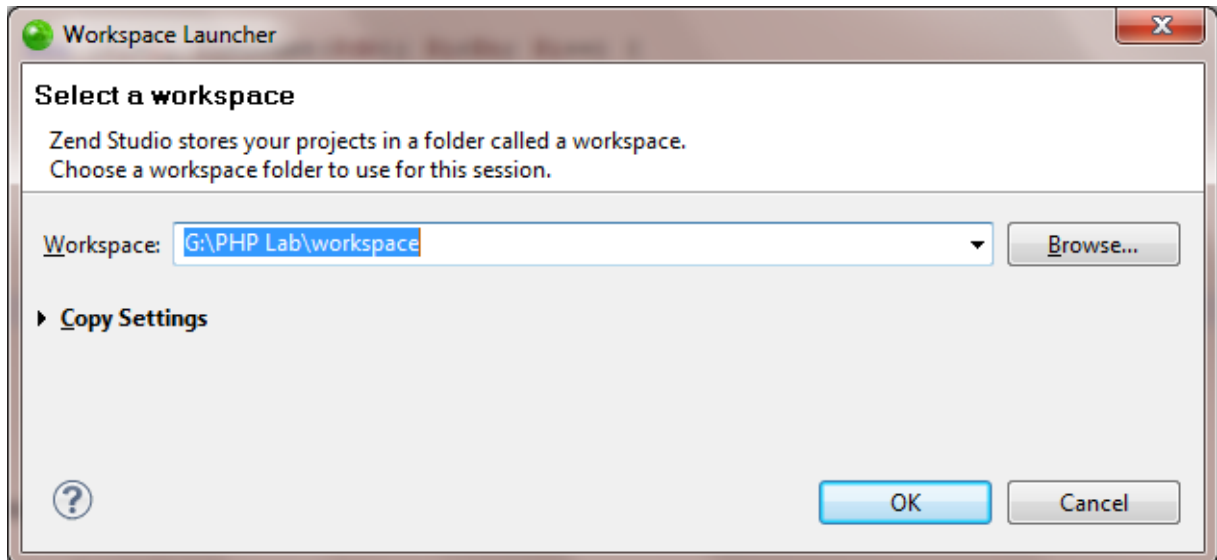
Prepared: TrangNTT

- Lab 4. Working with Arrays 1
 - 4.1. Open workspace and Create Folder 2
 - 4.2. Tuna Café..... 3
 - 4.3. Drive Distance..... 5
 - 4.4. Exercise 1 - User Sorting..... 6
 - 4.5. Exercise 2 – File list 8

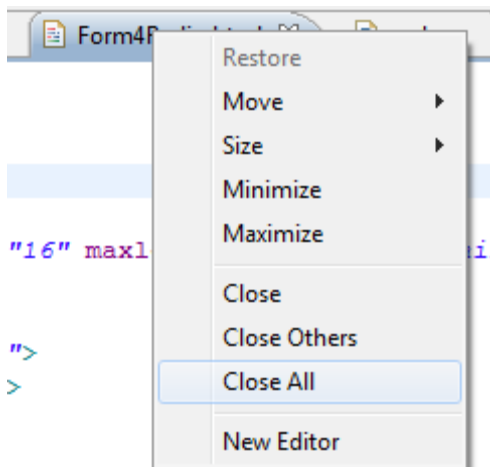
4.1. Open workspace and Create Folder

Step 1. Open Zend Studio 7.0

- Select Start → All Programs → Zend Studio – 7.0.0 → Zend Studio – 7.0.0
- Choose OK to confirm the workspace.



- Close all the opening file in the **PHP Editor** view by right click to a file and choose **Close All**.



Step 2. Right click on LabProject, choose New Folder. Enter Lab4 as folder name.

From now, put all files in this lab to Lab4 folder.

4.2. Tuna Café

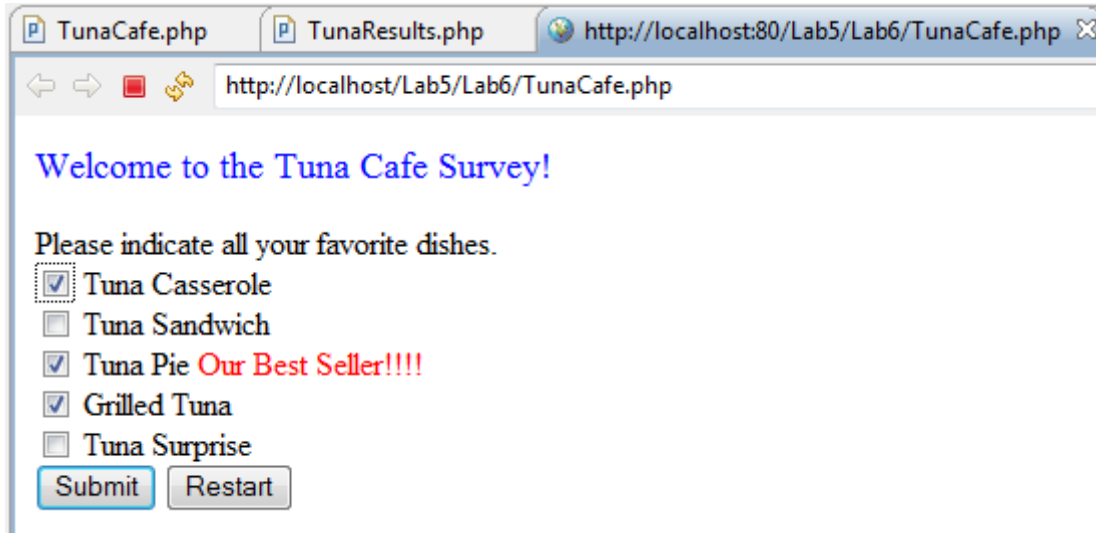
Step 1. TunaCafe.php

```
TunaCafe.php X TunaResults.php http://localhost:80/Lab5/Lab6/TunaCafe.php
1 <html>
2 <head><title> Tuna Cafe </title></head>
3 <body><font size=4 color="blue">
4 Welcome to the Tuna Cafe Survey! </font>
5 <form action="TunaResults.php" method="GET">
6 <?php
7     $menu = array('Tuna Casserole', 'Tuna Sandwich',
8                   'Tuna Pie', 'Grilled Tuna', 'Tuna Surprise');
9     $bestseller = 2;
10    print 'Please indicate all your favorite dishes.<br>';
11    for ($i=0; $i < count($menu); $i++) {
12        print "<input type=\"checkbox\" name=\"prefer[]\" value=\"$i\" $menu[$i]";
13        if ($i == $bestseller) {
14            print '<font color=red> Our Best Seller!!!! </font>';
15        }
16        print '<br>';
17    }
18    ?>
19    <input type="submit" value="Submit">
20    <input type="reset" value="Restart">
21 </form>
22 </body>
23 </html>
```

Step 2. TunaResults.php

```
TunaCafe.php TunaResults.php X http://localhost:80/Lab5/Lab6/TunaCafe.php
1 <html>
2 <head><title> Tuna Cafe </title></head>
3 <body>
4 <font size=4 color="blue"> Tuna Cafe Results Received </font>
5 <?php
6     $menu = array('Tuna Casserole', 'Tuna Sandwich', 'Tuna Pie', 'Grilled Tuna', 'Tuna Surprise');
7
8     $prefer = $_GET[prefer];
9     if (count($prefer) == 0 ) {
10        print 'Oh no! Please pick something as your favorite! ';
11    } else {
12        print '<br>Your selections were <ul>';
13        foreach ($prefer as $item) {
14            print "<li>$menu[$item]</li>";
15        }
16        print '</ul>';
17    }
18    ?>
19 </body>
20 </html>
```

Step 3. Run



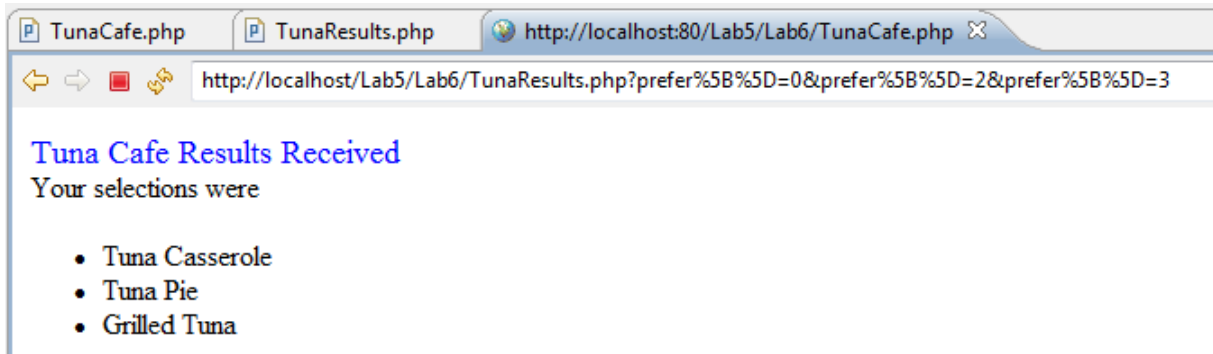
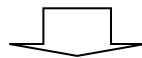
A screenshot of a web browser window. The address bar shows 'http://localhost/Lab5/Lab6/TunaCafe.php'. The page title is 'Tuna Cafe Survey!'. The text 'Please indicate all your favorite dishes.' is followed by a list of five items, each with a checkbox: 'Tuna Casserole' (checked), 'Tuna Sandwich' (unchecked), 'Tuna Pie Our Best Seller!!!!' (checked), 'Grilled Tuna' (checked), and 'Tuna Surprise' (unchecked). At the bottom are 'Submit' and 'Restart' buttons.

http://localhost/Lab5/Lab6/TunaCafe.php

Welcome to the Tuna Cafe Survey!

Please indicate all your favorite dishes.

- ☒ Tuna Casserole
- ☐ Tuna Sandwich
- ☒ Tuna Pie Our Best Seller!!!!
- ☒ Grilled Tuna
- ☐ Tuna Surprise



A screenshot of a web browser window. The address bar shows 'http://localhost/Lab5/Lab6/TunaResults.php?prefer%5B%5D=0&prefer%5B%5D=2&prefer%5B%5D=3'. The page title is 'Tuna Cafe Results Received'. The text 'Your selections were' is followed by a bulleted list of three items: 'Tuna Casserole', 'Tuna Pie', and 'Grilled Tuna'.

http://localhost/Lab5/Lab6/TunaResults.php?prefer%5B%5D=0&prefer%5B%5D=2&prefer%5B%5D=3

Tuna Cafe Results Received

Your selections were

- Tuna Casserole
- Tuna Pie
- Grilled Tuna

4.3. Drive Distance

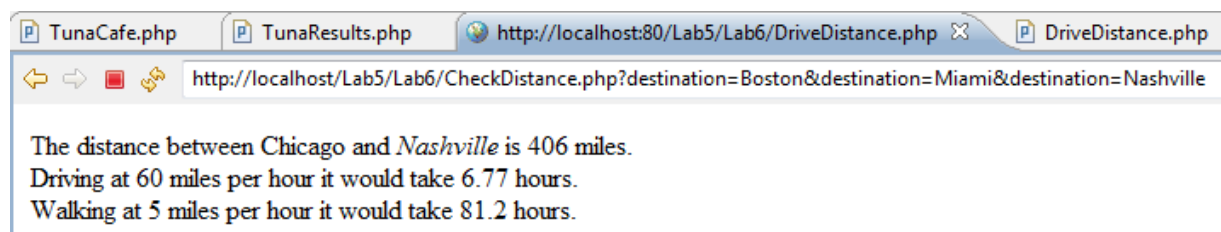
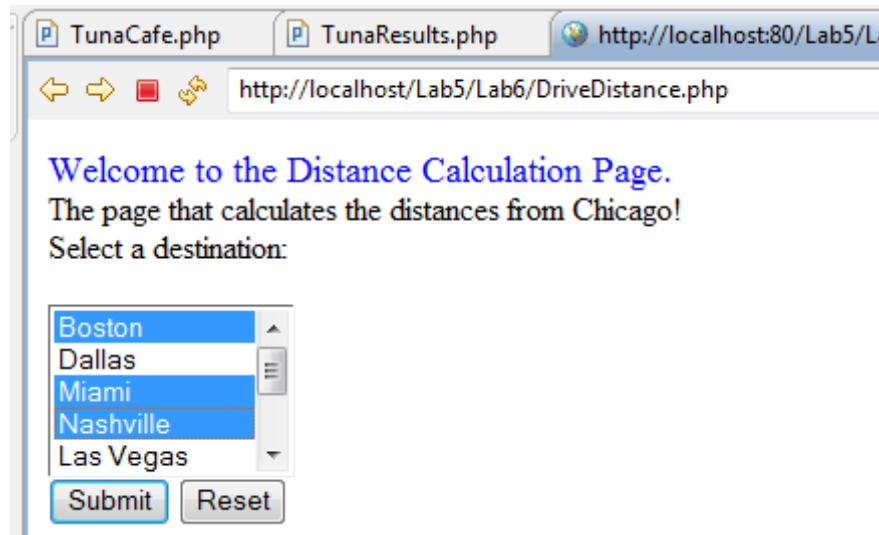
Step 1. DriveDistance.php

```
TunaCafe.php TunaResults.php http://localhost:80/Lab5/Lab6/
1 <HTML>
2 <HEAD><TITLE> Distance from Chicago!</TITLE></HEAD>
3 <BODY>
4 <FONT SIZE=4 COLOR=BLUE> Welcome to the Distance Ca
5 <br>The page that calculates the distances from Chi
6 <BR>Select a destination:
7 <FORM ACTION="CheckDistance.php" METHOD="GET">
8   <SELECT NAME="destination" SIZE=5 MULTIPLE>
9     <OPTION> Boston </OPTION>
10    <OPTION> Dallas </OPTION>
11    <OPTION> Miami </OPTION>
12    <OPTION> Nashville </OPTION>
13    <OPTION> Las Vegas </OPTION>
14    <OPTION> Pittsburgh </OPTION>
15    <OPTION> San Francisco </OPTION>
16    <OPTION> Toronto </OPTION>
17    <OPTION> Washington, DC </OPTION>
18  </SELECT>
19  <BR>
20  <INPUT TYPE="SUBMIT" VALUE="Submit">
21  <INPUT TYPE="RESET" VALUE="Reset" >
22 </FORM></BODY></HTML>
```

Step 2. CheckDistance.php

```
TunaCafe.php TunaResults.php http://localhost:80/Lab5/Lab6/DriveDistance.php DriveDistance.php CheckDistance.php
1 <html>
2   <head><title> Distance and Time Calculations </title></head>
3   <body>
4     <?php
5       $cities = array ('Dallas' => 803, 'Toronto' => 435, 'Boston' => 848,
6                       'Nashville' => 406, 'Las Vegas' => 1526,
7                       'San Francisco' => 1835, 'Washington, DC' => 595,
8                       'Miami' => 1189, 'Pittsburgh' => 409);
9       $destination = $_GET[destination];
10      if (isset($cities[$destination])) {
11        $distance = $cities[$destination];
12        $time = round( ($distance / 60), 2);
13        $walktime = round( ($distance / 5), 2);
14        print "The distance between Chicago and <I>$destination</I> is $distance miles.";
15        print "<br>Driving at 60 miles per hour it would take $time hours.";
16        print "<br>Walking at 5 miles per hour it would take $walktime hours.";
17      } else {
18        print "Sorry, do not have destination information for $destination.";
19      }
20    }
21  </body>
22 </html>
```

Step 3. Run



Step 4. Note that although the user selects 3 destination, the result only calculate for the last destination. Modify CheckDistance.php to calculate and display information for all destinations, each destination is located in a row of a table (the table has 5 columns: No., Destination, Distance, Driving time (60 mph), Walking time (5 mph)).

From Chicago to:

No.	Destination	Distance	Driving time	Walking time
1	Boston	848	14.13	169.6
2	Dallas	803	13.38	160.6
3	Las Vegas	1526	25.43	305.2

4.4. Exercise 1 - User Sorting

Given following suggestion code.

```
<?php

function user_sort($a, $b) {
    // smarts is all-important, so sort it first
    if($b == 'smarts') {
        return 1;
    }
    else if($a == 'smarts') {
        return -1;
    }
    return ($a == $b) ? 0 : (($a < $b) ? -1 : 1);
}
```

```

$values = array('name' => 'Buzz Lightyear',
                'email_address' => 'buzz@starcommand.gal',
                'age' => 32,
                'smarts' => 'some');

if($submitted) {
    if($sort_type == 'usort' || $sort_type == 'uksort' || $sort_type == 'uasort') {
        $sort_type($values, 'user_sort');
    }
    else {
        $sort_type($values);
    }
}
?>

<form action="UserSorting.php" method="post">
<p>
    <input type="radio" name="sort_type" value="sort" checked="checked" />
    Standard sort<br />
    <input type="radio" name="sort_type" value="rsort" /> Reverse sort<br />
    <input type="radio" name="sort_type" value="usort" /> User-defined sort<br />
    <input type="radio" name="sort_type" value="ksort" /> Key sort<br />
    <input type="radio" name="sort_type" value="krsort" /> Reverse key sort<br />
    <input type="radio" name="sort_type" value="uksort" /> User-defined key sort<br />
/>
    <input type="radio" name="sort_type" value="asort" /> Value sort<br />
    <input type="radio" name="sort_type" value="arsort" /> Reverse value sort<br />
    <input type="radio" name="sort_type" value="uasort" /> User-defined value
sort<br />
</p>

<p align="center">
    <input type="submit" value="Sort" name="submitted" />
</p>

<p>
    Values <?= $submitted ? "sorted by $sort_type" : "unsorted"; ?>:
</p>

<ul>
    <?php
        foreach($values as $key=>$value) {
            echo "<li><b>$key</b>: $value</li>";
        }
    ?>
</ul>
</form>

```

Modify to run it in one PHP file allowing users to sort the data as they choose in the radio button.

The following result is the example:

Standard sort
Reverse sort
User-defined sort
Key sort
Reverse key sort
User-defined key sort
Value sort
Reverse value sort
User-defined value sort

Sort

Values unsorted (before sort):

- name: Buzz Lightyear
- email_address: buzz@starcommand.gal
- age: 32
- smarts: some

↓

http://localhost/Lab5/Lab6/UserSorting.php?sort_type=usort&submitted=Sort

☐ Standard sort
 ☒ User-defined sort
 ☐ Reverse key sort
 ☐ Value sort

☐ Reverse sort
 ☐ Key sort
 ☐ User-defined key sort
 ☐ Reverse value sort

☐ User-defined value sort

Values unsorted (before sort):

- **name:** Buzz Lightyear
- **email_address:** buzz@starcommand.gal
- **age:** 32
- **smarts:** some

Values sorted by usort:

- **0:** Buzz Lightyear
- **1:** buzz@starcommand.gal
- **2:** some
- **3:** 32

↓

http://localhost/Lab5/Lab6/UserSorting.php?sort_type=asort&submitted=Sort

☐ Standard sort
 ☐ User-defined sort
 ☐ Reverse key sort
 ☒ Value sort

☐ Reverse sort
 ☐ Key sort
 ☐ User-defined key sort
 ☐ Reverse value sort

☐ User-defined value sort

Values unsorted (before sort):

- **name:** Buzz Lightyear
- **email_address:** buzz@starcommand.gal
- **age:** 32
- **smarts:** some

Values sorted by asort:

- **name:** Buzz Lightyear
- **email_address:** buzz@starcommand.gal
- **smarts:** some
- **age:** 32

4.5. Exercise 2 – File list

Write a web page allowing users:

- View the detail information of uploaded files in a folder (create a folder in the project named “upload” and copy some files to that folder) including: File Name, Type, Uploaded Date, Size
- Sort list of file by File Name or Uploaded Date.