CO551
Open Source Systems

Logbook

21606555

28th March 2018

Computing & Web Development
Buckinghamshire New University

Summary

Section 1

Week 1

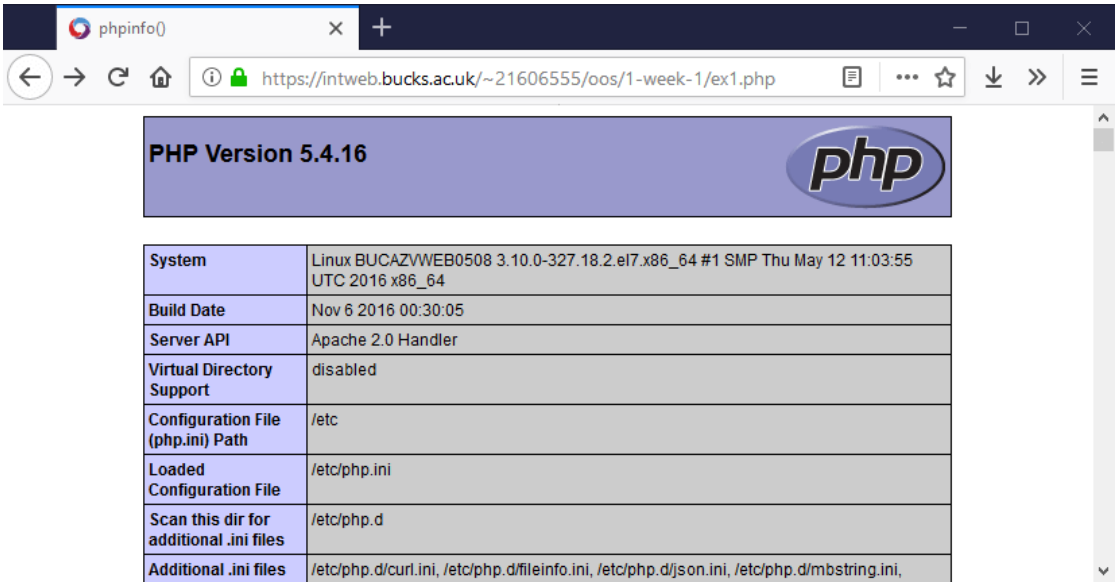
1.1 Exercise 1

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex1.php

Figure 1.1: ex1.php

```
1 <?php phpinfo(); ?>
```

Figure 1.2: Output of exercise 1



System	Linux BUCAZVWEB0508 3.10.0-327.18.2.el7.x86_64 #1 SMP Thu May 12 11:03:55 UTC 2016 x86_64
Build Date	Nov 6 2016 00:30:05
Server API	Apache 2.0 Handler
Virtual Directory Support	disabled
Configuration File (php.ini) Path	/etc
Loaded Configuration File	/etc/php.ini
Scan this dir for additional .ini files	/etc/php.d
Additional .ini files	/etc/php.d/curl.ini, /etc/php.d/fileinfo.ini, /etc/php.d/json.ini, /etc/php.d/mbstring.ini,

1.2 Exercise 2

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex2.php

Figure 1.3: ex2.php

```
User Agent:
<?php echo $_SERVER["HTTP_USER_AGENT"]; ?>
<br><br>
Name:
<?php echo $_SERVER["SERVER_NAME"]; ?>
<br><br>
Protocol:
<?php echo $_SERVER["SERVER_PROTOCOL"]; ?>
```

Figure 1.4: Output of exercise 2

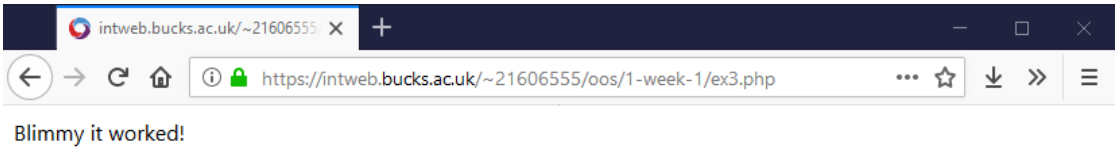
1.3 Exercise 3

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex3.php

Figure 1.5: ex3.php

```
<html>
  <body>
    <?php echo "Blimmy it worked!"; ?>
  </body>
</html>
```

Figure 1.6: Output of exercise 3



1.4 Exercise 4

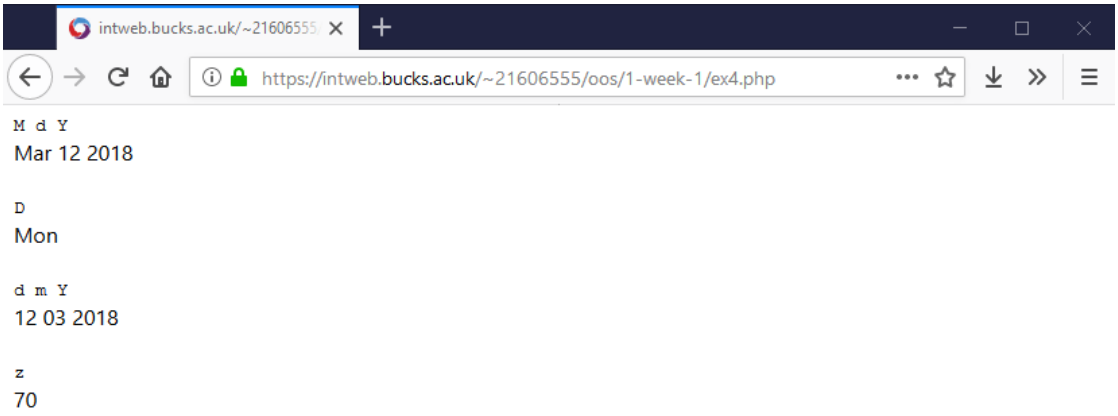
The last argument of `gmddate()` allows you to provide a specific date to format. This argument is optional and will use the current date and time if it is not provided.

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex4.php

Figure 1.7: ex4.php

```
<html>
  <body>
    <code>M d Y</code><br>
    <?php echo gmdate("M d Y"); ?>

    <br><br><code>D</code><br>
    <?php echo gmdate("D"); ?>

    <br><br><code>d m Y</code><br>
    <?php echo gmdate("d m Y"); ?>

    <br><br><code>z</code><br>
    <?php echo gmdate("z"); ?>
  </body>
</html>
```

Figure 1.8: Output of exercise 4



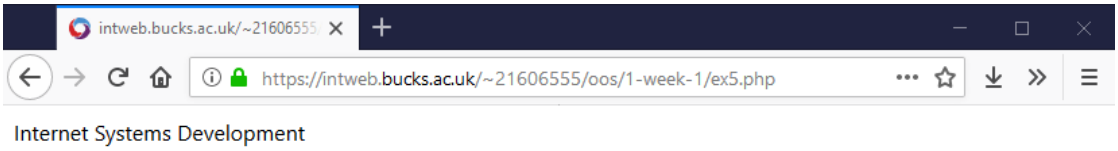
1.5 Exercise 5

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex5.php

Figure 1.9: ex5.php

```
<html>
  <body>
    <?php
      $myFavouriteModule = "Internet Systems Development";
      echo $myFavouriteModule;
    ?>
  </body>
</html>
```

Figure 1.10: Output of exercise 5



1.6 Exercise 6

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex6.php

Figure 1.11: ex6.php

```
<html>
  <body>
    <?php
      $firstName = "Richard";
      $lastName = "Mather";
      $space = " ";
      $name = $firstName . $space . $lastName;
      echo $name;
    ?>
  </body>
</html>
```

Figure 1.12: Output of exercise 6

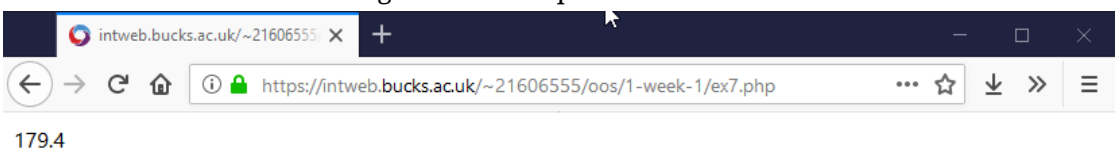
1.7 Exercise 7

https://intweb.bucks.ac.uk/~21606555/oos/1-week-1/ex7.php

Figure 1.13: ex7.php

```
<html>
  <body>
    <?php
      $hourlyRate = 5.75;
      $hoursPerWeek = 40;
      $gross = $hourlyRate * $hoursPerWeek;

      $taxRate = 1 - 0.22;
      $netWage = $gross * $taxRate;
      echo $netWage;
    ?>
  </body>
</html>
```

Figure 1.14: Output of exercise 7



Section 2

Syntax

2.1 Exercise 1

Converted the shorthand PHP tags (`<?...?>`) to `<?php...?>` as IntWeb has disabled the language feature.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex1.php

Figure 2.1: ex1.php

```
<?php
  $hourlyRate = 5.75;
  $hoursPerWeek = 40;
  $gross = $hourlyRate * $hoursPerWeek;
?>

<html>
  <body>
    <p>My gross wage is <?php print "$gross"; ?></p>
  </body>
</html>
```

Figure 2.2: Output of exercise 1

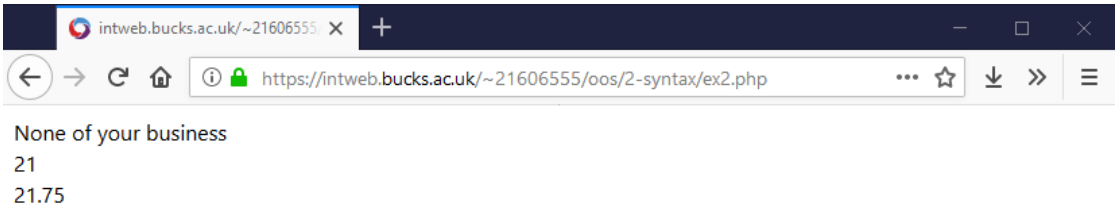
2.2 Exercise 2

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex2.php

Figure 2.3: ex2.php

```php
<?php
  $myAge = "None of your business<br>";
  print $myAge;

  $myAge = 21;
  print $myAge . "<br>";

  $myAge = 21.75;
  print $myAge;
?>
```

Figure 2.4: Output of exercise 2



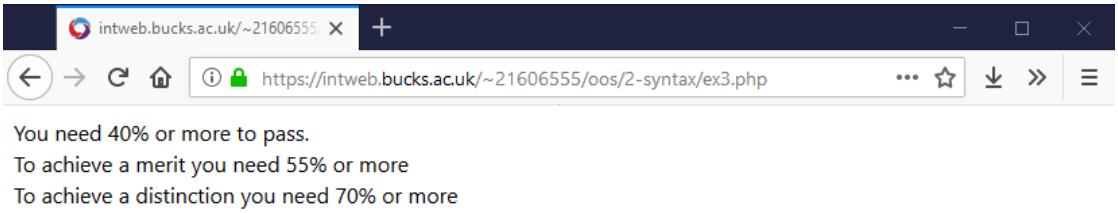
2.3 Exercise 3

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex3.php

Figure 2.5: ex3.php

```php
<?php
  define("PASS_GRADE", 40);

  echo "You need " . PASS_GRADE . "% or more to pass.<br>";

  $grade = PASS_GRADE + 15;
  echo "To achieve a merit you need $grade% or more<br>";

  $grade = PASS_GRADE + 30;
  echo "To achieve a distinction you need $grade% or more<br>";
?>
```

Figure 2.6: Output of exercise 3



2.4 Exercise 4

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex4.php

Figure 2.7: ex4.php

```
<!-- Speed camera advise script -->

<?php
  $points = 13;
  // Output appropriate message depending on points awarded.
  if ($points >= 12) {
    echo "Public transport is your best option.<br>";
  } elseif ($points >= 9) {
    echo "If you get caught you say your grandmother was driving.<br>";
  } else {
    echo "There is no need to worry about the speed limit.<br>";
  }
?>
```

Figure 2.8: Output of exercise 4

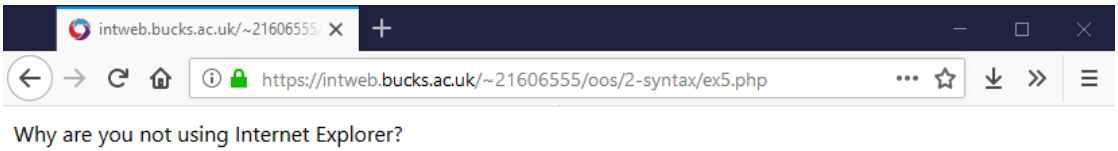
2.5 Exercise 5

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex5.php

Figure 2.9: ex5.php

```php
<?php
  if (strpos($_SERVER["HTTP_USER_AGENT"], "MSIE") != false) {
    echo "You are using Internet Explorer";
  } else {
    echo "Why are you not using Internet Explorer?";
  }
?>
```

Figure 2.10: Output of exercise 5



2.6 Exercise 6

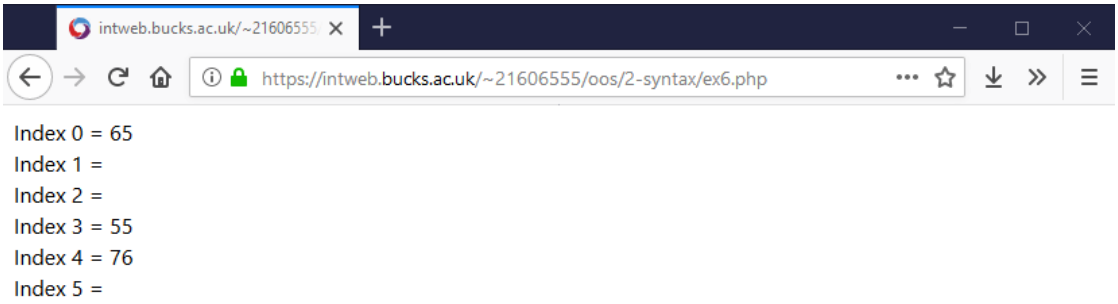
If an index position isn't specified when assigning a value to an array then the value is inserted after the last index position in the array. In this case, 76 is inserted into the 4th index.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex6.php

Figure 2.11: ex6.php

```php
<?php
  $marks[0] = 65;
  $marks[3] = 55;
  $marks[] = 76;

  echo "Index 0 = $marks[0]<br>";
  echo "Index 1 = $marks[1]<br>";
  echo "Index 2 = $marks[2]<br>";
  echo "Index 3 = $marks[3]<br>";
  echo "Index 4 = $marks[4]<br>";
  echo "Index 5 = $marks[5]<br>";
?>
```

Figure 2.12: Output of exercise 6



2.7 Exercise 7

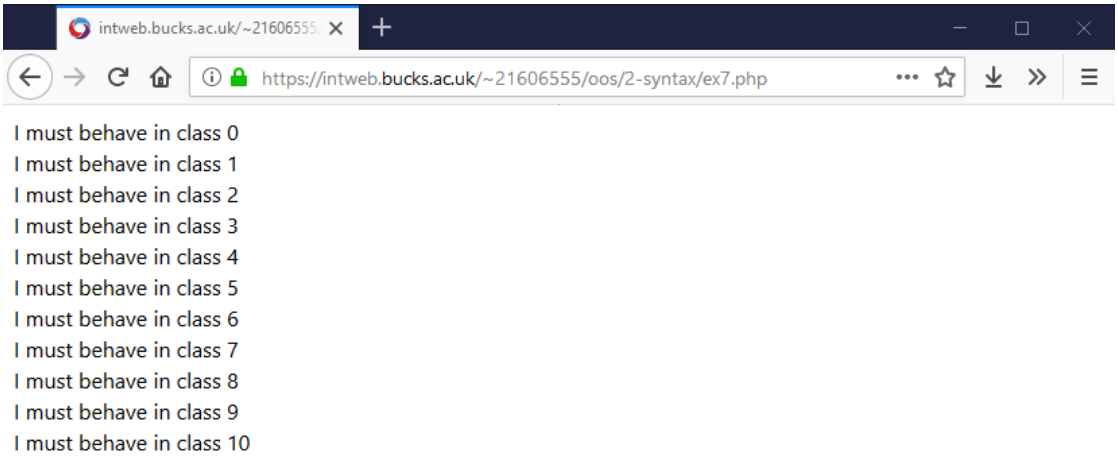
11 lines will be displayed as it will loop from 0 until it reaches 10.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex7.php

Figure 2.13: ex7.php

```php
<?php
  for($count = 0; $count <= 10; $count++) {
    echo "I must behave in class $count<br>";
  }
?>
```

Figure 2.14: Output of exercise 7



2.8 Exercise 8

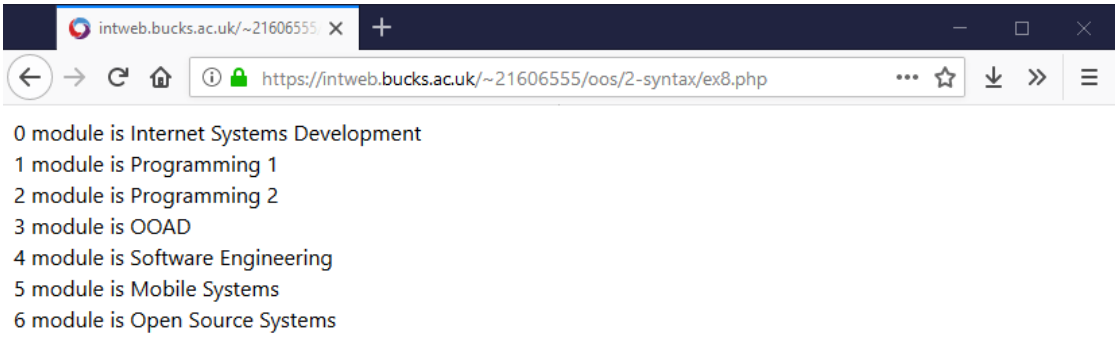
Fixed the for loop syntax where a comma was used instead of a semi-colon.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex8.php

Figure 2.15: ex8.php

```php
<?php
  $topModules[0] = "Internet Systems Development";
  $topModules[1] = "Programming 1";
  $topModules[2] = "Programming 2";
  $topModules[3] = "OOAD";
  $topModules[4] = "Software Engineering";
  $topModules[5] = "Mobile Systems";
  $topModules[6] = "Open Source Systems";

  for($count = 0; $count < sizeof($topModules); $count++) {
    echo "$count module is $topModules[$count]<br>";
  }
?>
```

Figure 2.16: Output of exercise 8



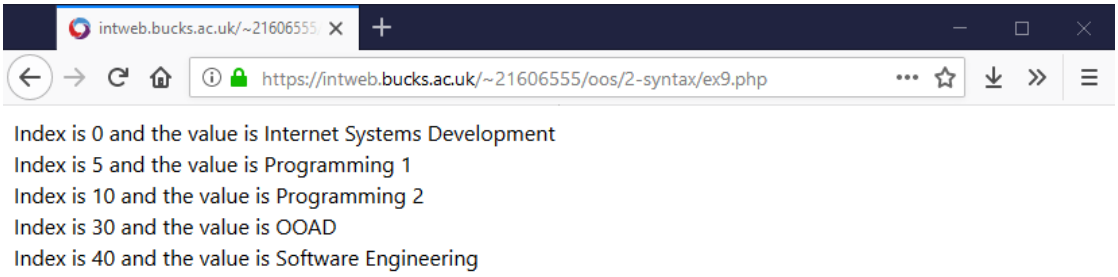
2.9 Exercise 9

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex9.php

Figure 2.17: ex9.php

```php
<?php
  $topModules[0] = "Internet Systems Development";
  $topModules[5] = "Programming 1";
  $topModules[10] = "Programming 2";
  $topModules[30] = "OOAD";
  $topModules[40] = "Software Engineering";

  while(list($index, $value) = each($topModules)) {
    echo "Index is $index and the value is $value<br>";
  }
?>
```

Figure 2.18: Output of exercise 9



2.10 Exercise 10

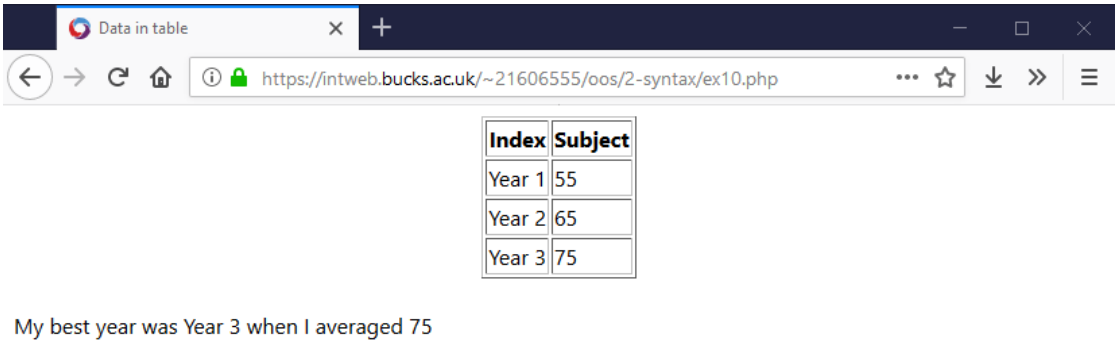
This has been revised to be displayed in a table from the exercise 12 task.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex10.php

Figure 2.19: ex10.php

```
<?php
  $myMarks["Year 1"] = 55;
  $myMarks["Year 2"] = 65;
  $myMarks["Year 3"] = 75;
?>

<html>
  <head>
    <title>Data in table</title>
  </head>
  <body>
    <table border=1 align=center>
      <tr>
        <th>Index</th>
        <th>Subject</th>
      </tr>
      <?php
        while(list($index, $value) = each($myMarks)) {
          echo "<tr><td>$index</td><td>$value</td></tr>";
        }
      ?>
    </table>
    <?php echo "<br>My best year was Year 3 when I averaged " . $myMarks["Year 3"]; ?
  </body>
</html>
```

Figure 2.20: Output of exercise 10



2.11 Exercise 11

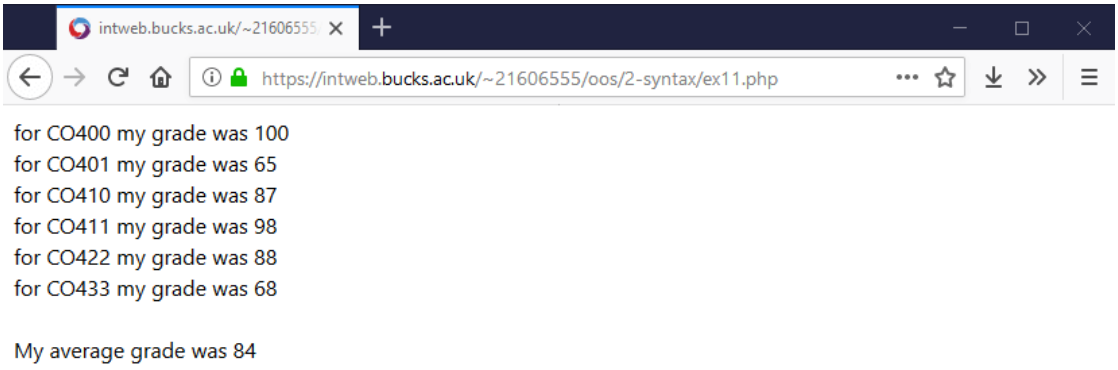
I opted to use the `sizeof` function to calculate the averages so that the average will always be correct if marks are removed or added to the array.

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex11.php

Figure 2.21: ex11.php

```php
<?php
  $myMarks["CO400"] = 100;
  $myMarks["CO401"] = 65;
  $myMarks["CO410"] = 87;
  $myMarks["CO411"] = 98;
  $myMarks["CO422"] = 88;
  $myMarks["CO433"] = 68;
  $total = 0;

  while(list($index, $value) = each($myMarks)) {
    $total += $value;
    echo "for $index my grade was $value<br>";
  }

  $average = round($total / sizeof($myMarks));
  echo "<br>My average grade was $average";
?>
```

Figure 2.22: Output of exercise 11



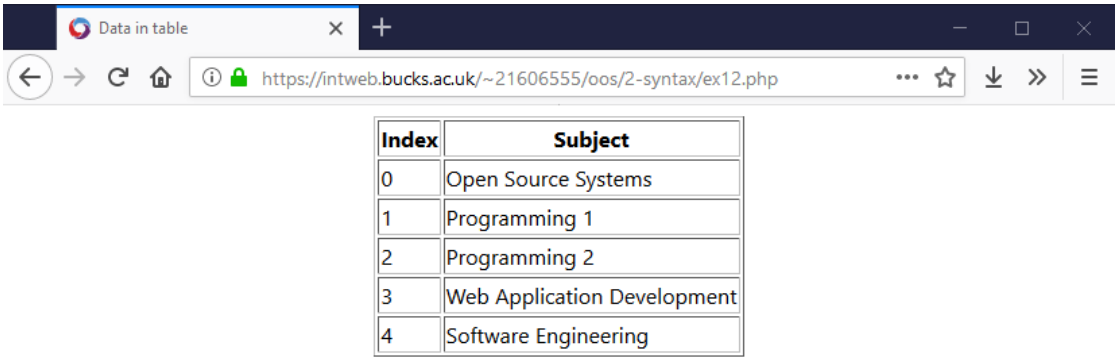
2.12 Exercise 12

https://intweb.bucks.ac.uk/~21606555/oos/2-syntax/ex12.php

Figure 2.23: ex12.php

```php
<?php
  $topModules[0] = "Open Source Systems";
  $topModules[1] = "Programming 1";
  $topModules[2] = "Programming 2";
  $topModules[3] = "Web Application Development";
  $topModules[4] = "Software Engineering";
?>
<html>
  <head>
    <title>Data in table</title>
  </head>
  <body>
    <table border=1 align=center>
      <tr>
        <th>Index</th>
        <th>Subject</th>
      </tr>
      <?php
        for ($count = 0; $count < sizeof($topModules); $count++) {
          echo "<tr><td>$count</td><td>$topModules[$count]</td></tr>";
        }
      ?>
    </table>
  </body>
</html>
```

Figure 2.24: Output of exercise 12



Index	Subject
0	Open Source Systems
1	Programming 1
2	Programming 2
3	Web Application Development
4	Software Engineering

Section 3

Forms

3.1 Exercise 1

https://intweb.bucks.ac.uk/~21606555/oos/3-forms/ex1.html

Figure 3.1: ex1.html

```
<html>
  <head>
    <title>HTML form submitted to php file</title>
  </head>
  <body>
    <form action='./ex1-action.php' method='post'>
      Enter your name
      <input type='text' name='txtname' size='30'>
      <br><br>Gender
      <input type='radio' name='radsex' value='male'>Male
      <input type='radio' name='radsex' value='female'>Female
      <input type='radio' name='radsex' value='other'>Other
      <br><br>Occupation
      <select name='seloccupation'>
        <option>Lecturer</option>
        <option>Beach Bum</option>
        <option>Traffic Warden</option>
      </select>
      <br><br>
      <input type='submit' value='display'>
    </form>
  </body>
</html>
```

Figure 3.2: ex1-action.php

```
<html>
  <head>
    <title>Response to form</title>
  </head>
  <body>
    <?php
      echo "Your name is $_POST[txtname]<br>";
      echo "Your gender is $_POST[radsex]<br>";
      echo "Your occupation is $_POST[seloccupation]<br>";
    ?>
  </body>
</html>
```

Figure 3.3: Input form for exercise 1

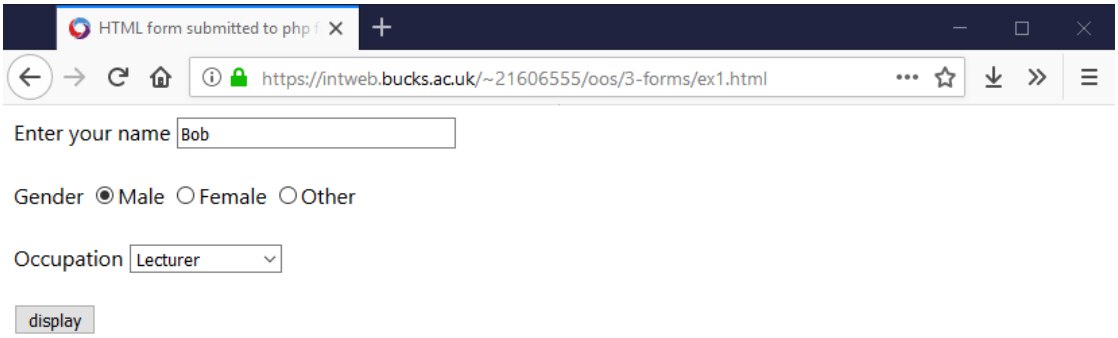
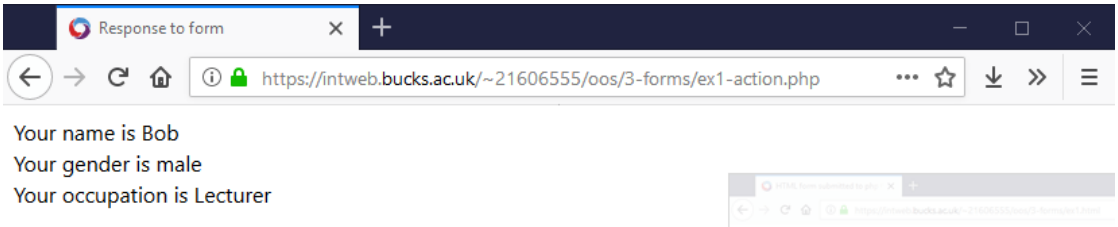


Figure 3.4: Output of exercise 1



3.2 Exercise 2

https://intweb.bucks.ac.uk/~21606555/oos/3-forms/ex2.html

Figure 3.5: ex2.html

```
<html>
  <head>
    <title>Selection Exercise 1</title>
  </head>
  <body>
    <form action='./ex2-action.php' method='post'>
      Enter your age:
      <input type='text' name='txtage'>
      <br><br>Gender
      <input type='radio' name='radsex' value='male'>Male
      <input type='radio' name='radsex' value='female'>Female
      <br>
      <input type='submit'>
    </form>
  </body>
</html>
```

Figure 3.6: ex2-action.php

```
<?php
  if ($_POST["txtage"] < 21) {
    echo "You are under 21 years old";
  } else {
    echo "You are 21 years old or over";
  }

  if ($_POST["radsex"] == "male") {
    echo " and male";
  } else {
    echo " and female";
  }
?>
```

Figure 3.7: Input form for exercise 2 showing male/under 21

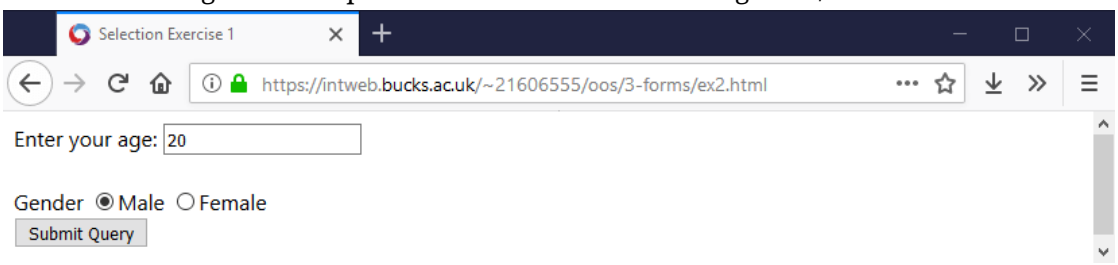


Figure 3.8: Output of exercise 2 showing male/under 21

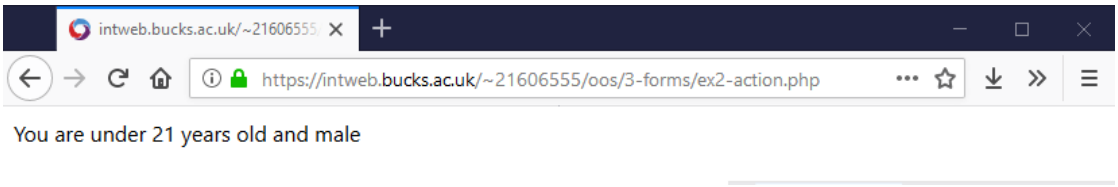


Figure 3.9: Input form for exercise 2 showing female/over 21

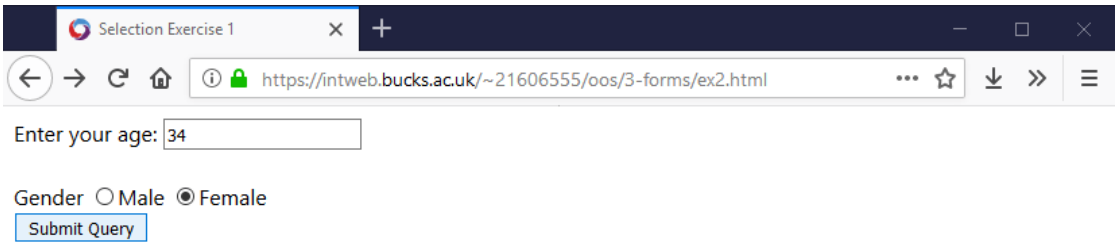


Figure 3.10: Output of exercise 2 showing female/over 21

3.3 Exercise 3

If all the breaks are removed from the switch statement then the correct case is still accessed (selecting Part Time prints the part time course and not HNC, HND or BSC). However it will also print the default switch case (no course selected).

This happens as the `break` keyword is used to exit the switch statement. If there is no break then it will keep going through the switch cases for anything else that matches - the default case matches anything.

https://intweb.bucks.ac.uk/~21606555/oos/3-forms/ex3.html

Figure 3.11: ex3.html

```
<html>
  <head>
    <title>Selection Exercise 2</title>
  </head>
  <body>
    <form action='./ex3-action.php' method='post'>
      Select your course<br>
      <input type='radio' name='radcourse' value='HNC'>HNC<br>
      <input type='radio' name='radcourse' value='HND'>HND<br>
      <input type='radio' name='radcourse' value='BSC'>BSC<br>
      <input type='radio' name='radcourse' value='PTBSC'>Part Time BSC</br>
      <input type='submit'>
    </form>
  </body>
</html>
```

Figure 3.12: ex3-action.php

```php
<?php
  switch ($_POST["radcourse"]) {
    case "HNC":
      echo "You have selected an HNC course";
      break;
    case "HND":
      echo "You have selected an HND course";
      break;
    case "BSC":
      echo "You have selected a BSC course";
      break;
    case "PTBSC":
      echo "You have selected Part Time BSC";
      break;
    default:
      echo "No course selected";
  }
?>
```

Figure 3.13: Input form for exercise 3

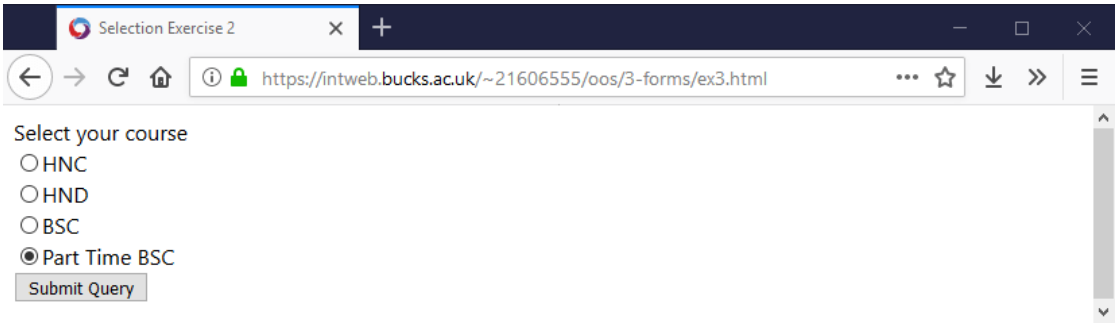


Figure 3.14: Output of exercise 3

3.4 Exercise 4

https://intweb.bucks.ac.uk/~21606555/oos/3-forms/ex4.php

Figure 3.15: ex4.php

```
<html>
  <head>
    <title>My Guestbook</title>
  </head>
  <body>
    <h1>Welcome to my Guestbook</h1>
    <h2>Please write me a little note below</h2>

    <form action='<?php echo htmlspecialchars($_SERVER["PHP_SELF"]);?>' method='post'
      <textarea cols='40' rows='5' name='note' wrap='virtual'></textarea>
      <input type='submit' value='Send it'>
    </form>

    <?php
      if (isset($_POST[note])) {
        // Substitute your login in place of "yourLogin"
        $fp = fopen('yourLogin.txt', 'a');
        fputs($fp, nl2br($_POST[note]) . '<br>');
        fclose($fp);
      }
    ?>
    <h2>The entries so far:</h2>

    <?php
      // Substitute your login in place of "yourLogin"
      @readfile('yourLogin.txt');
    ?>
```

Figure 3.16: Input form for exercise 4

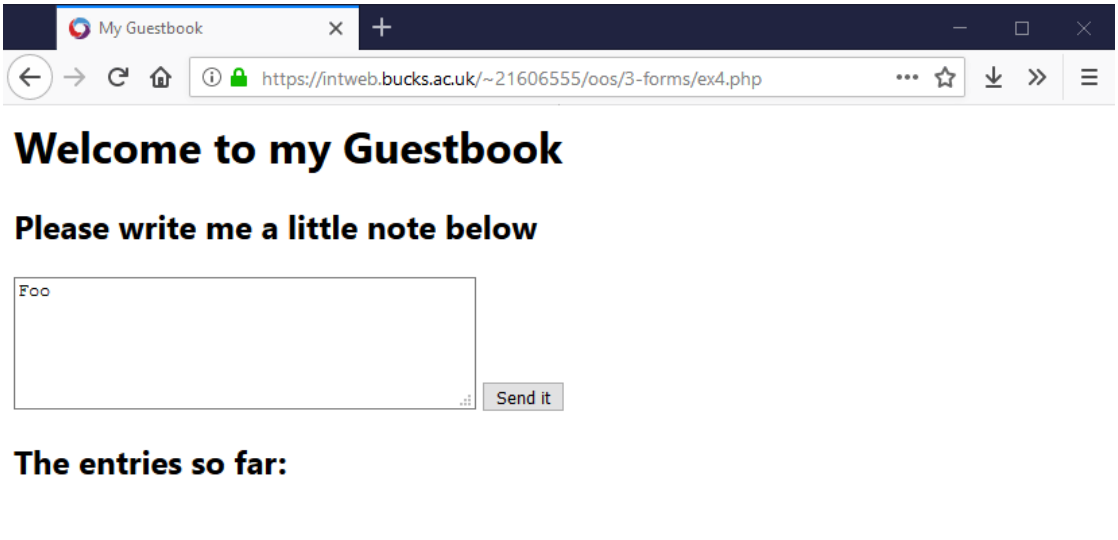
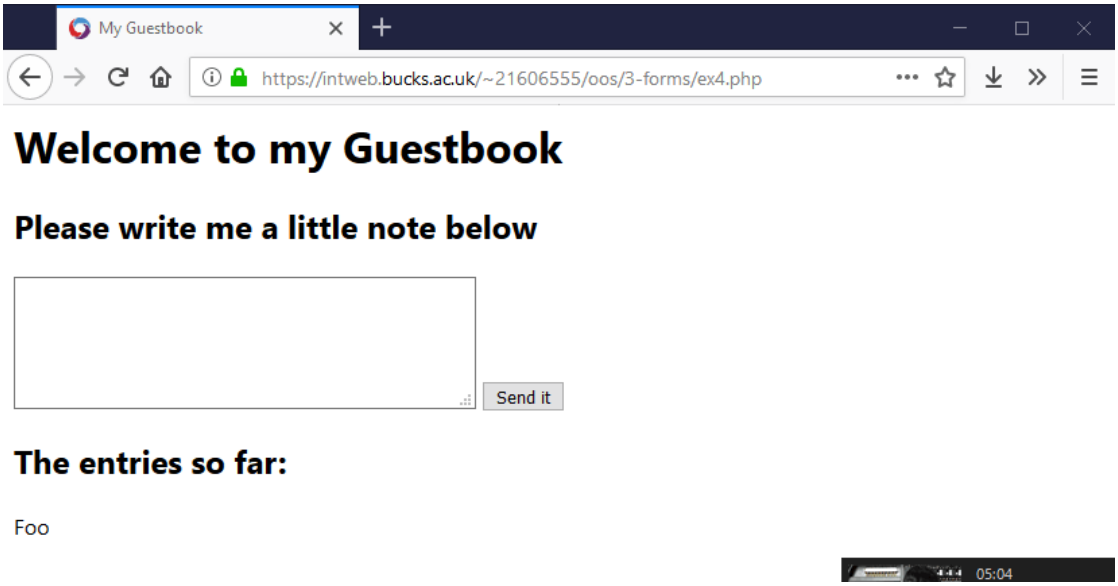


Figure 3.17: Output of exercise 4



Section 4

Sessions

4.1 Exercise 1

The code has been amended to properly display the widget order quantity on the confirmation page. This was the second most easiest method to implement as it only required starting a PHP Session in each of the scripts and using `$_SESSION` instead of `$_POST`.

Cookies are the hardest (but still straight forward) implement as a function is used to set the cookies using `setcookie(key, value)` and then using `$_COOKIE` instead of `$_POST` to access the value.

Hidden fields is the easiest method to implement as it only requires adding an extra field to the intermediary colour selection page that is hidden to the user.

https://intweb.bucks.ac.uk/~21606555/oos/4-sessions/ex1.html

Figure 4.1: ex1.html

```
<html>
  <head>
    <title>Shopping Page</title>
  </head>
  <body>
    <form action='ex1-colour.php' method='post'>
      Select the quantity of widgets you require
      <select name='selqty'>
        <option>1</option>
        <option>2</option>
        <option>3</option>
        <option>4</option>
        <option>5</option>
      </select>
      <br><br>
      <input type='submit' value='Buy'>
    </form>
  </body>
</html>
```

Figure 4.2: ex1-colour.php using session variables

```
<?php
  session_start();
  $_SESSION['widgetQuantity'] = $_POST[selqty];
?>

<html>
  <head>
    <title>Select colour page</title>
  </head>
  <body>
    <form action='ex1-confirm.php' method='post'>
      Select the colour for the <?= $_SESSION[widgetQuantity] ?>
      widgets you are ordering
      <select name='selcolour'>
        <option>White</option>
        <option>Red</option>
        <option>Yellow</option>
        <option>Green></option>
        <option>Blue</option>
      </select>
      <br><br>
      <input type='submit' value='Next'>
    </form>
  </body>
</html>
```

Figure 4.3: ex1-confirm.php using session variables

```
<?php
  session_start();
  $quantity = $_SESSION[widgetQuantity];
  $colour = $_POST[selcolour];
  echo "<h2>Your order quantity is $quantity</h2>";
  echo "<h2>and the selected colour is $colour.</h2>";
?>
```

Figure 4.4: Select quantity

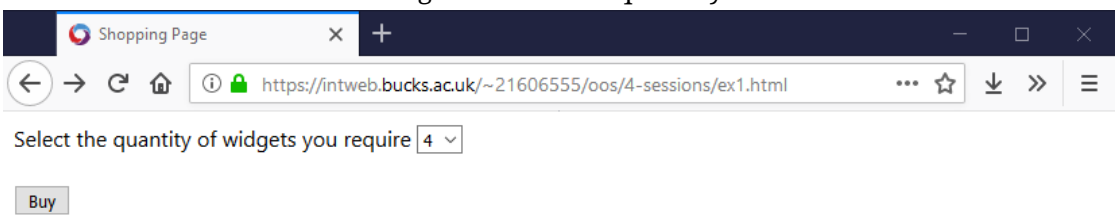


Figure 4.5: Select colour

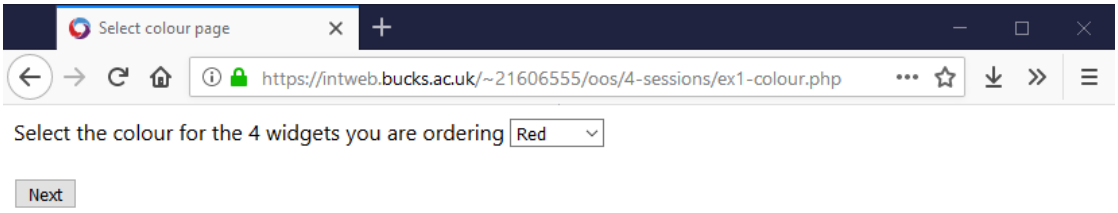
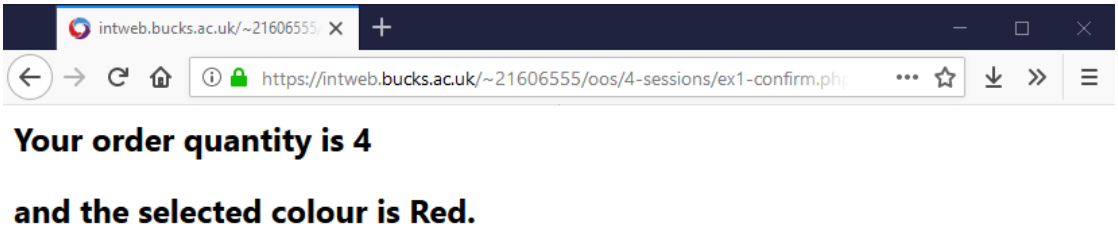


Figure 4.6: Result



4.2 Exercise 2

Price has been added to `ex1.html` and stored as a session variable so it can be used on the confirmation page.
https://intweb.bucks.ac.uk/~21606555/oos/4-sessions/ex2.html

Figure 4.7: ex2.html

```
<html>
  <head>
    <title>Shopping Page</title>
  </head>
  <body>
    <form action='ex2-colour.php' method='post'>
      Select the quantity of widgets you require
      <select name='selqty'>
        <option>1</option>
        <option>2</option>
        <option>3</option>
        <option>4</option>
        <option>5</option>
      </select>
      <br><br>
      <input type='text' name='txtprice' size='10' value='15.75' readonly>
      <input type='submit' value='Buy'>
    </form>
  </body>
</html>
```

Figure 4.8: ex2-colour.php setting price session variable

```
<?php
  session_start();
  $_SESSION['widgetQuantity'] = $_POST[selqty];
  $_SESSION['widgetPrice'] = $_POST[txtprice];
?>

<html>
  <head>
    <title>Select colour page</title>
  </head>
  <body>
    <form action='ex2-confirm.php' method='post'>
      Select the colour for the <?= $_SESSION[widgetQuantity] ?>
      widgets you are ordering
      <select name='selcolour'>
        <option>White</option>
        <option>Red</option>
        <option>Yellow</option>
        <option>Green></option>
        <option>Blue</option>
      </select>
      <br><br>
      <input type='submit' value='Next'>
    </form>
  </body>
</html>
```

Figure 4.9: ex2-confirm.php displaying price

```
<?php
  session_start();

  $colour = $_POST[selcolour];
  $quantity = $_SESSION[widgetQuantity];
  $totalPrice = $_SESSION[widgetPrice] * $quantity;

  echo "<h2>Your order quantity is $quantity</h2>";
  echo "<h2>and the selected colour is $colour.</h2>";
  echo "<h2>for a total price of £$totalPrice</h2>";
?>
```

Figure 4.10: Select quantity

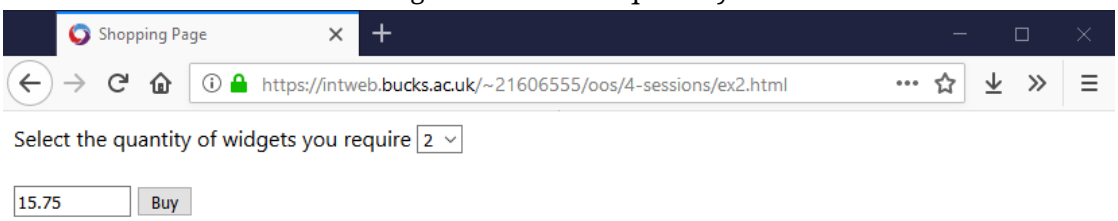


Figure 4.11: Select colour

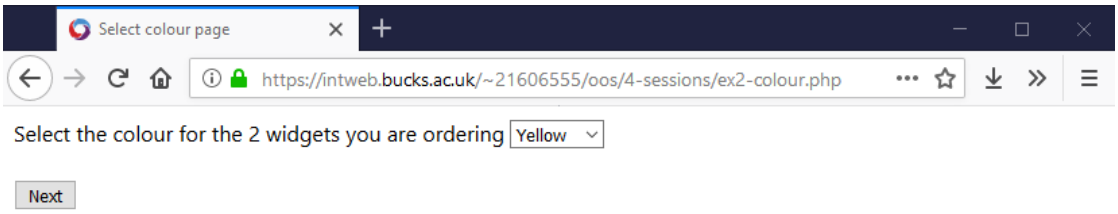


Figure 4.12: Result

4.3 Exercise 3

The price has been removed from `ex3.html` and added to a new page `ex3-size.php`. This includes a dropdown to select the size of widgets. I decided to include the price in the dropdown (`Small (£15.75)`) which is then split when the data is stored in session variables. This avoids having to use a switch statement and keeps the sizing and pricing data in the same place for easier modification.
https://intweb.bucks.ac.uk/~21606555/oos/4-sessions/ex3.html

Figure 4.13: ex3.html

```
<html>
  <head>
    <title>Shopping Page</title>
  </head>
  <body>
    <form action='ex3-size.php' method='post'>
      Select the quantity of widgets you require
      <select name='selqty'>
        <option>1</option>
        <option>2</option>
        <option>3</option>
        <option>4</option>
        <option>5</option>
      </select>
      <br><br>
      <input type='submit' value='Buy'>
    </form>
  </body>
</html>
```

Figure 4.14: ex3-size.php

```
<?php
  session_start();
  $_SESSION[widgetQuantity] = $_POST[selqty];
?>

<html>
  <head>
    <title>Select size page</title>
  </head>
  <body>
    <form action='ex3-colour.php' method='post'>
      Select the size of the <?= $_SESSION[widgetQuantity] ?>
      widgets you are ordering
      <select name='selsize'>
        <option>Small (£15.75)</option>
        <option>Medium (£16.75)</option>
        <option>Large (£17.75)</option>
        <option>Extra Large (£18.75)</option>
      </select>
      <br><br>
      <input type='submit' value='Next'>
    </form>
  </body>
</html>
```

Figure 4.15: ex3-colour.php

```php
<?php
  session_start();
  // Input is "Small (£15)"
  // Output is [ "Small", "15" ]
  $selSize = explode('(£', rtrim($_POST[selsize], ')'));
  $_SESSION[widgetSize] = $selSize[0];
  $_SESSION[widgetPrice] = $selSize[1];
?>

<html>
  <head>
    <title>Select colour page</title>
  </head>
  <body>
    <form action='ex3-confirm.php' method='post'>
      Select the colour for the <?= $_SESSION[widgetQuantity] ?>
      <?= $_SESSION[widgetSize] ?> widgets you are ordering
      <select name='selcolour'>
        <option>White</option>
        <option>Red</option>
        <option>Yellow</option>
        <option>Green></option>
        <option>Blue</option>
      </select>
      <br><br>
      <input type='submit' value='Next'>
    </form>
  </body>
</html>
```

Figure 4.16: ex3-confirm.php

```php
<?php
  session_start();

  $colour = $_POST[selcolour];
  $size = $_SESSION[widgetSize];
  $quantity = $_SESSION[widgetQuantity];
  $totalPrice = $_SESSION[widgetPrice] * $quantity;

  echo "<h2>Your order quantity is $quantity</h2>";
  echo "<h2>and the selected colour is $colour.</h2>";
  echo "<h2>and the selected size is $size</h2>";
  echo "<h2>for a total price of £$totalPrice</h2>";
?>
```

Figure 4.17: Select quantity

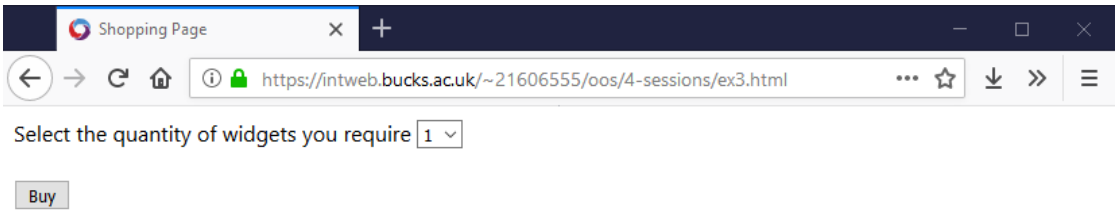


Figure 4.18: Select size

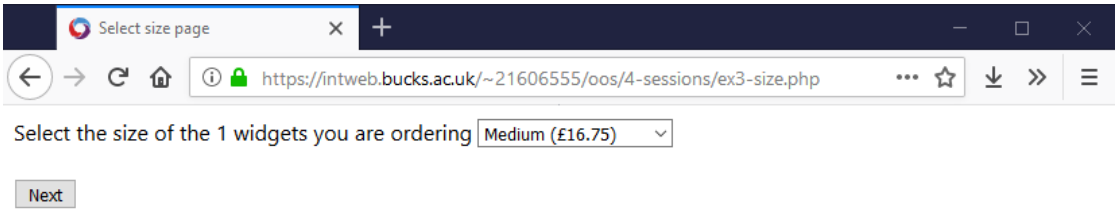


Figure 4.19: Select colour

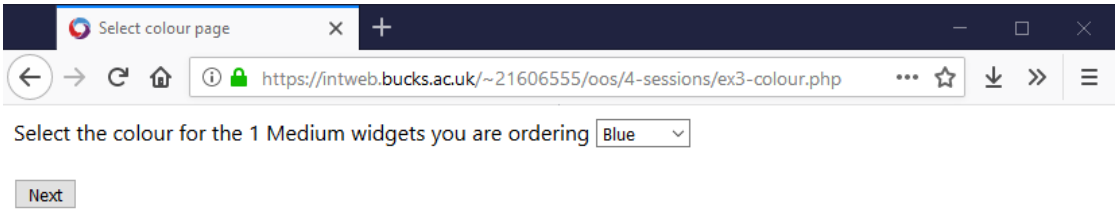


Figure 4.20: Result

4.4 Exercise 4

Figure 4.21: Order Form

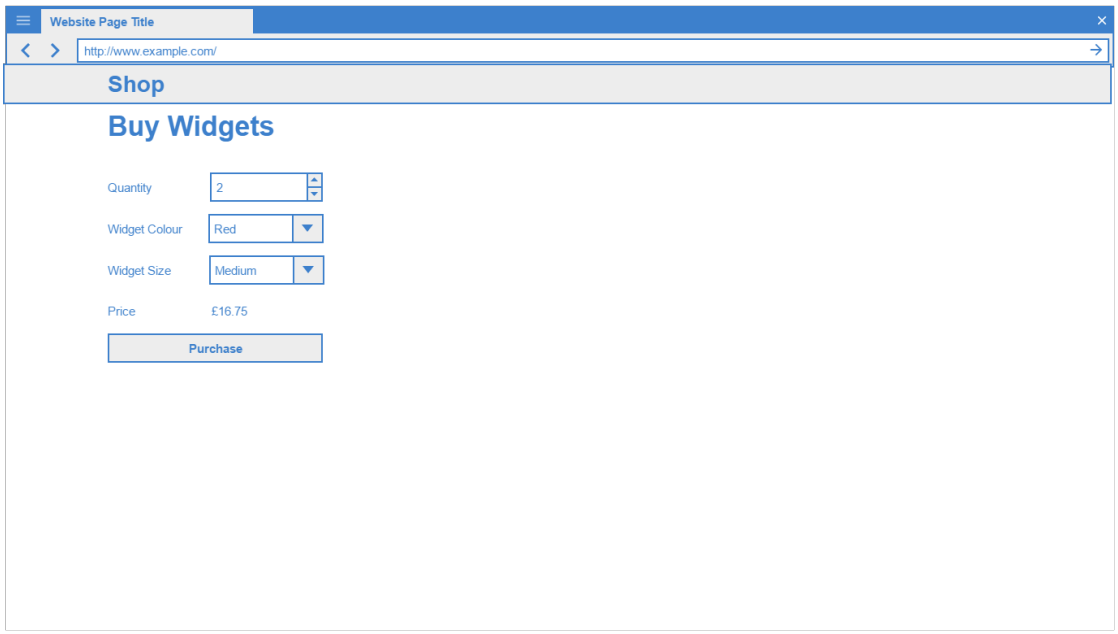
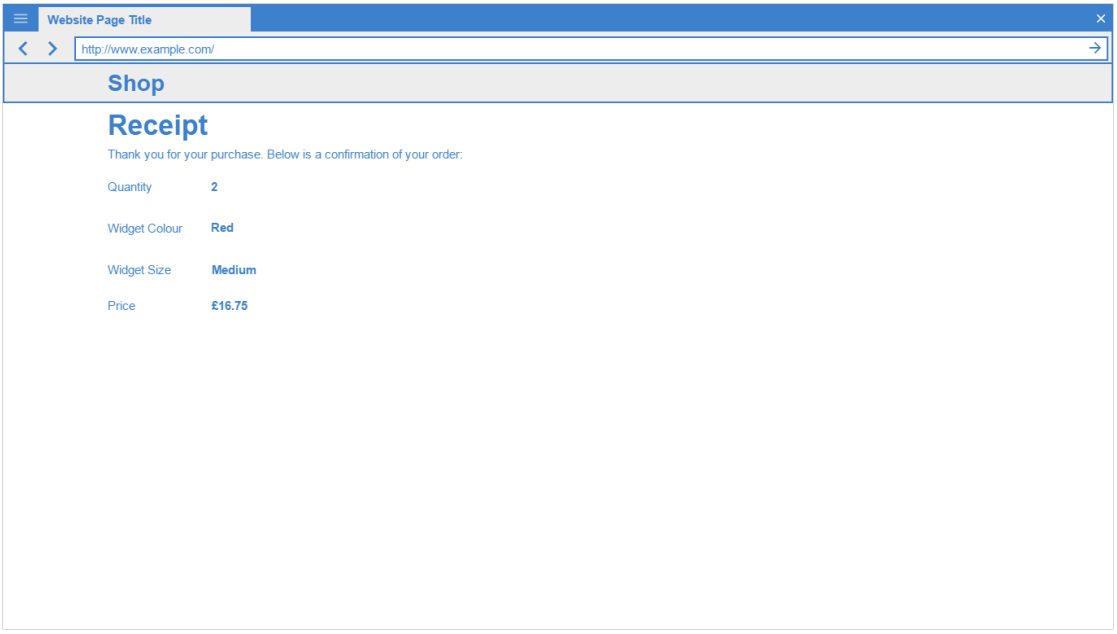


Figure 4.22: Order Confirmation



Section 5

Object Oriented PHP & MySQL

5.1 Exercise 1

https://intweb.bucks.ac.uk/~21606555/oos/5-oophp-mysql/index.php

Figure 5.1: class_lib.php

```php
<?php
  class person {
    var $name;

    function set_name($new_name) {
      $this->name = $new_name;
    }

    function get_name() {
      return $this->name;
    }
  }
?>
```

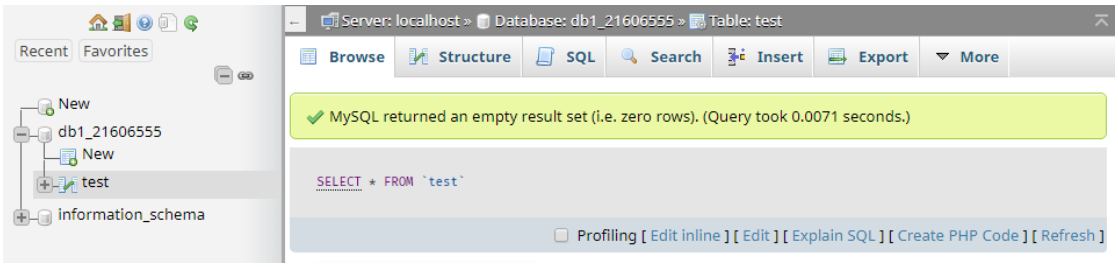
Figure 5.2: index.php

```php
<?php
  include('class_lib.php');

  $stefan = new person();
  $stefan->set_name('Stefan Mischook');
  echo 'Stefan\'s full name: ' . $stefan->get_name();

  $jimmy = new person;
  $jimmy->set_name('Nick Waddles');
  echo 'Nick\'s full name: ' . $jimmy->get_name();
  ?>
```

Figure 5.3: Result

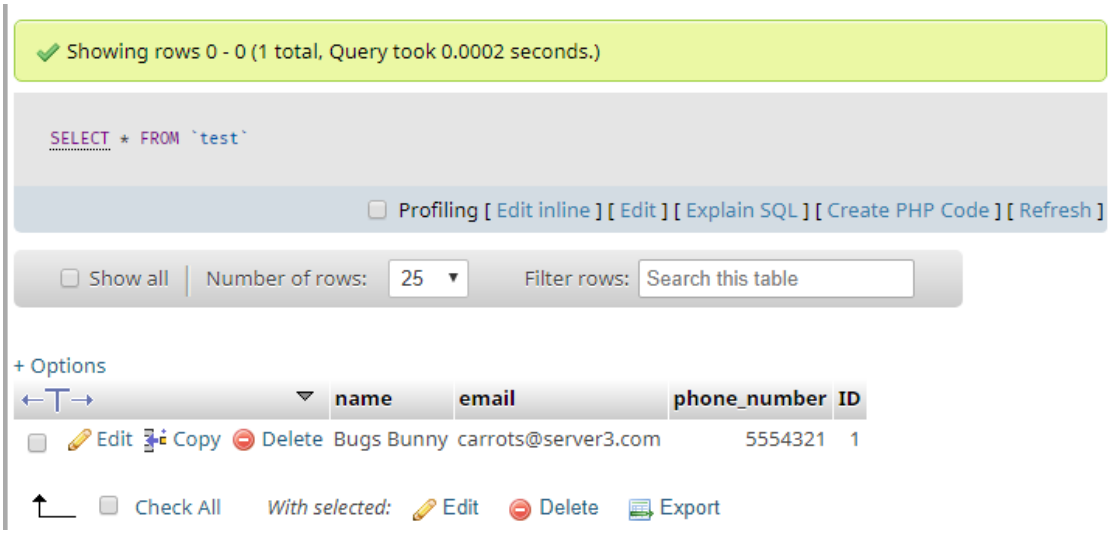
5.2 Exercise 2

Figure 5.4: Test table created



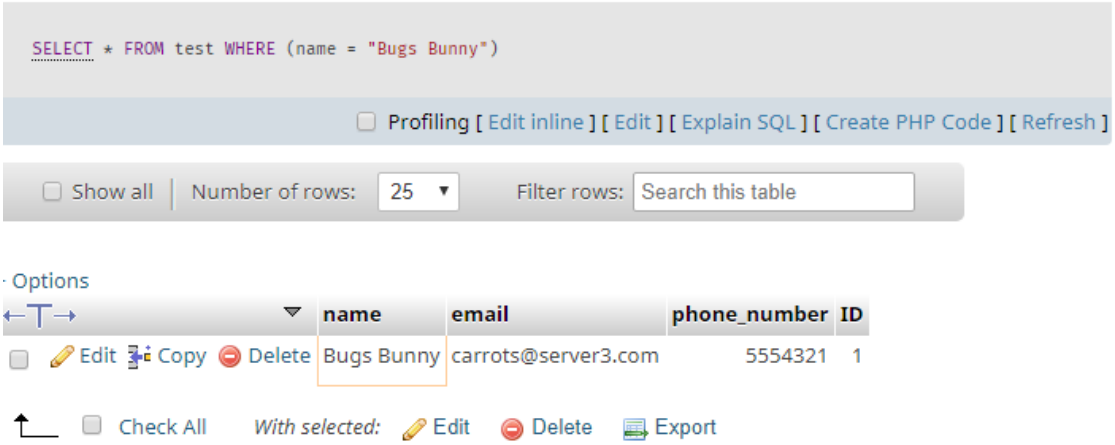
5.3 Exercise 3

Figure 5.5: Added row to table



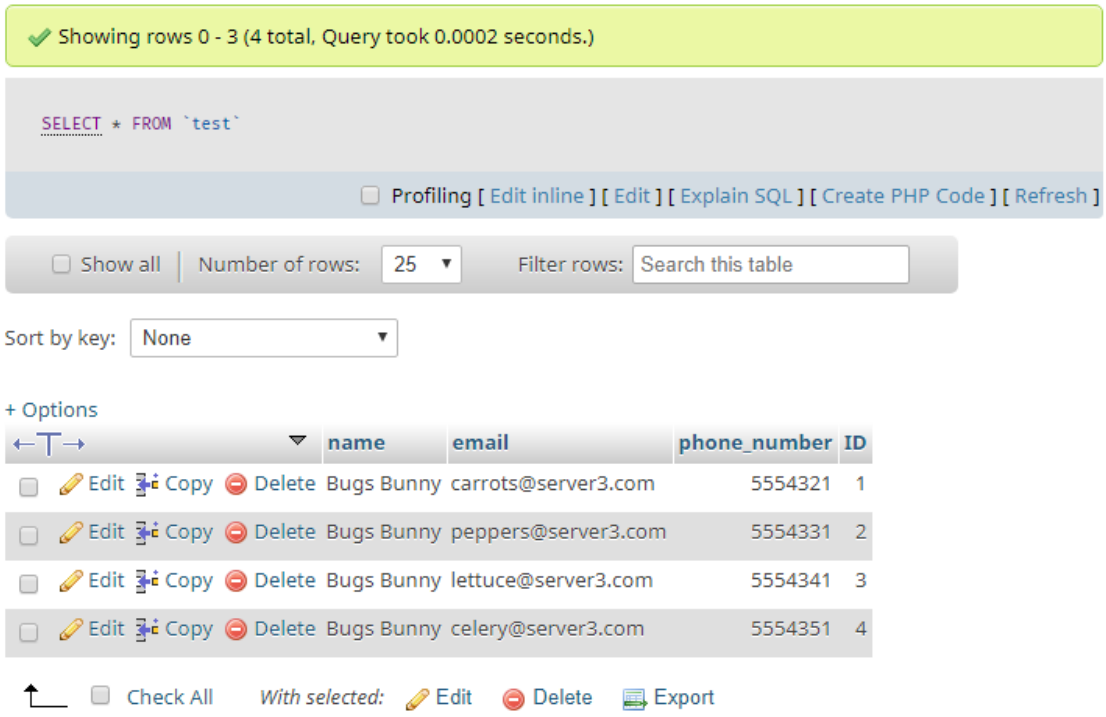
5.4 Exercise 4

Figure 5.6: Searching for Bugs Bunny



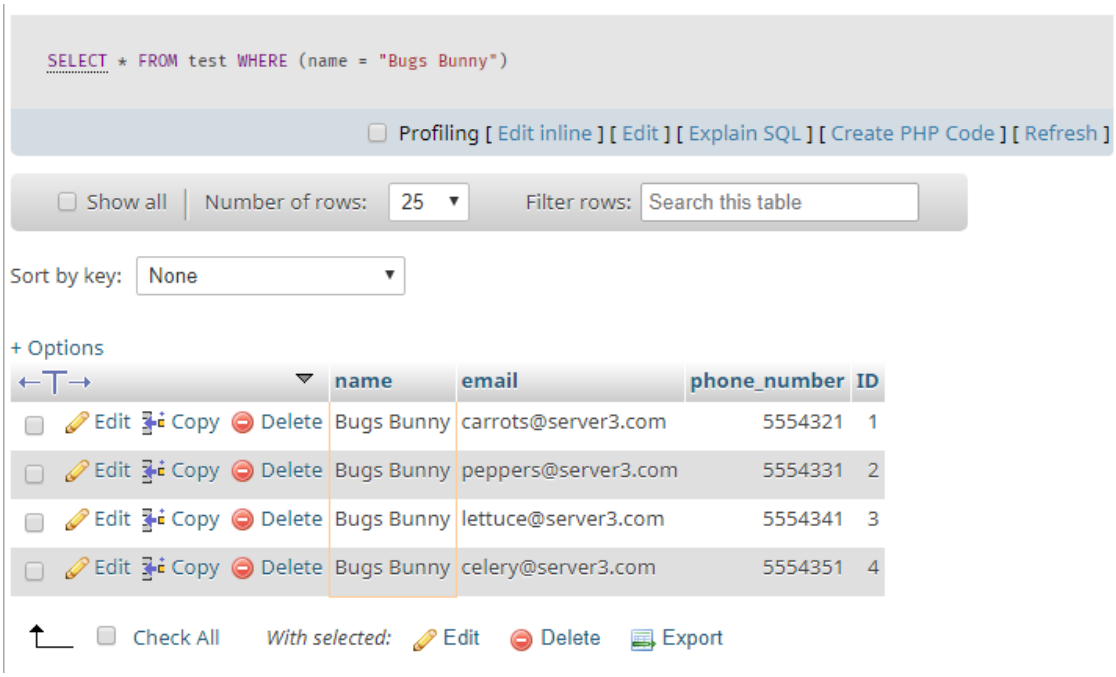
5.5 Exercise 5

Figure 5.7: Adding more rows to test table



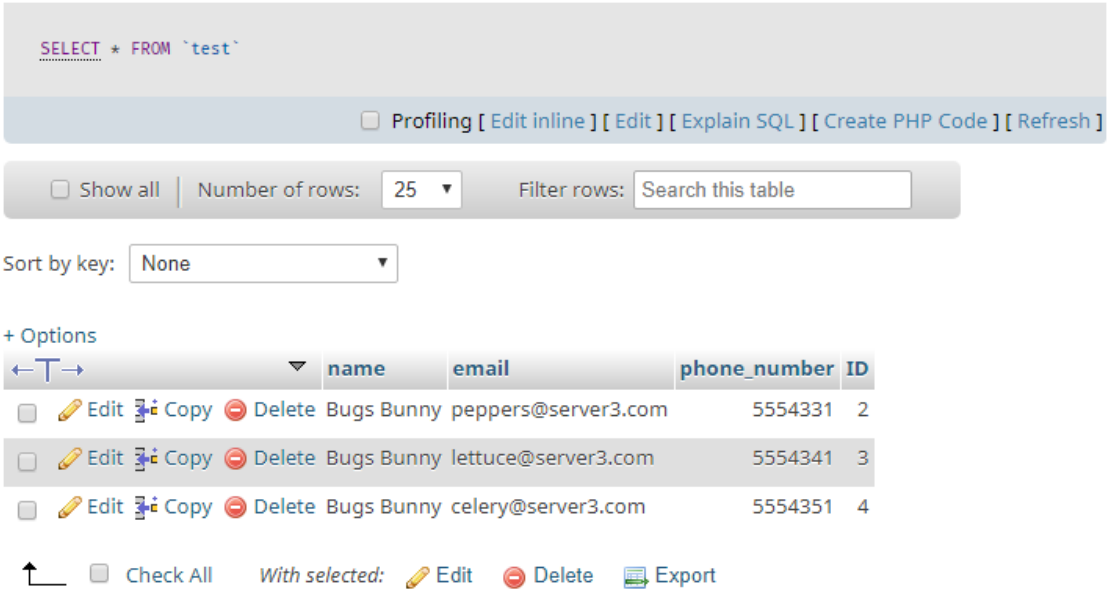
5.6 Exercise 6

Figure 5.8: Searching for all Bugs Bunny



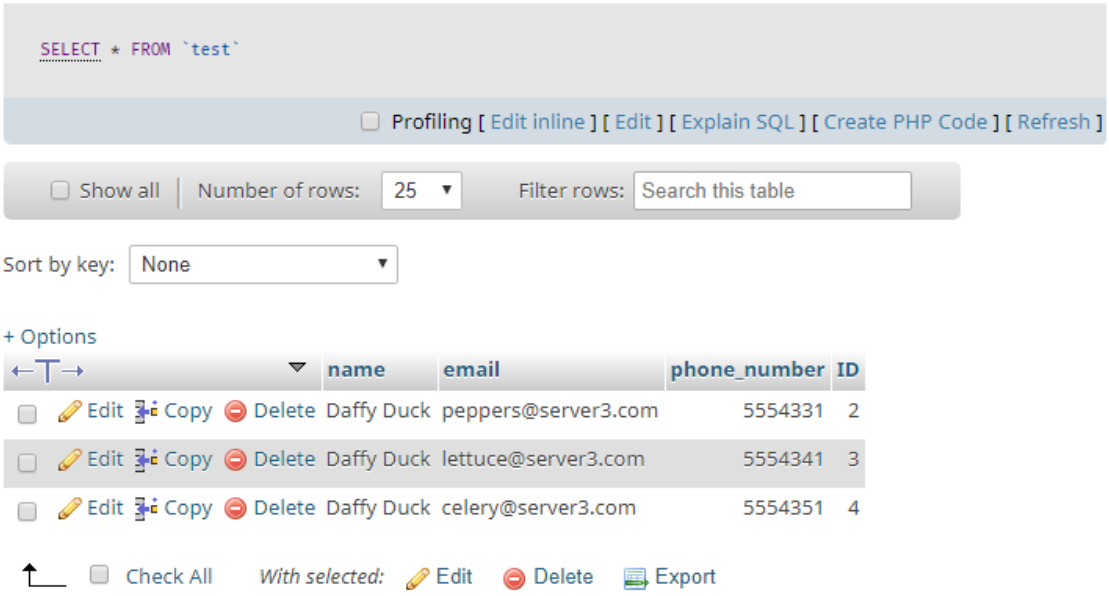
5.7 Exercise 7

Figure 5.9: Deleting a row



5.8 Exercise 8

Figure 5.10: Modifying cells in tables



Section 6

Database Connectivity & Functions

6.1 Exercise 1

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex1.html

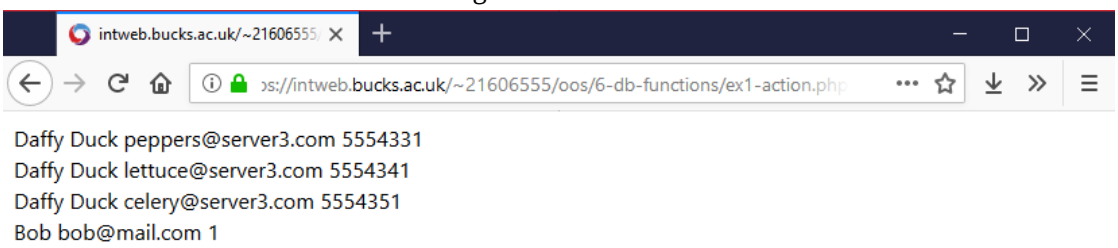
Figure 6.1: ex1.html

```
<html>
  <body>
    <form action='ex1-action.php' method='post'>
      Enter name:
      <input type='text' name='txtName'>
      <br>
      Enter email:
      <input type='text' name='txtEmail'>
      <br>
      Enter Tel number:
      <input type='text' name='txtPhoneNumber'>
      <br><br>
      <input type='submit' value='Save Data'>
    </form>
  </body>
</html>
```

Figure 6.2: ex1-action.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $txtName = $_POST[txtName];
  $txtEmail = $_POST[txtEmail];
  $txtPhoneNumber = $_POST[txtPhoneNumber];

  $query = "INSERT INTO test (name,email,phone_number) VALUES (?,?,?)";
  $sql = $db->prepare($query);
  $sql->bind_param('ssi', $txtName, $txtEmail, $txtPhoneNumber);
  $sql->execute();

  $result = $db->query("SELECT * from test");
  while ($row = $result->fetch_assoc()) {
    echo "$row[name] $row[email] $row[phone_number]<br>";
  }
?>
```

Figure 6.3: Result



6.2 Exercise 2

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex2.php

Figure 6.4: ex2.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $result = $db->query("SELECT * from test");
?>

<html>
  <body>
    <?php
      while ($row = $result->fetch_assoc()) {
        echo "<a href=\"ex2-action.php?id=$row[name]\">$row[name]</a> ";
        // See Exercise 4
        echo "<a href=\"ex4-action.php?id=$row[name]\">[DELETE]</a><br>";
      }
    ?>
  </body>
</html>
```

Figure 6.5: ex2-action.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $result = $db->query("SELECT * from test WHERE name='$_GET[id]'");
  $row = $result->fetch_assoc()
?>

<html>
  <body>
    <!-- See Exercise 3 for ex3-action.php -->
    <form action='ex3-action.php' method='post'>
      Name:
      <input type='text' name='txtname' value="<?= $row[name] ?>" readonly>
      <br>
      Phone Number:
      <input type='text' name='txttelno' value="<?= $row[phone_number] ?>">
      <br>
      Email:
      <input type='text' name='txtemail' value="<?= $row[email] ?>">
      <br>
      <input type='submit' value='Update Data'>
    </form>
  </body>
</html>
```

Figure 6.6: List of users

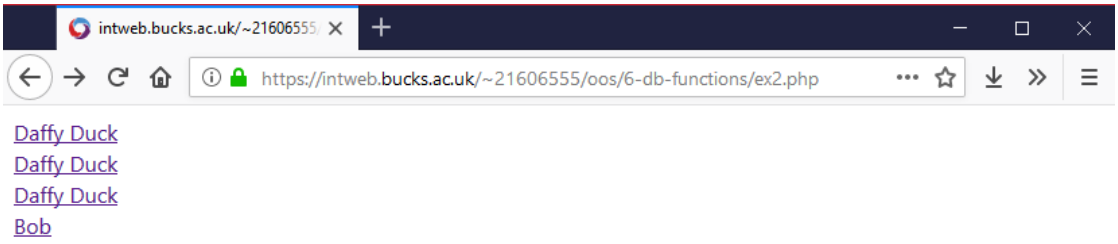
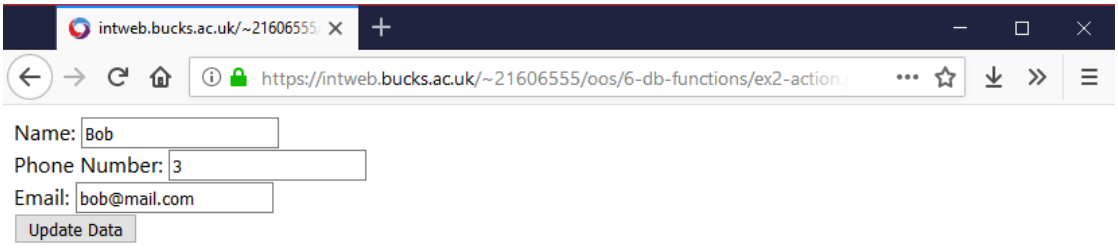


Figure 6.7: Editing a user



6.3 Exercise 3

Figure 6.8: ex3-action.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $txtName = $_POST[txtname];
  $txtEmail = $_POST[txtemail];
  $txtPhoneNumber = $_POST[txttelno];

  $query = "UPDATE test SET email=?,phone_number=? WHERE name=?";

  $sql = $db->prepare($query);
  $sql->bind_param('sis', $txtEmail, $txtPhoneNumber, $txtName);
  $sql->execute();

  printf("%d row updated.", $sql->affected_rows);
?>
```

Figure 6.9: Updated the user from exercise 2

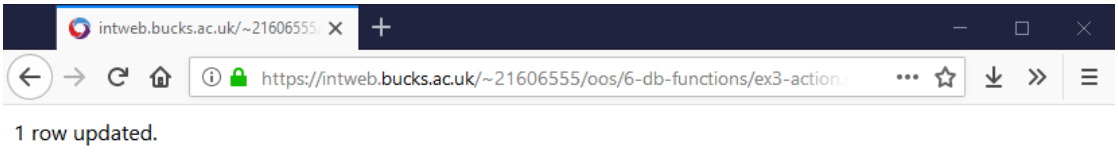


Figure 6.10: User data has been updated

6.4 Exercise 4

Figure 6.11: ex4-action.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "DELETE FROM test WHERE name=?";

  $sql = $db->prepare($query);
  $sql->bind_param('s', $_GET[id]);
  $sql->execute();

  printf("%d row deleted.", $sql->affected_rows);
?>
```

Figure 6.12: Ability to delete users

Figure 6.13: Deleting a user

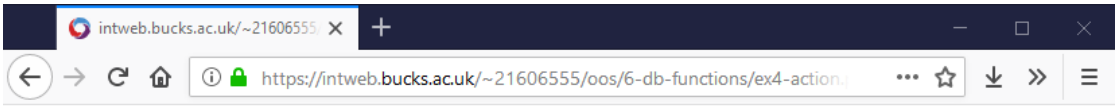
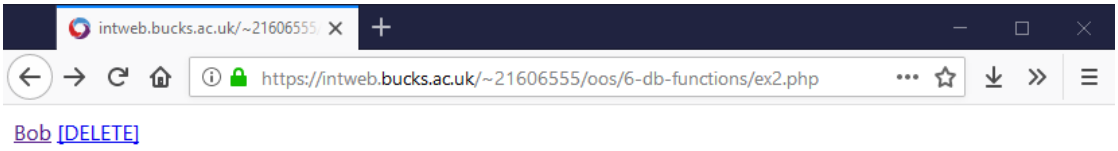


Figure 6.14: List without deleted users



6.5 Exercise 5

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex5.php

Figure 6.15: ex5-functions.php

```php
<?php
  function html_header($pageTitle) {
    echo "<html><head><title>$pageTitle</title></head><body>";
  }

  function html_h1($text) {
    echo "<h1>$text</h1>";
  }

  function html_h2($text) {
    echo "<h2>$text</h2>";
  }

  function html_footer() {
    echo "</body></html>";
  }
?>
```

Figure 6.16: ex5.php

```php
<?php
  include('ex5-functions.php');
  html_header('My first function demo');
  html_h1('These functions are going to');
  html_h2('save me lots of time');
  html_footer();
?>
```

Figure 6.17: Output

6.6 Exercise 6

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex6.php

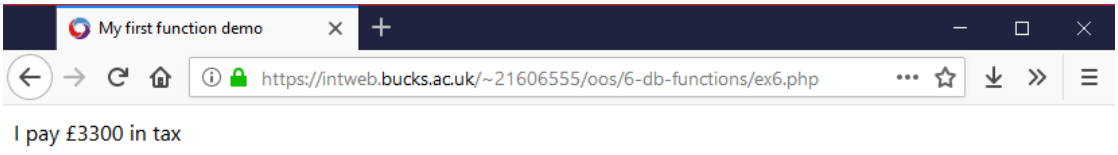
Figure 6.18: ex6-functions.php

```php
<?php
  function html_header($pageTitle) {
    echo "<html><head><title>$pageTitle</title></head><body>";
  }

  function calculate_tax($salary, $rate) {
    return $salary * ($rate / 100);
  }

  function html_footer() {
    echo "</body></html>";
  }
?>
```

Figure 6.19: ex6.php

```php
<?php
  include('ex6-functions.php');
  html_header('My first function demo');
  printf("I pay £%d in tax", calculate_tax(15000, 22));
  html_footer();
?>
```

Figure 6.20: Output



6.7 Exercise 7

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex7.php

Figure 6.21: ex7-functions.php

```php
<?php
  function html_header($pageTitle) {
    echo "<html><head><title>$pageTitle</title></head><body>";
  }

  function calculate_tax($salary, $rate=40) {
    return $salary * ($rate / 100);
  }

  function html_footer() {
    echo "</body></html>";
  }
?>
```

Figure 6.22: ex7.php

```php
<?php
  include('ex7-functions.php');
  html_header('My first function demo');
  printf("I pay £%d in tax", calculate_tax(15000));
  html_footer();
?>
```

Figure 6.23: Output

6.8 Exercise 8

https://intweb.bucks.ac.uk/~21606555/oos/6-db-functions/ex8.php

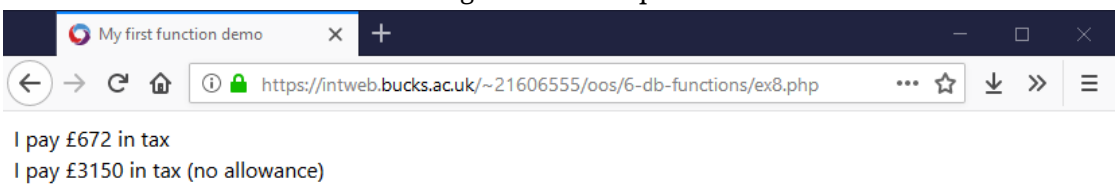
Figure 6.24: ex8-functions.php

```php
<?php
  function html_header($pageTitle) {
    echo "<html><head><title>$pageTitle</title></head><body>";
  }

  function calculate_tax($salary, $rate=40, $allowance=0) {
    return ($salary - $allowance) * ($rate / 100);
  }

  function html_footer() {
    echo "</body></html>";
  }
?>
```

Figure 6.25: ex8.php

```php
<?php
  include('ex8-functions.php');
  html_header('My first function demo');
  printf("I pay £%d in tax<br>", calculate_tax(15000, 21, 11800));
  printf("I pay £%d in tax (no allowance)", calculate_tax(15000, 21));
  html_footer();
?>
```

Figure 6.26: Output

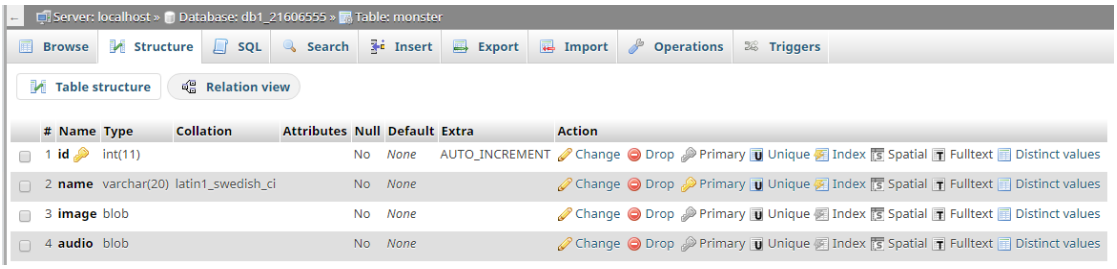


Section 7

PHP & Multimedia

7.1 Exercise 1

Figure 7.1: Table created



7.2 Exercise 2

https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex2.html

Figure 7.2: ex2.html

```
<html>
  <head>
    <link
      rel="stylesheet"
      href="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css"
      integrity="sha384-BVYiiSIFeK1dGmJRAkycuHAHRg32OmUcww7on3RYdg4Va+PmSTsz/K68vbdEj
      crossorigin="anonymous"
    >
    <script
      src="https://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"
      integrity="sha384-Tc5IQib027qvyjSMfHjOMaLkfuWVxZxUPnCJA7l2mCWNIpG9mGCD8wGNIcPD7
      crossorigin="anonymous"
    ></script>
  </head>
  <body>
    <h2>Monster Details</h2>

    <form enctype='multipart/form-data' action='ex2-savemonster.php' method='post'>
      Monster name:
      <input
        type='text'
        name='txtname'
        size='15'
        class='form-control'
      >
      <br><br>
      Monster image:
      <input
        type='file'
        name='monsterimage'
        accept='immage/jpeg'
        class='form-control'
      >
      <br><br>
      Monster sound:
      <input
        type='file'
        name='monsteraudio'
        accept='audio/basic'
```

```
          class='form-control'
        >
        <br><br>
        <input type='submit' class='btn btn-default' value='Save'>
      </form>
    </body>
```

Figure 7.3: ex2-savemonster.php

```
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $name = $_POST['txtname'];
  $image = $_FILES['monsterimage']['tmp_name'];
  $audio = $_FILES['monsteraudio']['tmp_name'];

  // Get binary file data
  $imagedata = addslashes(file_get_contents($image));
  $audiodata = addslashes(file_get_contents($audio));

  // bind_param accepts only variables
  $null = NULL;

  $query = "INSERT INTO monster (name,image,audio) VALUES (?,?,?)";
  $sql = $db->prepare($query);
  $sql->bind_param('sbb', $name, $null, $null);
  $sql->send_long_data(1, $imagedata);
  $sql->send_long_data(2, $audiodata);
  $sql->execute();
?>
```

Figure 7.4: Upload form

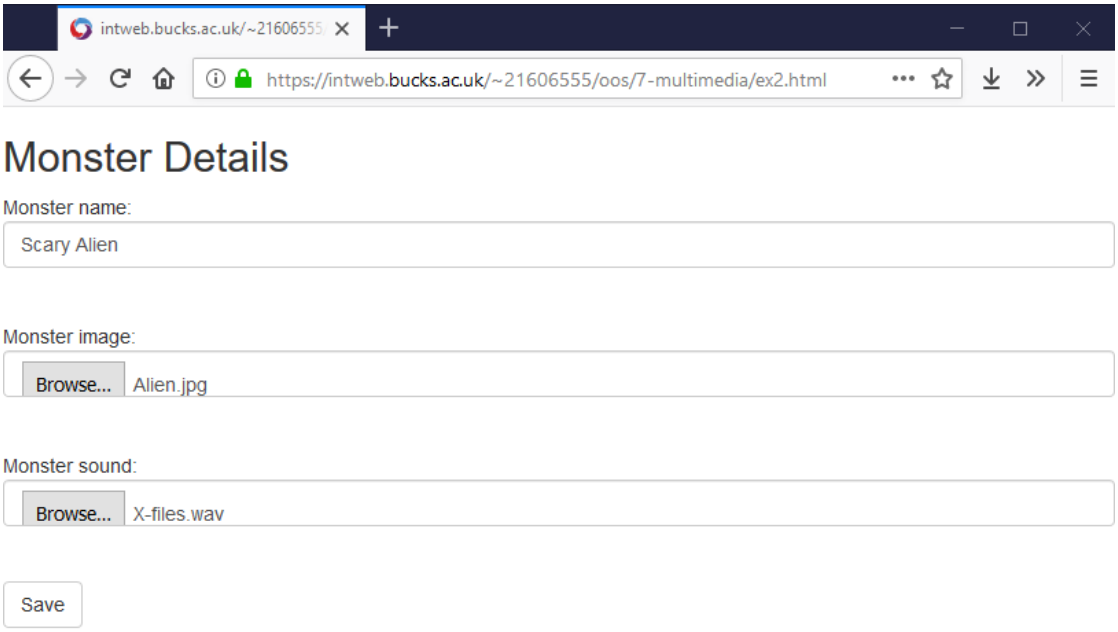


Figure 7.5: Uploaded data

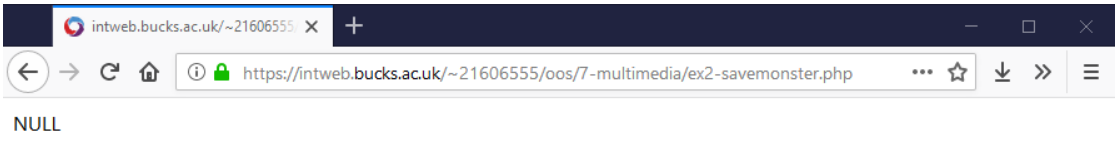


Figure 7.6: Data added to table

7.3 Exercise 3

https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex3-getjpg.php?id=1

https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex3-getwav.php?id=1

https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex3-displaymonster.php

Figure 7.7: ex3-getjpg.php

```php
<?php
  header('Content-Type: image/jpeg');

  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "SELECT image FROM monster WHERE id='$_GET[id]'";
  $result = $db->query($query);
  $row = $result->fetch_assoc();
  echo stripslashes($row['image']);
?>
```

Figure 7.8: ex3-getwav.php

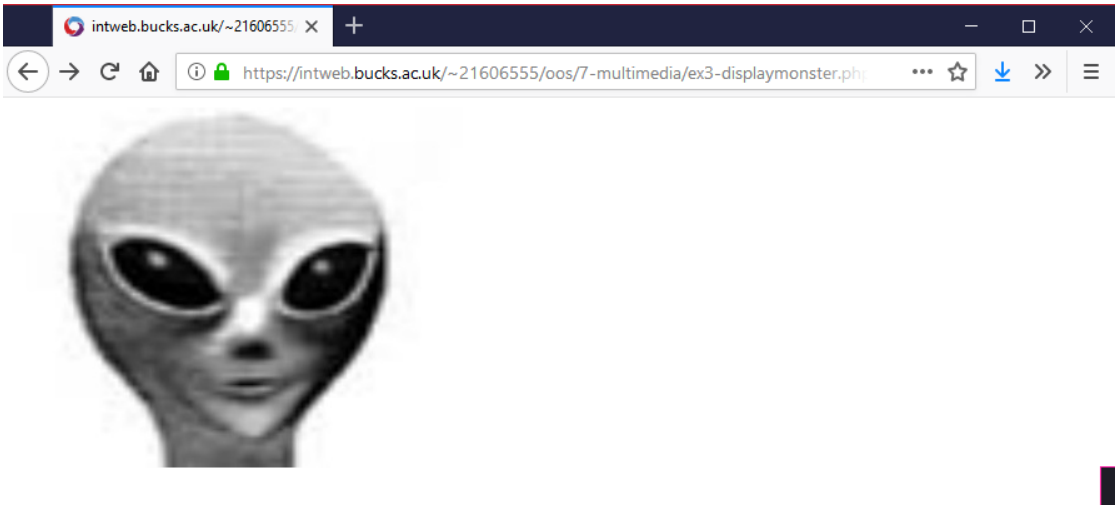
```php
<?php
  header('Content-Type: audio/wav');

  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "SELECT audio FROM monster WHERE id='$_GET[id]'";
  $result = $db->query($query);
  $row = $result->fetch_assoc();
  echo stripslashes($row['audio']);
?>
```

Figure 7.9: ex3-displaymonster.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "SELECT id FROM monster";
  $result = $db->query($query);
  $row = $result->fetch_assoc();

  echo "<img src='ex3-getjpg.php?id=$row[id]'>";
?>
```

Figure 7.10: Displaying image from database

Figure 7.11: Displaying monster from database



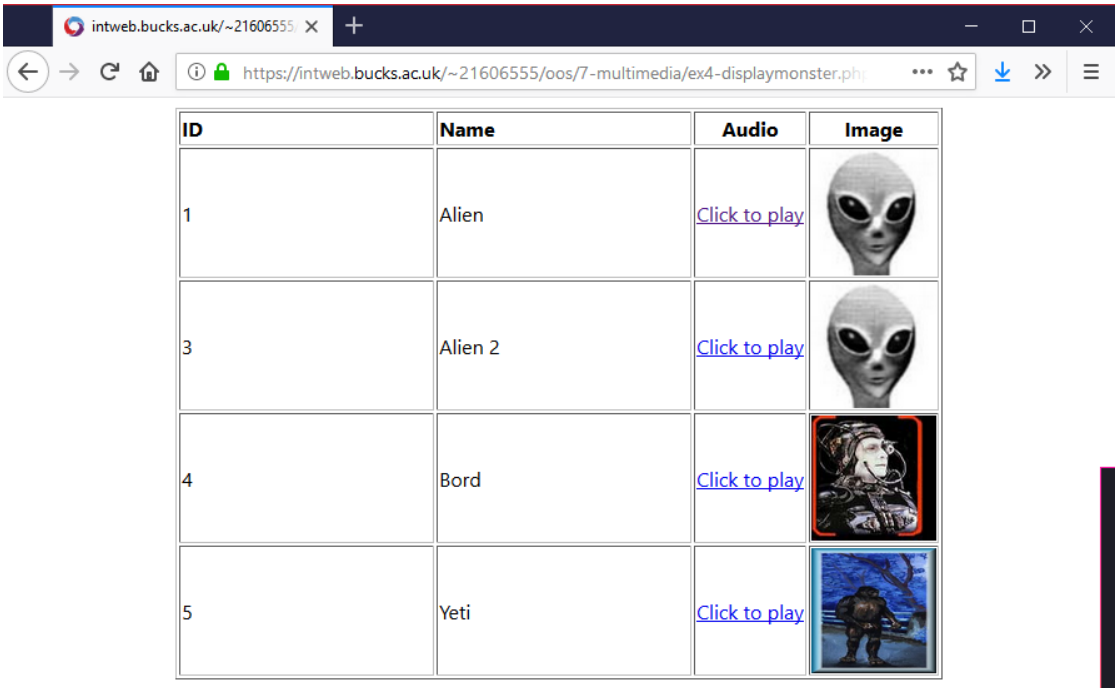
7.4 Exercise 4

https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex4-displaymonster.php

Figure 7.12: ex4-displaymonster.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "SELECT id,name FROM monster";
  $result = $db->query($query);

  echo "<table align='center' border='1'>";
  echo "<tr><th width='200' align='left'>ID</th>";
  echo "<th width='200' align='left'>Name</th>";
  echo "<th>Audio</th><th>Image</th></tr>";

  while($row = $result->fetch_assoc()) {
    echo "<tr>";
    echo "<td>$row[id]</td>";
    echo "<td>$row[name]</td>";
    echo "<td><a href='ex3-getwav.php?id=$row[id]'>Click to play</a></td>";
    echo "<td><img src='ex3-getjpg.php?id=$row[id]' height='100' width='100'></td>";
    echo "</tr>";
  }

  echo "</table>";
?>
```

Figure 7.13: Displaying all monsters in the database



7.5 Exercise 5

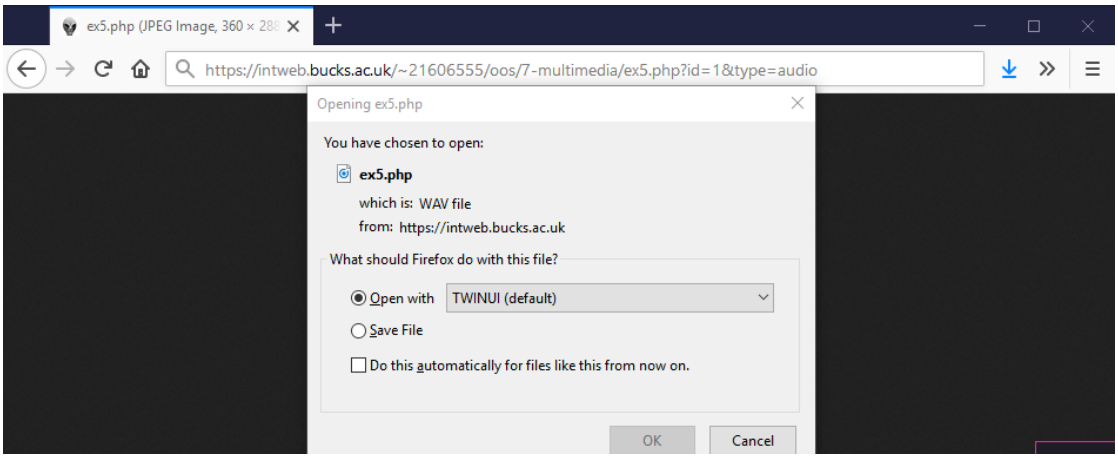
https://intweb.bucks.ac.uk/~21606555/oos/7-multimedia/ex5.php

Figure 7.14: ex5.php

```php
<?php
  switch ($_GET[type]) {
    case 'image':
      $contentType = 'image/jpeg';
      $column = 'image';
      break;
    case 'audio':
      $contentType = 'audio/wav';
      $column = 'audio';
      break;
  }

  header("Content-Type: $contentType");

  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "SELECT $column FROM monster WHERE id='$_GET[id]'";
  $result = $db->query($query);
  $row = $result->fetch_assoc();
  echo stripslashes($row[$column]);
?>
```

Figure 7.15: Combining the image and audio scripts (image)

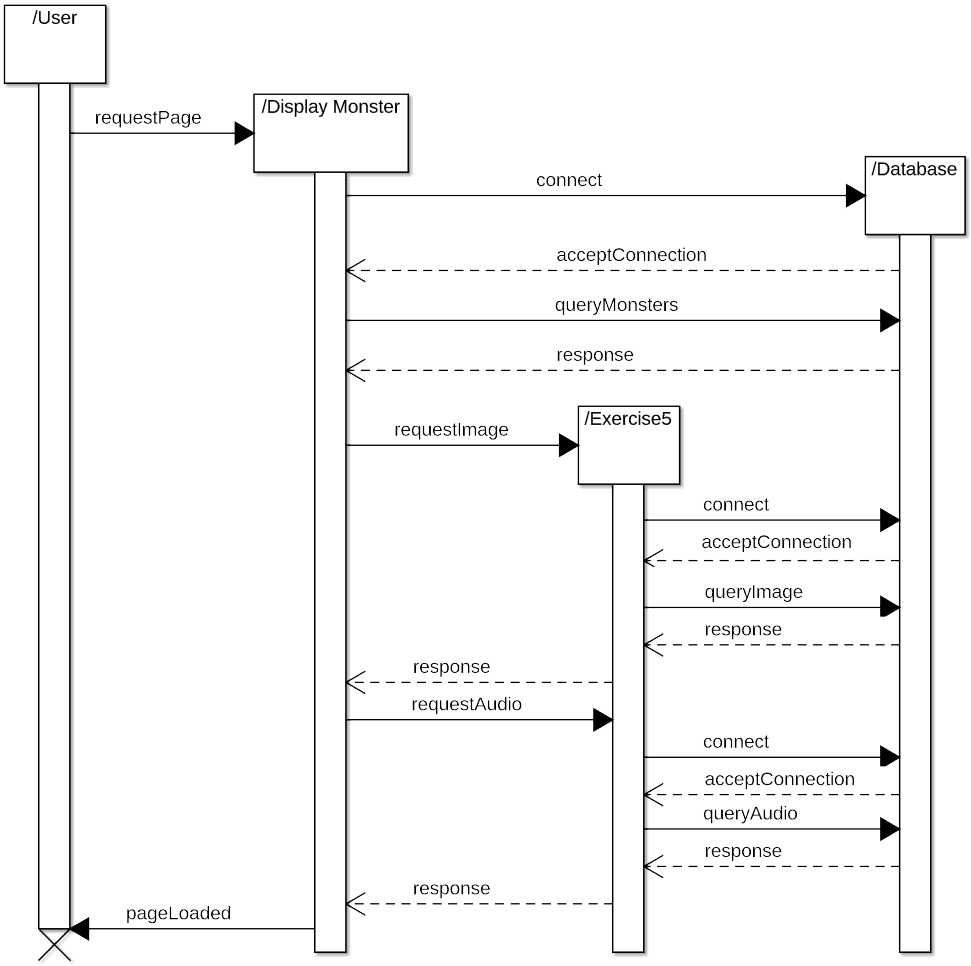
Figure 7.16: Combining the image and audio scripts (audio)



7.6 Exercise 6

Sequence diagram of the script created in Exercise 5 being used.

Figure 7.17: UML Sequence Diagram



Section 8

Database Connectivity & Testing

8.1 Exercise 1

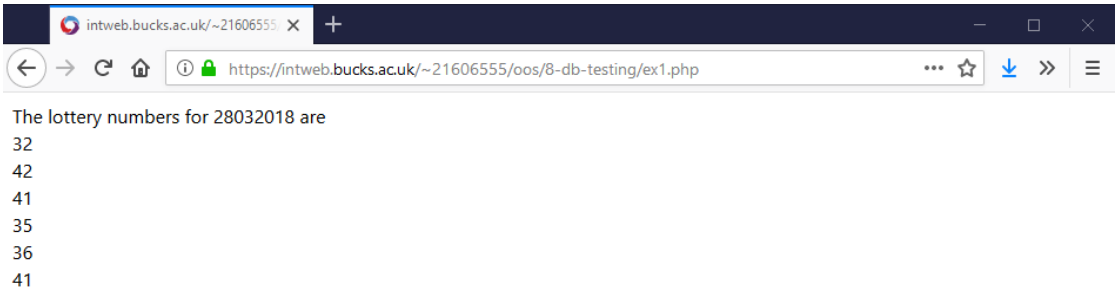
https://intweb.bucks.ac.uk/~21606555/oos/8-db-testing/ex1.php

Figure 8.1: ex1.php

```php
<?php
  $lottodate = date("dmY");

  echo "The lottery numbers for $lottodate are ";

  for ($n = 1; $n < 7; $n++) {
    $number[$n] = rand(1,49);
    echo "<br>$number[$n]";
  }
?>
```

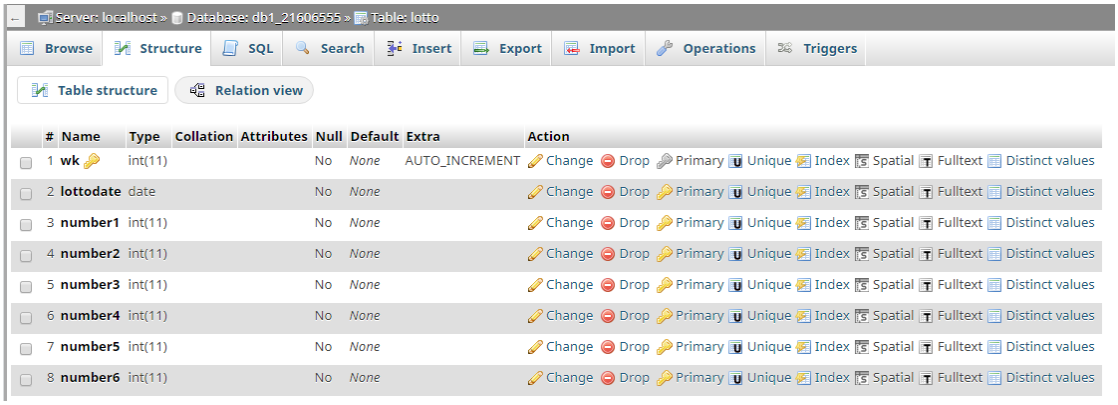
Figure 8.2: Lotto numbers output

Figure 8.3: Checking randomness



8.2 Exercise 2

Figure 8.4: Lotto table created



8.3 Exercise 3

https://intweb.bucks.ac.uk/~21606555/oos/8-db-testing/ex3.php

Figure 8.5: ex3.php

```php
<?php
  date_default_timezone_set('Europe/London');
  $lottoDate = date("Ymd");

  echo "The lottery numbers for $lottoDate are:";

  for ($n = 1; $n < 7; $n++) {
    $number[$n] = rand(1,49);
    echo "<br>$number[$n]";
  }

  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  $query = "INSERT INTO lotto (lottodate,number1,number2,number3";
  $query.= ",number4,number5,number6) VALUES (?,?,?,?,?,?,?)";
  $sql = $db->prepare($query);
  $sql->bind_param(
    'siiiiii',
    $lottoDate,
    $number[1],
    $number[2],
    $number[3],
    $number[4],
    $number[5],
    $number[6]
  );
  $sql->execute();

  echo "<br>This weeks numbers have been saved";
?>
```

Figure 8.6: Lotto numbers and saving to database

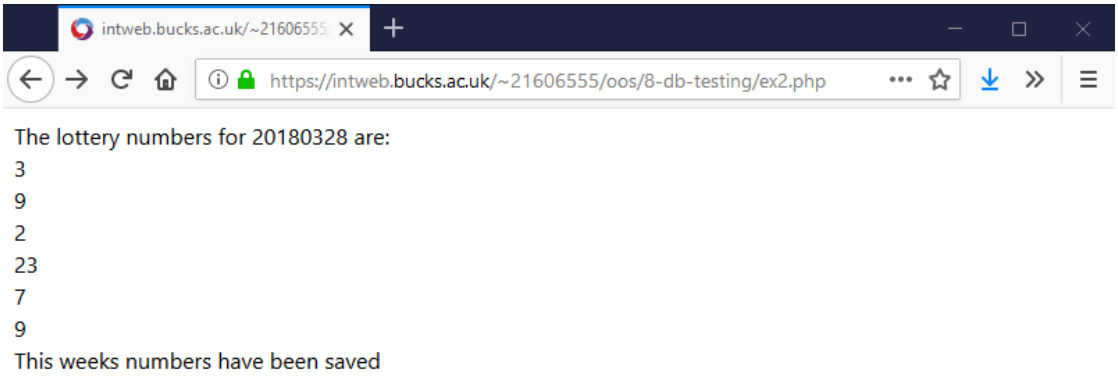


Figure 8.7: Data in database

	wk	lottodate	number1	number2	number3	number4	number5	number6
Edit Copy Delete	1	2018-03-28	43	23	15	5	31	35
Edit Copy Delete	2	2018-03-28	3	9	2	23	7	9

8.4 Exercise 4

https://intweb.bucks.ac.uk/~21606555/oos/8-db-testing/ex4.php

Figure 8.8: ex4.php

```php
<?php
  $hostname = "localhost";
  $username = "21606555";
  $password = "mysqluser";
  $database = "db1_21606555";
  $db = new mysqli($hostname, $username, $password, $database);

  if ($db->connect_errno) {
    printf("Connect failed: %s\n", $db->connect_error);
    exit();
  }

  if ($_POST[selweek]) {
    $query = "SELECT * FROM lotto WHERE wk=?";
    $sql = $db->prepare($query);
    $sql->bind_param('i', $_POST[selweek]);
    $sql->execute();
    $sql->bind_result(
      $wk,
      $lottodate,
      $number1,
      $number2,
      $number3,
      $number4,
      $number5,
      $number6
    );
    $sql->fetch();

    date_default_timezone_set('Europe/London');
    $date = date("jS F, Y", strtotime($lottodate));
    printf("Lotto numbers for %s (week %s)<br>", $date, $wk);

    for ($n = 1; $n < 7; $n++) {
      // Dynamically referencing number1 to 6 with ${...}
      printf("Number %s is %s<br>", $n, ${'number' . $n});
    }
  } else {
    $query = "SELECT wk FROM lotto";
    $sql = $db->prepare($query);
```

```
37      $sql->execute();
38      $sql->bind_result($wk);

39      echo "<form action='$_SESSION[PHP_SELF]' method='post'>";
40      echo "<br>Select the lottery week: ";
41      echo "<select name='selweek'>";

42      while ($sql->fetch()) {
43        echo "<option value='$wk'>$wk</option>";
44      }

45      echo "</select><br>";
46      echo "<input type='submit' value='Select'>";
47      echo "</form>";
48    }
49  ?>
```

Figure 8.9: Selecting from weeks in database

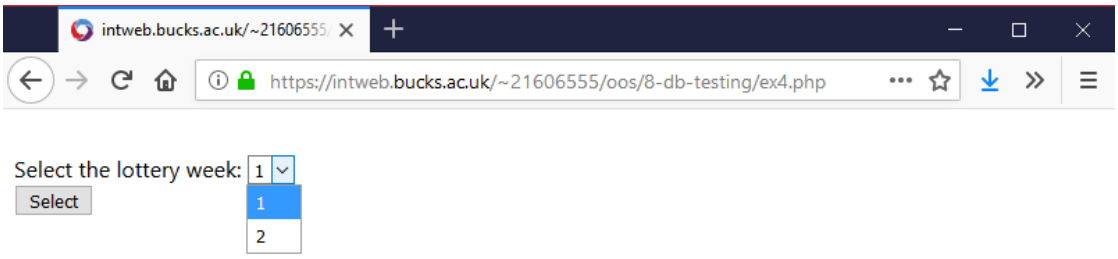
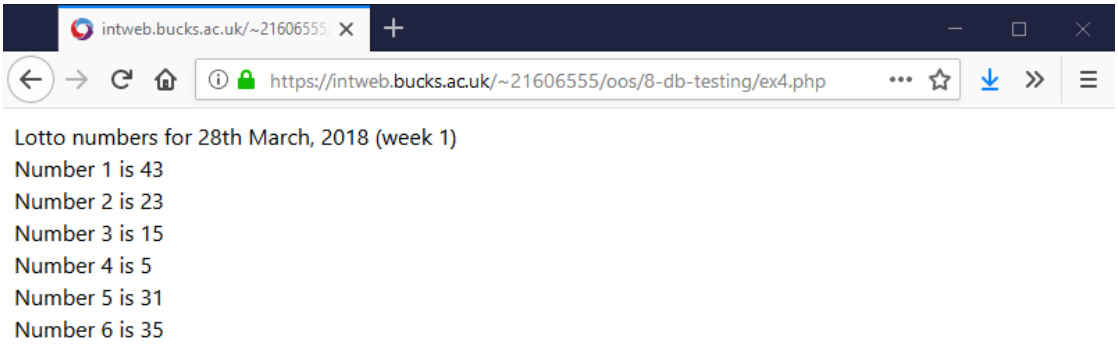


Figure 8.10: Displaying lotto numbers from the selected week



8.5 Exercise 5

Figure 8.11: Design for the lotto results page with navigation to prev/next week