CSCE 464/864: Internet Systems & Programming

Lecture 9: PHP

Dr. Qiang Liu, Spring 2022

IN OUR GRIT, OUR GLORY™

Overview

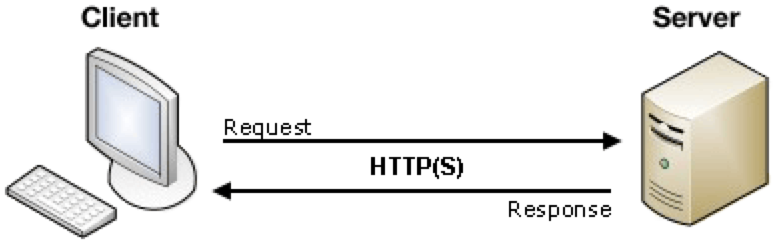
- ❖ GET & POST method
  - Data transmission

- ❖ Example – “Nature Resource”
  - Feedback
  - BmiForm

Overview

- Visit a webpage
  - The browser sends HTTP requests
  - Get data from server, e.g., HTML, CSS

- How to send data from client to server
  - Use HTTP, as well
  - HTTP methods: GET, POST, DEL, PUT, etc.



<Form>

- Form element
  - Define a form for collecting user inputs
  - Use <input> with various types, e.g., text, button, color
- Attribute: “action”
  - Define where the form is submitted to
  - Default GET method
- Attribute: “method”
  - GET, POST

```
<form action="/action_page.php" method="get">
  <label for="fname">First name:</label>
  <input type="text" id="fname" name="fname"><br><br>
  <label for="lname">Last name:</label>
  <input type="text" id="lname" name="lname"><br><br>
  <input type="submit" value="Submit">
</form>
```

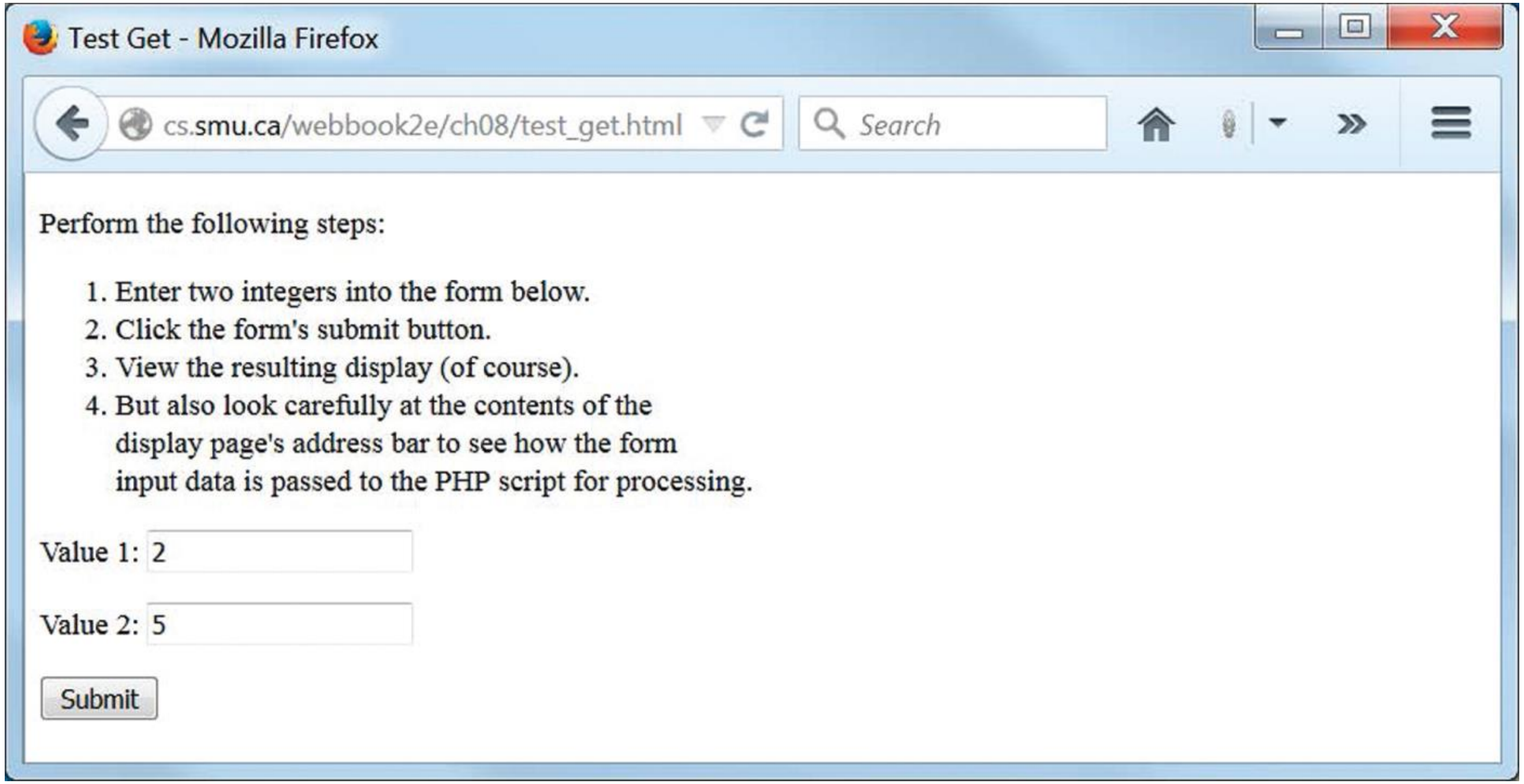
GET vs. POST

- ❖ GET
  - Embed/append the data into the web address
  - All data are visible to everyone
  - Limited size of the data
  - Bookmarked, cached, etc.
  - Retrieve nothing from server

- ❖ POST
  - Carry the data the payload of HTTP packets
  - Data encryption
  - Unlimited data size

Example - GET

- test_get.html
  - Submit form with values and show in server side



Test Get - Mozilla Firefox
cs.smu.ca/webbook2e/ch08/test_get.html
Search

Perform the following steps:

1. Enter two integers into the form below.
2. Click the form's submit button.
3. View the resulting display (of course).
4. But also look carefully at the contents of the display page's address bar to see how the form input data is passed to the PHP script for processing.

Value 1: 2

Value 2: 5

Example - test_get.html

```html
<!DOCTYPE html>
<!-- test_get.html -->
<html lang="en">
  <head>
    <meta charset="utf-8">
    <title>Test Get</title>
  </head>
  <body>
    <p>Perform the following steps:<p>
    <ol>
      <li>Enter two integers into the form below.</li>
      <li>Click the form's submit button.</li>
      <li>View the resulting display (of course).</li>
      <li>But also look carefully at the contents of the
        <br>display page's address bar to see how the form
        <br>input data is passed to the PHP script for processing.</li>
    </ol>
    <form method="get" action="test_get.php">
      <p>Value 1: <input type="text" name="value1"></p>
      <p>Value 2: <input type="text" name="value2"></p>
      <p><input type="submit" name="submit" value="Submit"></p>
    </form>
  </body>
</html>
```

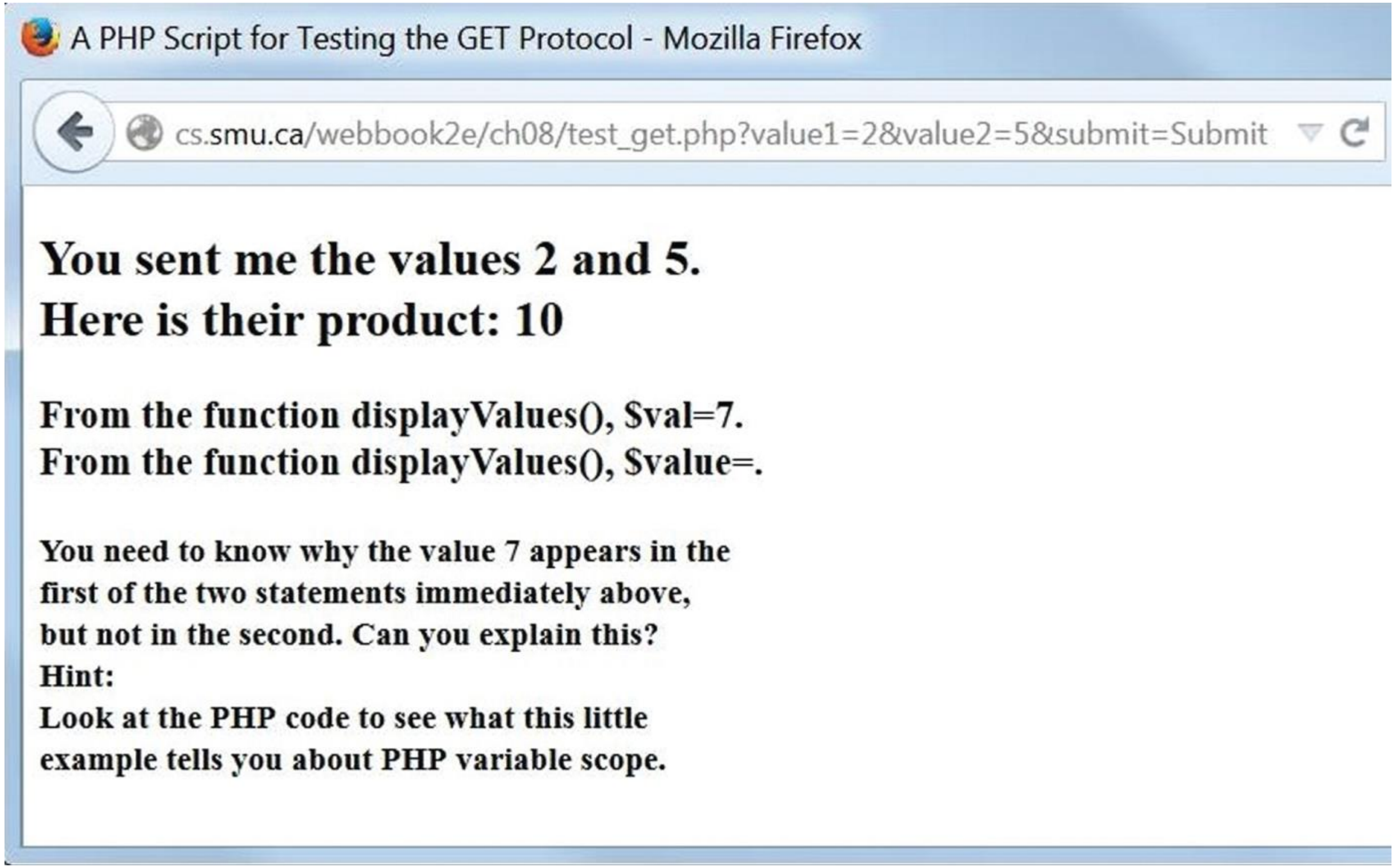
Example - test_get.php

```
<?php
echo "<h2>You sent me the values $_GET[value1] and $_GET[value2].
    <br>Here is their product: ";
echo $_GET['value1'] * $_GET['value2'];
echo "</h2>";
//…
$value = 7;
displayValues($value); //Note: function call (this line)
                       //can precede function definition:
function displayValues($val)
{
    echo "<h3>From the function displayValues(), \$val=$val.";
    echo "<br>From the function displayValues(), \$value=$value.</h3>";
    echo "<h4>You need to know why the value 7 appears in the
        <br>first of the two statements immediately above,<br>
        but not in the second. Can you explain this?<br>Hint:
        <br>Look at the PHP code to see what this little<br>
        example tells you about PHP variable scope.</h4>";
}
?>
```

Example - GET

- The server receive the following HTTP request
  - http://localhost/ch08/test_get.php?value1=2&value2=3&submit=Submit
  - The ? sign separates the URL and the following values
  - The & sign separates different values
  - The value submit is because there is an input in the client side
- _GET “superglobals” has these values
  - Automatically done by the server
  - You just need to query it if you need
- The function displayValues() is for showing the variable scope
  - The “value” is not declared as global, so not showing

Example - GET



A PHP Script for Testing the GET Protocol - Mozilla Firefox
cs.smu.ca/webbook2e/ch08/test_get.php?value1=2&value2=5&submit=Submit

You sent me the values 2 and 5.
Here is their product: 10

From the function displayValues(), $val=7.
From the function displayValues(), $value=.

You need to know why the value 7 appears in the first of the two statements immediately above, but not in the second. Can you explain this?
Hint:
Look at the PHP code to see what this little example tells you about PHP variable scope.

Example - POST

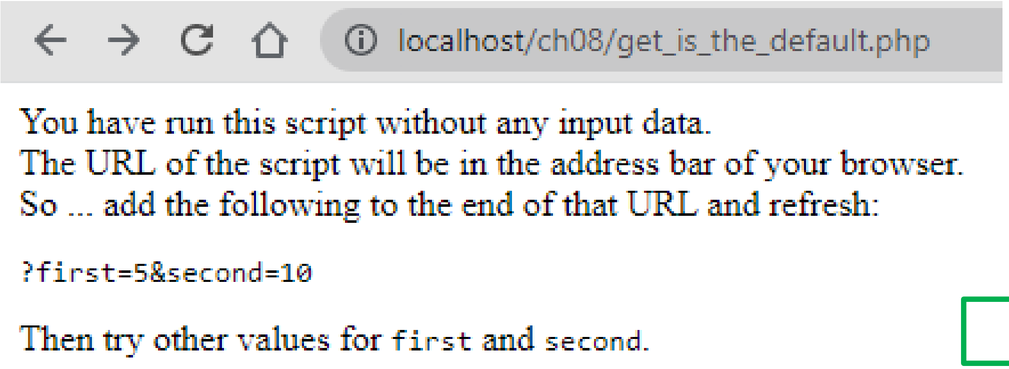
- In the <form> attributes, set method="get"
- The server receive the following HTTP request
  - http://localhost/ch08/test_post.php

- _POST “superglobals” has these values
  - Automatically done by the server
  - You just need to query it if you

```
<?php
echo "<h2>You sent me the values $_POST[value1] and $_POST[value2].
    <br>Here is their product: ";
echo $_POST['value1'] * $_POST['value2'];
echo "</h2>";
```

GET is the Default

- Every URL entered in browser starts a HTTP GET by default
  - http://localhost/ch08/get_is_the_default.php



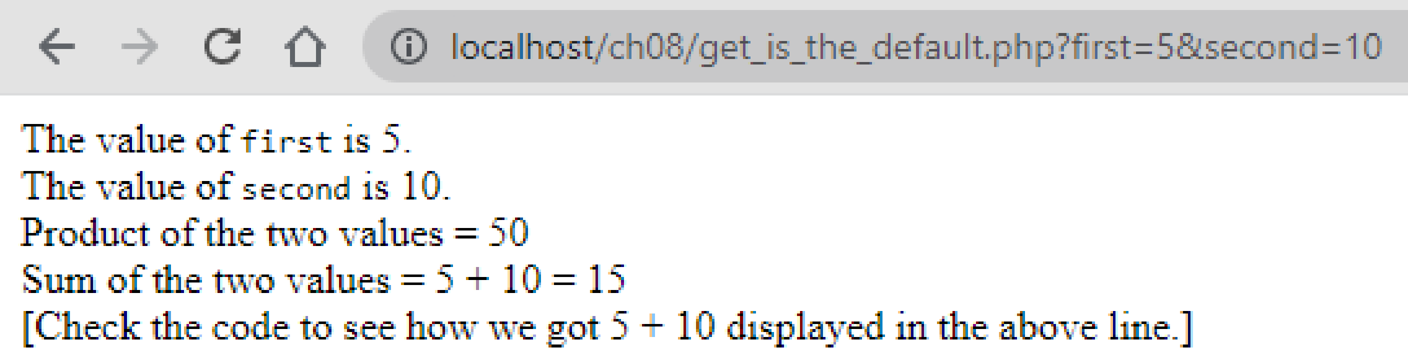
You have run this script without any input data.
The URL of the script will be in the address bar of your browser.
So ... add the following to the end of that URL and refresh:

```
?first=5&second=10
```

Then try other values for `first` and `second`.

No need for form element!

- So, one can directly customize URL to send data to server
  - http://localhost/ch08/get_is_the_default.php?first=5&second=10



The value of `first` is 5.
The value of `second` is 10.
Product of the two values = 50
Sum of the two values = 5 + 10 = 15
[Check the code to see how we got 5 + 10 displayed in the above line.]

Example - GET

```php
<?php
if (count($_GET) == 0)
{
    echo <<<INFO
You have run this script without any input data.<br>
The URL of the script will be in the address bar of your browser.<br>
So ... add the following to the end of that URL and refresh:<br>
<pre>?first=5&amp;second=10</pre>
Then try other values for <code>first</code> and <code>second</code>.
INFO;
    exit(0);
}
echo "The value of <code>first</code> is $_GET[first].<br>
    The value of <code>second</code> is $_GET[second].<br>";
$product = $_GET['first'] * $_GET['second'];
echo "Product of the two values = $product<br>
    Sum of the two values = {$_GET['first']} + {$_GET['second']} = ";
//Note the use of braces to achieve interpolation in the above line.
echo $_GET['first'] + $_GET['second'], "<br>";
echo "[Check the code to see how we got 5 + 10
       displayed in the above line.]";
?>
```

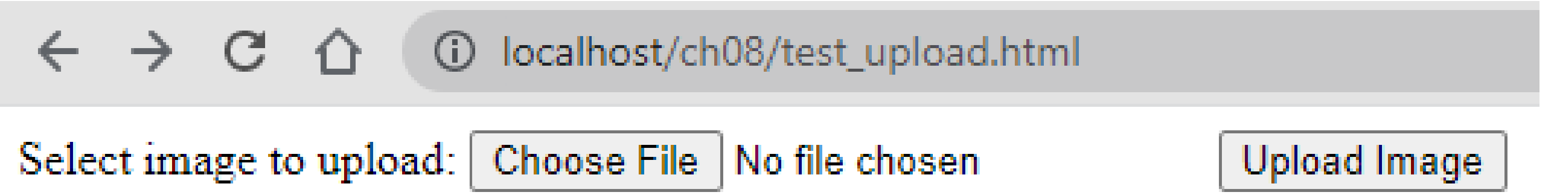
Exit and do not execute following codes

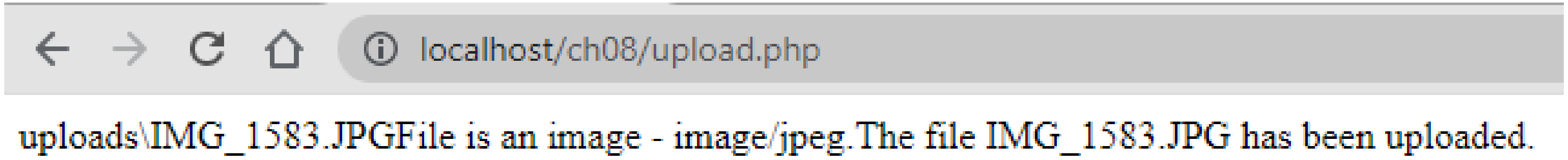
Guidance for GET & POST

- Use POST if you are transferring a lot of data.
- Use POST if you don’t want the user to see what you’re sending.
- Use POST if your data is going to change something at the destination (database content, for example).
- Use GET if you’re just getting some information from the destination (from a database on a server, for example)
- Use GET for short communications between scripts on the server.
- Definitely use GET if you’re sending small amounts of data, you don’t care who sees it, and you don’t want to bother setting up a form to receive and transmit the data.

Upload File to Server

- Use <form> the client to upload some files to server
  - Create <input type=“file”>
  - Create a submit button to trigger the upload





```
uploads\IMG_1583.JPGFile is an image - image/jpeg.The file IMG_1583.JPG has been uploaded.
```

https://www.w3schools.com/php/php_file_upload.asp

Upload File to Server

- Configure the "php.ini" Fil
  - Enable “`file_uploads = On`”
  - MAMP by default enabled

- Create The HTML Form
  - Allow client to select the file and click the upload button

- Create The Upload File PHP Script
  - Server-side process, e.g., copy to folder

Upload File to Server - HTML

- Simple HTML with <form> element
  - enctype="multipart/form-data“ → specifies which content-type

```
<!DOCTYPE html>
<html>
    <body>
        <form action="upload.php" method="post" enctype="multipart/form-data">
          Select image to upload:
          <input type="file" name="fileToUpload" id="fileToUpload">
          <input type="submit" value="Upload Image" name="submit">
        </form>
        </body>
</html>
```

Upload File to Server - PHP

- Use built-in function (`move_uploaded_file`)
  - Move the uploaded file from _FILES to folder

```php
<?php
$target_file = basename($_FILES["fileToUpload"]["name"]);

if (move_uploaded_file($_FILES["fileToUpload"]["tmp_name"], $target_file))
{
  echo "The file ". htmlspecialchars( basename(
$_FILES["fileToUpload"]["name"])). " has been uploaded.";
} else {
  echo "Sorry, there was an error uploading your file.";
}
```

_FILES

- The _FILES array contains following properties –
  - Here, 'file’ is the uploaded file
  - $_FILES['file']['name'] - The original name of the file to be uploaded.
  - $_FILES['file']['type'] - The mime type of the file.
  - $_FILES['file']['size'] - The size, in bytes, of the uploaded file.
  - $_FILES['file']['tmp_name'] - The temporary filename of the file in which the uploaded file was stored on the server.
  - $_FILES['file']['error'] - The error code associated with this file upload.

Upload File to Server - PHP

- Check if File Already Exists

```
if (file_exists($target_file)) {
  echo "Sorry, file already exists.";
  $uploadOk = 0;}
```

- Limit File Size

```
if ($_FILES["fileToUpload"]["size"] > 500000) {
  echo "Sorry, your file is too large.";
  $uploadOk = 0;}
```

- Limit File Type

```
// Allow certain file formats
if($imageFileType != "jpg" && $imageFileType != "png" && $imageFileType != "jpeg"
&& $imageFileType != "gif" ) {
  echo "Sorry, only JPG, JPEG, PNG & GIF files are allowed.";
  $uploadOk = 0;
}
```

Upload File to Server - PHP

- Check if it is actually an image
  - Use built-in function (`getimagesize`)

```
if(isset($_POST["submit"])) {
  $check = getimagesize($_FILES["fileToUpload"]["tmp_name"]);
  if($check !== false) {
    echo "File is an image - " . $check["mime"] . ".";
    $uploadOk = 1;
  } else {
    echo "File is not an image.";
    $uploadOk = 0;
  }
}
```

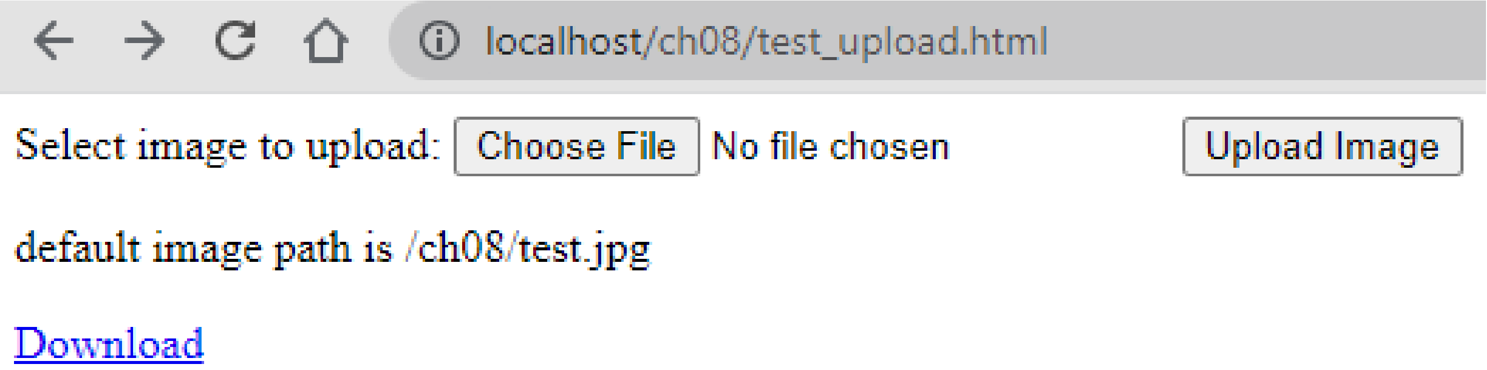
extension

Download File to Server - HTML

- Use <href> element with “download” attribute

```
<p>default image path is /ch08/test.jpg</p>

<a href="/ch08/test.jpg" download>Download</a>
```



Select image to upload: Choose File No file chosen Upload Image

default image path is /ch08/test.jpg

Download

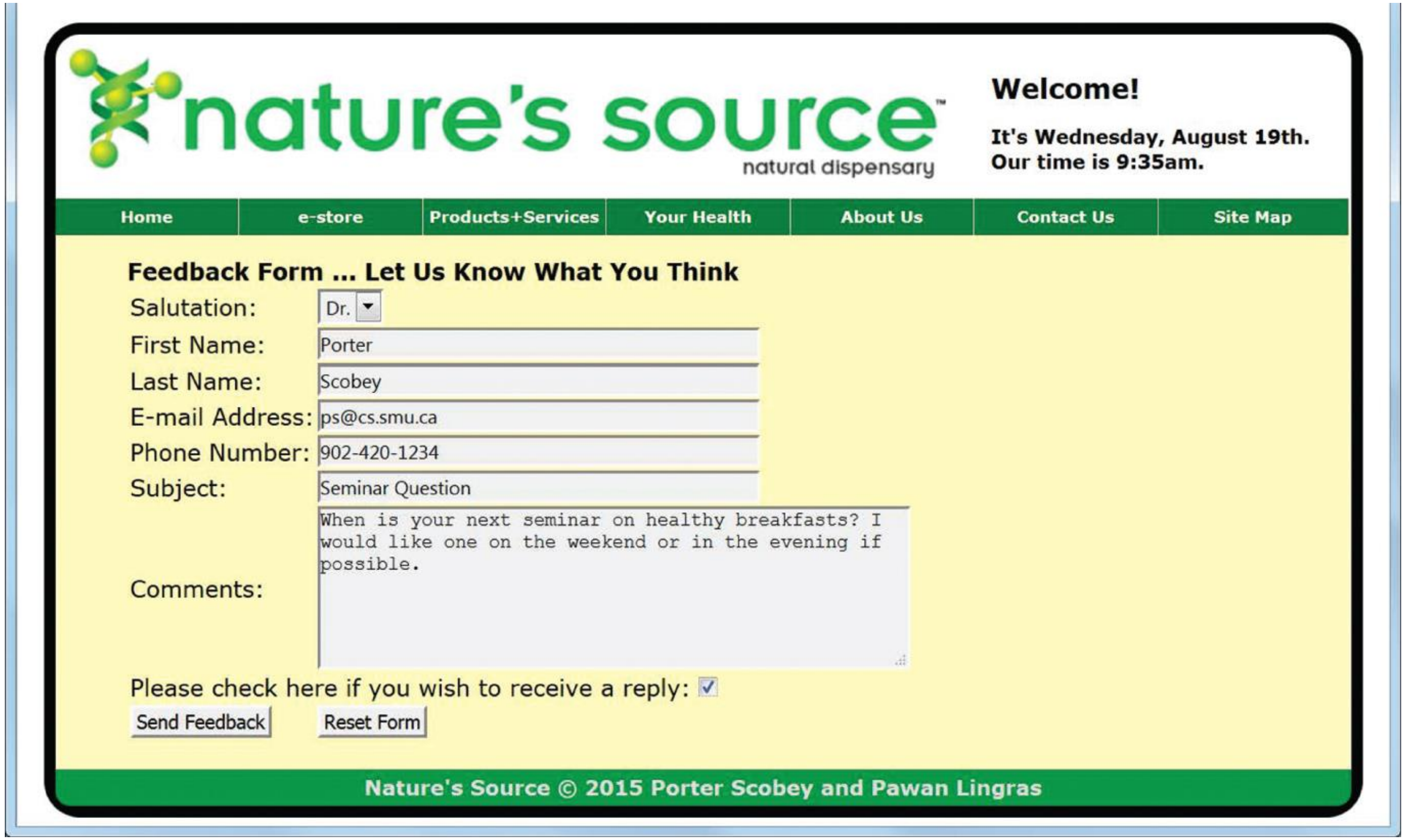
Practice

❖ Create a homework submission webpage

- Allow client to send data (name) to the server, with GET or POST
- Allow client to send file (**PDF**) to the server, with <form>
- Server process the data, e.g., PDF size, etc.
- Allow client to download the file it just uploaded

Nature Source - Feedback

- Full version of feedback
  - Server-side processing, email notification, logging



Welcome!

It's Wednesday, August 19th.
Our time is 9:35am.

Feedback Form ... Let Us Know What You Think

Salutation: Dr. ▾

First Name: Porter

Last Name: Scobey

E-mail Address: ps@cs.smu.ca

Phone Number: 902-420-1234

Subject: Seminar Question

Comments:

```
When is your next seminar on healthy breakfasts? I
would like one on the weekend or in the evening if
possible.
```

Please check here if you wish to receive a reply: ☑

Send Feedback   Reset Form

Nature's Source © 2015 Porter Scobey and Pawan Lingras

Nature Source – index.php

- Server side include
  - Use “include” function in PHP if you need it (raise warnings)
  - Use “require” function in PHP if you require it (raise errors)

```
  <?php include("common/document_head.html");
?>
  <!-- index.php for ch08/nature -->
  <body onload="startRotation()">
    <header>
      <?php
      include("common/banner.php");
      include("common/menus.html");
      ?>
    </header>
```

Nature Source – FeedbackForm.php

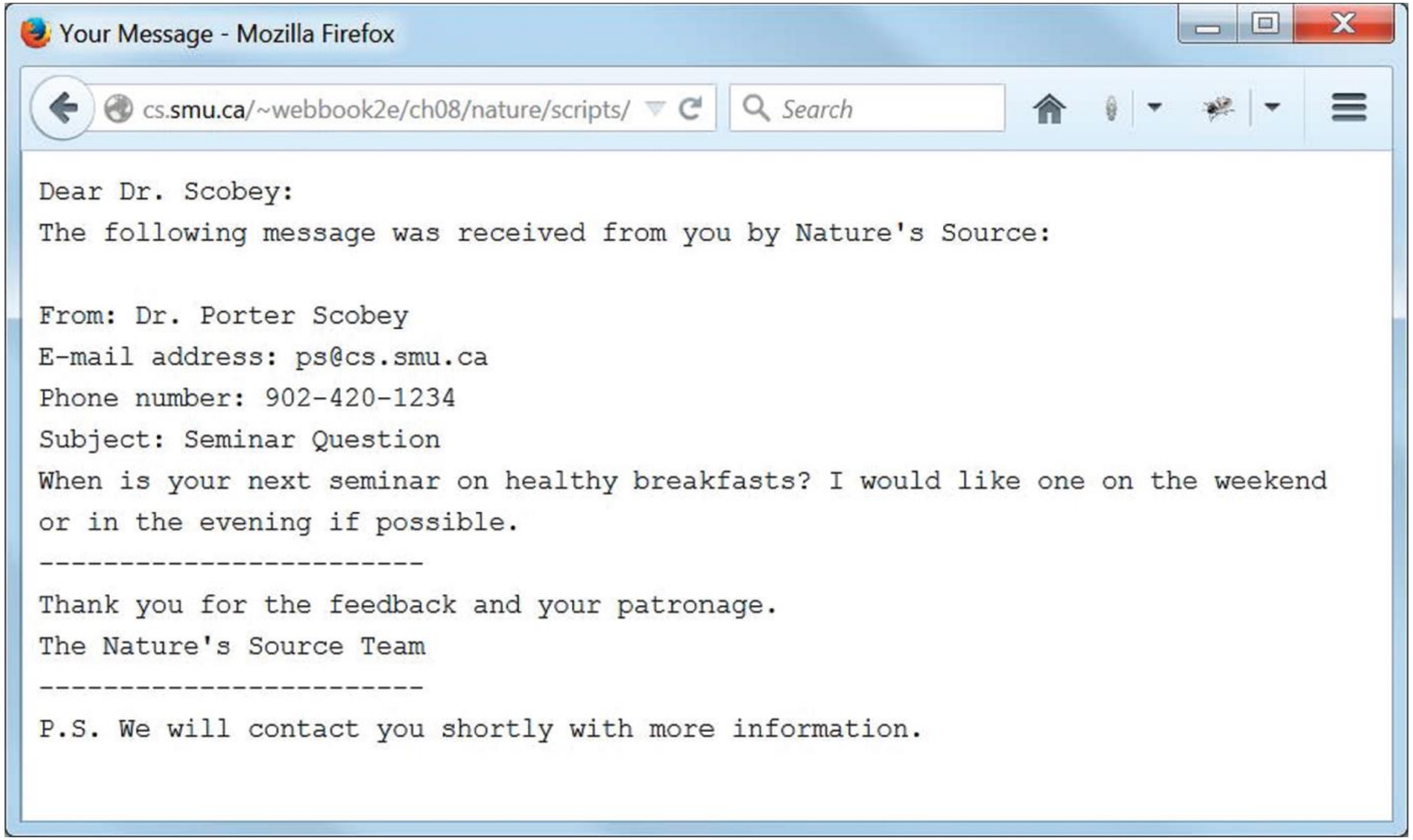
- Invoke the server-side php

```
<form id="contactForm"
action="scripts/feedbackFormProcess.php"
              method="post">
// …
</form>
```

- Functionalities
  - sends an e-mail message to the business
  - sends a slightly modified e-mail message to the client
  - returns a confirmation message to the client's browser
  - logs the feedback in a file on the server

Nature Source – Feedback

- Screen confirmation

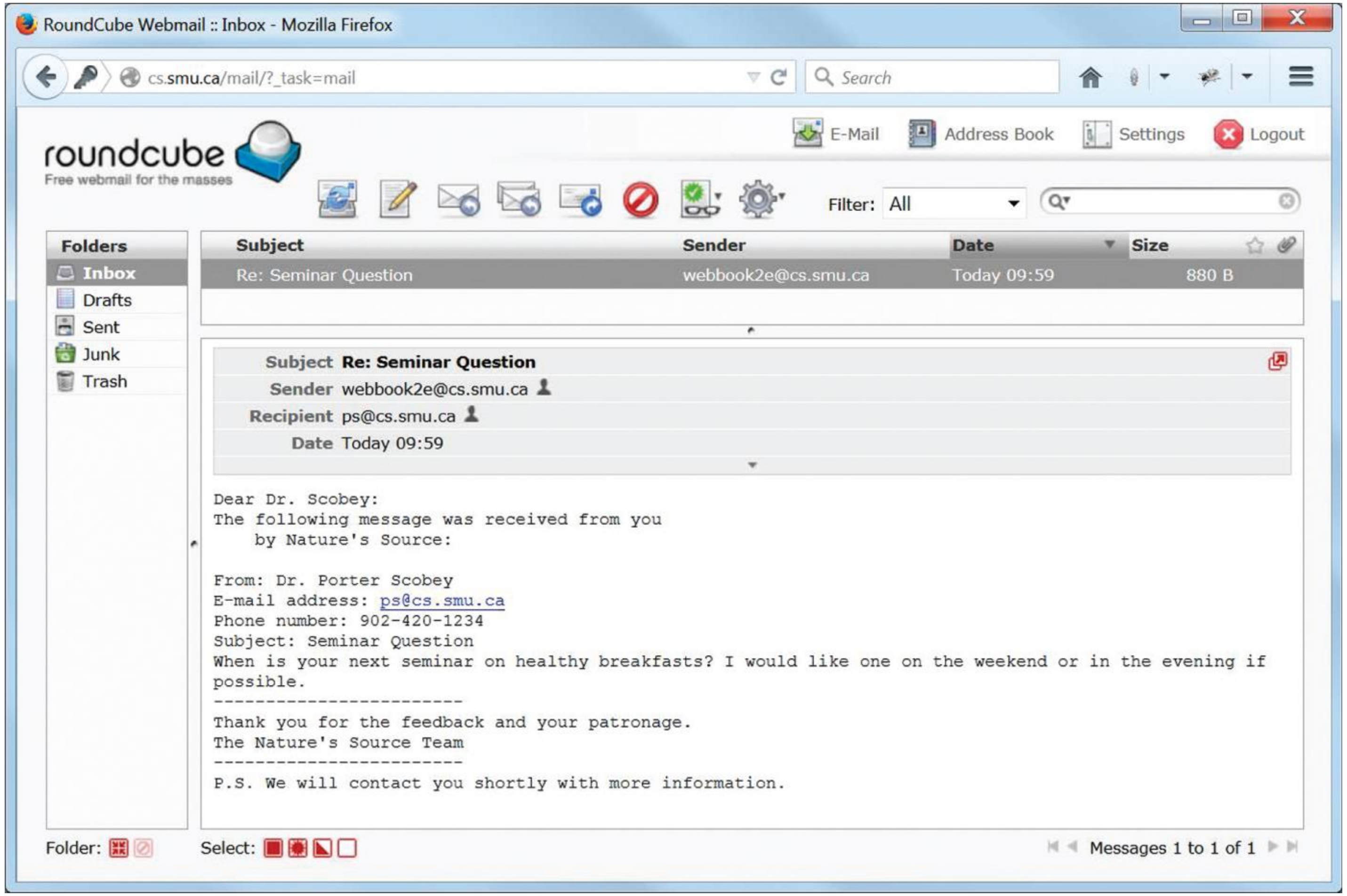

Your Message - Mozilla Firefox
cs.smu.ca/~webbook2e/ch08/nature/scripts/
Search

```
Dear Dr. Scobey:
The following message was received from you by Nature's Source:

From: Dr. Porter Scobey
E-mail address: ps@cs.smu.ca
Phone number: 902-420-1234
Subject: Seminar Question
When is your next seminar on healthy breakfasts? I would like one on the weekend
or in the evening if possible.
------------------------
Thank you for the feedback and your patronage.
The Nature's Source Team
------------------------
P.S. We will contact you shortly with more information.
```

Nature Source – Feedback

- Email



Subject	Sender	Date	Size
Re: Seminar Question	webbook2e@cs.smu.ca	Today 09:59	880 B

Subject Re: Seminar Question
Sender webbook2e@cs.smu.ca
Recipient ps@cs.smu.ca
Date Today 09:59

```
Dear Dr. Scobey:
The following message was received from you
    by Nature's Source:

From: Dr. Porter Scobey
E-mail address: ps@cs.smu.ca
Phone number: 902-420-1234
Subject: Seminar Question
When is your next seminar on healthy breakfasts? I would like one on the weekend or in the evening if
possible.
------------------------
Thank you for the feedback and your patronage.
The Nature's Source Team
------------------------
P.S. We will contact you shortly with more information.
```

Nature Source – index.php

- Logging

```
-------------------------------------------------------
Date received: 28th of April, 2015 at 19:45:53
From: Dr. Porter Scobey
E-mail address: p_scobey@yahoo.ca
Phone number: 902-420-5790
Subject: Question
Where ...?

-------------------------------------------------------
Date received: 19th of August, 2015 at 09:43:55
From: Dr. Porter Scobey
E-mail address: ps@cs.smu.ca
Phone number: 902-420-1234
Subject: Seminar Question
When is your next seminar on healthy breakfasts?
I would like one on the weekend or in the evening if possible.
```

feedbackFromProcess.php

- Build the main content of emails

```php
//Construct e-mail confirmation message for the client,
//which is just a sligtly modified version of the message
//that went to the business
$messageToClient =
    "Dear $_POST[salute] $_POST[lastName]:\r\n".
    "The following message was received from you
    by Nature's Source:\r\n\r\n".
    $messageToBusiness.
    "------------------------\r\n".
    "Thank you for the feedback and your patronage.\r\n".
    "The Nature's Source Team\r\n".
    "------------------------\r\n";

if (isset($_POST['reply'])) $messageToClient.=
    "P.S. We will contact you shortly with more information.";
```

feedbackFromProcess.php

- Send email out

```
//Sends e-mail confirmation message to the client
$headerToClient = "From: webbook2e@cs.smu.ca\r\n";
```

```
mail($_POST['email'], "Re: $_POST[subject]",
    $messageToClient, $headerToClient);
```

```
mail(
    string $to,
    string $subject,
    string $message,
    array|string $additional_headers = [],
    string $additional_params = ""
): bool
```

feedbackFromProcess.php

- Show confirmation

```
//Transforms confirmation message to HTML 5 format for
//display in the client's browser
$display = str_replace("\r\n", "\r\n<br>", $messageToClient);
$display = "<!DOCTYPE html>
    <html lang='en'>
    <head><meta charset='utf-8'><title>Your Message</title></head>
    <body><code>$display</code></body>
    </html>";
echo $display;
```

feedbackFromProcess.php

- Logging in Filesystem

```
//Logs the message in data/feedback.txt on the web server
//Note: directory "data" is at same level as directory "scripts"
$fileVar = fopen("../data/feedback.txt", "a")
    or die("Error: Could not open the log file.");

fwrite($fileVar,
    "\n-------------------------------------------------------\n")
    or die("Error: Could not write divider to the log file.");

fwrite($fileVar, "Date received: ".date("jS \of F, Y \a\\t
H:i:s\n"))
    or die("Error: Could not write date to the log file.");

fwrite($fileVar, $messageToBusiness)
    or die("Error: Could not write message to the log file.");
?>
```

feedbackFromProcess.php

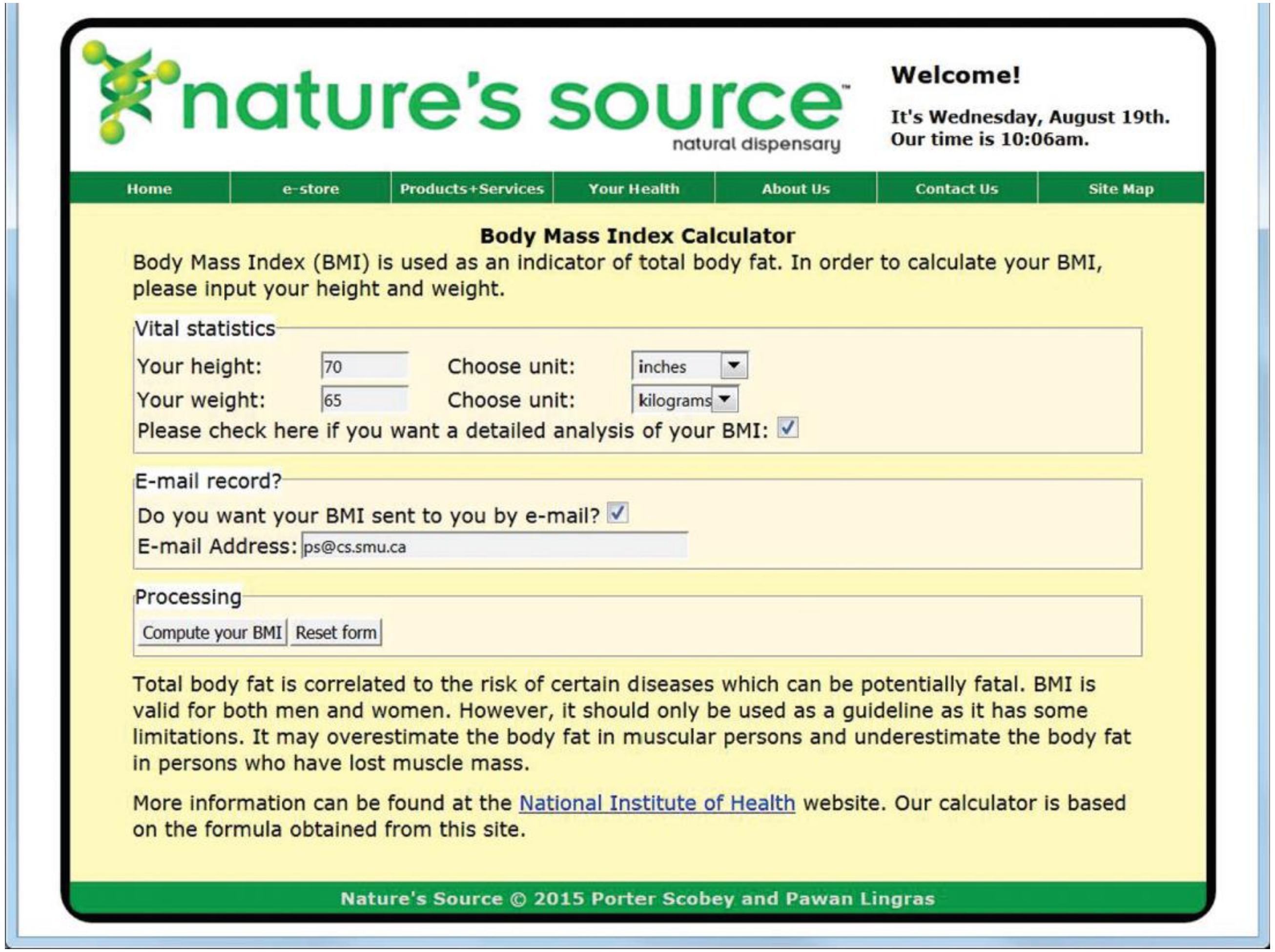
- Results

```
Dear Dr. Xyz:
The following message was received from you
by Nature's Source:

From: Dr. Abc Xyz
E-mail address: ab@unl.edu
Phone number: 000-000-0000
Subject: test
comment
------------------------
Thank you for the feedback and your patronage.
The Nature's Source Team
------------------------
P.S. We will contact you shortly with more information.
```

Nature Source–bmiFormProcess.php

- Functionalities
  - Server-side process
  - Send email results



Welcome!

It's Wednesday, August 19th. Our time is 10:06am.

Body Mass Index Calculator

Body Mass Index (BMI) is used as an indicator of total body fat. In order to calculate your BMI, please input your height and weight.

Vital statistics

Your height: 70 Choose unit: inches

Your weight: 65 Choose unit: kilograms

Please check here if you want a detailed analysis of your BMI: ☑

E-mail record?

Do you want your BMI sent to you by e-mail? ☑

E-mail Address: ps@cs.smu.ca

Processing

Total body fat is correlated to the risk of certain diseases which can be potentially fatal. BMI is valid for both men and women. However, it should only be used as a guideline as it has some limitations. It may overestimate the body fat in muscular persons and underestimate the body fat in persons who have lost muscle mass.

More information can be found at the National Institute of Health website. Our calculator is based on the formula obtained from this site.

Nature's Source © 2015 Porter Scobey and Pawan Lingras

Nature Source–bmiFormProcess.php

- Get data from _POST
- Mail message out

```
<?php
    include 'bmiCalculate.php';
    if (isset($_POST['details']))
        $message = detailedMessage($_POST['height'], $_POST['heightUnit'],
                                   $_POST['weight'], $_POST['weightUnit']);
    else
        $message = simpleMessage($_POST['height'], $_POST['heightUnit'],
                                 $_POST['weight'], $_POST['weightUnit']);
    echo $message;
    if (isset($_POST['wantMail']))
    {
        mailBMI($_POST['email'], $message);
        echo "<h2>The report has also been sent to you via e-mail.</h2>";
    }
?>
```

Nature Source – bmiCalculate.php

- Simplemessage()
  - Process the data with PHP

```php
//Constructs and returns the simple form of the BMI message
function simpleMessage($height, $heightUnit, $weight, $weightUnit)
{
   $bmi = sprintf("%1.2f", calculateBMI($height, $heightUnit,
                                        $weight, $weightUnit));
   $text = "<h1>BMI Report</h1><h3>Your BMI is $bmi.</h3>";
   return $text;
}
```

Nature Source – bmiCalculate.php

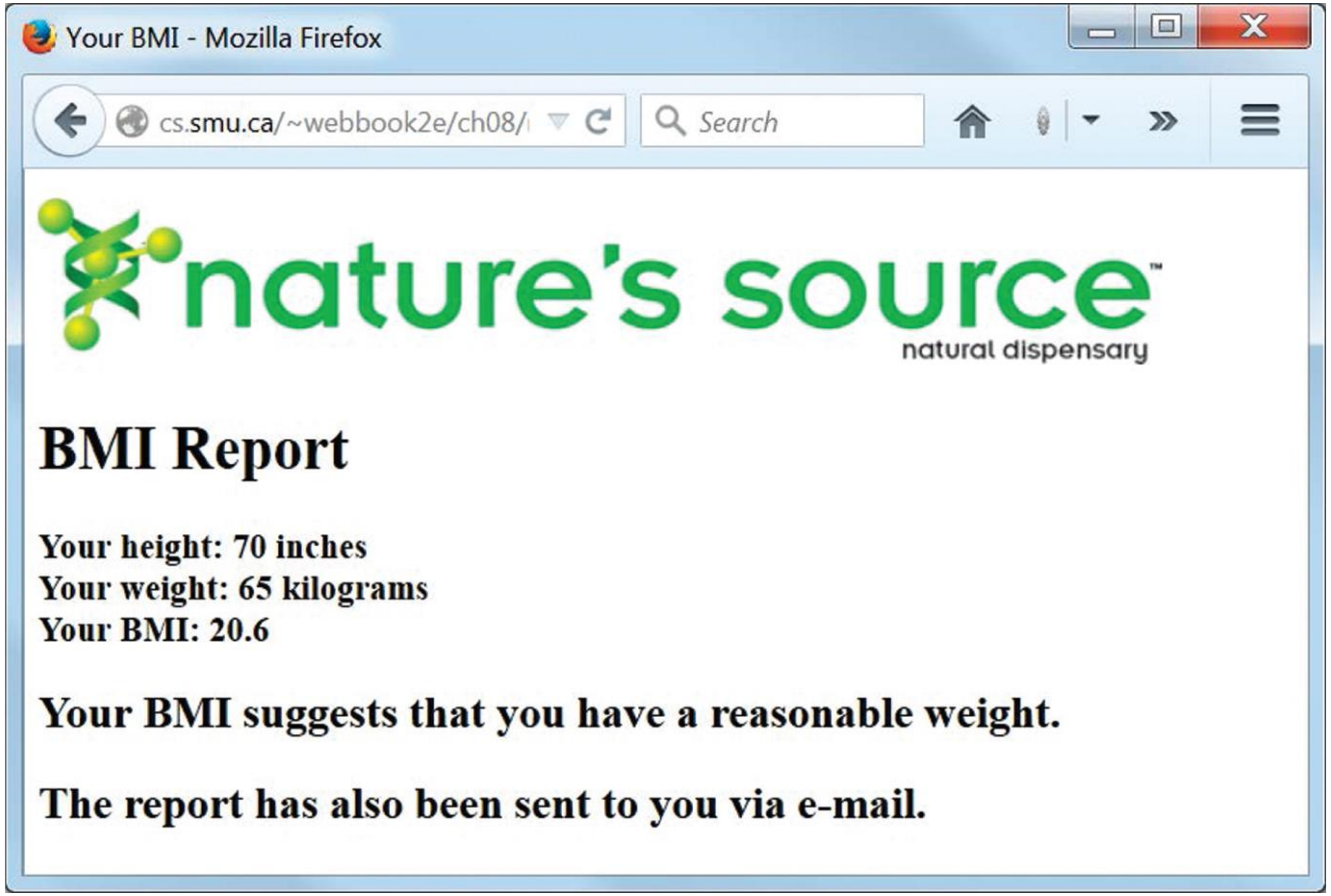
- mailBMI()
  - Use “mail()” function in PHP to send email out

```
mail(
    string $to,
    string $subject,
    string $message,
    array|string $additional_headers = [],
    string $additional_params = ""
): bool
```

```
//Constructs and sends the e-mail message as HTML
function mailBMI($email, $message)
{
    $header  = "MIME-Version: 1.0\r\n";
    $header .= "Content-type: text/html; charset=utf-8\r\n";
    $header .= "From: webbook2e@cs.smu.ca\r\n";
    mail($email, "BMI report from Nature's Source", $message, $header);
}
```

Nature Source – bmiCalculate.php

- Result



BMI Report

Your height: 70 inches
Your weight: 65 kilograms
Your BMI: 20.6

Your BMI suggests that you have a reasonable weight.

The report has also been sent to you via e-mail.

Practice

- ❖ Play with ch08/nature/index.php
  - Understand how server-side process works
  - Think about the placement of the functionalities in your final project, either in the client or server side