

[IS113] Extra Exercises - Weeks 4-5 - Form Processing

Objectives

- To master the concepts of HTML form elements and form processing
- To be able to differentiate between POST and GET
- To practice on retrieving form input values via POST and GET

Instructions

- Questions with no asterisk mark are easy peasy.
- Questions marked with * are slightly challenging.
- Questions marked with ** are challenging.
- Questions marked with *** are very challenging.

Download

- **Resources:** Click [here](#)

NOTE: If you spot any mistakes/errors in the questions, please contact your instructors by email and state the issues. We will try to address it as soon as possible.

Question 1 - Hello

Go to `hello` directory. Complete the following **Parts A** and **B** inside `sent.php` file.

`hello.html` page has TWO form input fields (TEXT).

Hello Form

Message:

Number:

Submit

Part A (*)

Complete `sent.php` file so that it takes the **Message** and prints it **Number** number of times.
Given the above input, `sent.php` page displays the following **HTML Table**.

S/N	Message
1	Hello
2	Hello
3	Hello

Part B ()**

Modify `sent.php` file so that it performs **Form Input Validation**.

hello.html	sent.php								
<div>Message: <input type="text"/></div> <div>Number: <input type="text"/></div> <div>Submit</div>	<div>1. Why No Message?</div> <div>2. Why No Number?</div>								
<div>Message: <input type="text" value="Hello"/></div> <div>Number: <input type="text"/></div> <div>Submit</div>	<div>1. Why No Number?</div>								
<div>Message: <input type="text"/></div> <div>Number: <input type="text" value="3"/></div> <div>Submit</div>	<div>1. Why No Message?</div>								
<div>Message: <input type="text" value="Hello"/></div> <div>Number: <input type="text" value="3.68"/></div> <div>Submit</div>	<div>1. Num is not an Integer</div>								
<div>Message: <input type="text" value="Hello"/></div> <div>Number: <input type="text" value="3"/></div> <div>Submit</div>	<table><tr><th>S/N</th><th>Message</th></tr><tr><td>1</td><td>Hello</td></tr><tr><td>2</td><td>Hello</td></tr><tr><td>3</td><td>Hello</td></tr></table>	S/N	Message	1	Hello	2	Hello	3	Hello
S/N	Message								
1	Hello								
2	Hello								
3	Hello								

Question 2 - Fruits

Go to `fruits` directory. Complete the following **Parts A** and **B** inside `sent.php` file.

`fruit.html` page allows the user to select a **fruit** and **quantity**.

Fruit Form

Choose Your Most Favorite Fruit

☐ Apple ☒ Orange ☐ Pear

Next, choose quantity

3 ▾

Submit

Part A (*)

Complete `sent.php` file so that it takes the **fruit's name** (e.g. "orange") and displays the selected fruit's image file **Quantity** number of times. Given the above input, `sent.php` page displays the following:



The image files are located in `/is113/extra4/fruits/images/`.

Part B (**)

Modify `sent.php` file so that it performs **Form Input Validation**.

fruit.html	sent.php
<p>Choose Your Most Favorite Fruit</p> <p><input type="radio"/> Apple <input type="radio"/> Orange <input type="radio"/> Pear</p> <p>Next, choose quantity</p> <p>2 ▾</p>	<p>Must select fruit</p>
<p>Choose Your Most Favorite Fruit</p> <p><input checked="" type="radio"/> Apple <input type="radio"/> Orange <input type="radio"/> Pear</p> <p>Next, choose quantity</p> <p>1 ▾</p>	
<p>Choose Your Most Favorite Fruit</p> <p><input type="radio"/> Apple <input type="radio"/> Orange <input checked="" type="radio"/> Pear</p> <p>Next, choose quantity</p> <p>4 ▾</p>	

Question 3 - K-Pop

Go to `kpop` directory. Complete the following **Parts A** and **B** inside `sent.php` file.

`kpop.html` page allows the user to select K-Pop celebrity names.

K-POP Form

Choose your favorite K-Pop stars!

☒ Hyori ☒ Hyuna ☐ Sandara

Submit

Part A (*)




Complete `sent.php` file so that it takes the user input (celebrity names) and displays their images. Given the above input, `sent.php` page displays the following:



The image files are located in `/is113/extra4/kpop/images/`.

Part B (**)

Modify `sent.php` file so that it performs **Form Input Validation**.

<code>kpop.html</code>	<code>sent.php</code>
<p>Choose your favorite K-Pop stars!</p> <p><input type="checkbox"/> Hyori <input type="checkbox"/> Hyuna <input type="checkbox"/> Sandara</p>	<p>OMG nobody selected</p>
<p>Choose your favorite K-Pop stars!</p> <p><input type="checkbox"/> Hyori <input checked="" type="checkbox"/> Hyuna <input type="checkbox"/> Sandara</p>	
<p>Choose your favorite K-Pop stars!</p> <p><input checked="" type="checkbox"/> Hyori <input checked="" type="checkbox"/> Hyuna <input type="checkbox"/> Sandara</p>	
<p>Choose your favorite K-Pop stars!</p> <p><input checked="" type="checkbox"/> Hyori <input checked="" type="checkbox"/> Hyuna <input checked="" type="checkbox"/> Sandara</p>	

Question 4 - Login

Go to `login` directory. Complete **Part A** below.

`login.html` page allows the user to key in **username**, **password** and retype the password (**confirm password**).

Login Form

Username

donald

Password

Confirm Password

Submit

Part A (**)

Page `sent.php` is **broken**. Fix the code so that it performs **Form Input Validation** as shown below:

login.html	sent.php
<div><div>Username</div><div>donald</div></div> <div><div>Password</div><div>*****</div></div> <div><div>Confirm Password</div><div>*****</div></div> <div>Password: trump123</div> <div>Confirm Password: trump123</div>	Wow! Login Successful!
<div><div>Username</div><div>george</div></div> <div><div>Password</div><div>*****</div></div> <div><div>Confirm Password</div><div>*****</div></div> <div>Password: trump123</div> <div>Confirm Password: trump123</div>	Username doesn't exist
<div><div>Username</div><div>donald</div></div> <div><div>Password</div><div>*****</div></div> <div><div>Confirm Password</div><div>*****</div></div> <div>Password: trump123</div> <div>Confirm Password: abcdefgh</div>	Passwords don't match
<div><div>Username</div><div>donald</div></div> <div><div>Password</div><div></div></div> <div><div>Confirm Password</div><div></div></div> <div>Password: <empty></div> <div>Confirm Password: <empty></div>	Password non-existent or empty
<div><div>Username</div><div>donald</div></div> <div><div>Password</div><div>*****</div></div> <div><div>Confirm Password</div><div></div></div> <div>Password: trump123</div> <div>Confirm Password: <empty></div>	Confirm Password non-existent or empty
<div><div>Username</div><div></div></div> <div><div>Password</div><div>*****</div></div> <div><div>Confirm Password</div><div>*****</div></div> <div>Password: trump123</div> <div>Confirm Password: trump123</div>	Username non-existent or empty

Question 5 - Asterisk

Go to **asterisk** directory. Complete the following **Parts A** and **B**.

Both pages **ex1.html** and **ex2.html** have ONE form input field (TEXT).

Num:

Part A (**)

Complete **ex1.php** file so that it takes the **Num** (Integer value) and prints **Num** lines of one or more asterisks. Given the above input **5**, it prints ONE line of 1 asterisk, ONE line of 2 asterisks, ONE line of 3 asterisks, ONE line of 4 asterisks and ONE last line of **5** asterisks. Given the input of ZERO **Num**, **ex1.php** would not print anything.

```
*
* *
* * *
* * * *
* * * * *
```

Part B (**)

Complete **ex2.php** file so that it takes the **Num** (Integer value) and prints **Num** lines of one or more asterisks. Given the above input **5**, it is to print the following shape. Given the input of ZERO **Num**, **ex2.php** would not print anything.

```
*
* *
* * *
* * * *
* * * * *
* * * * *
* * * * *
* * * *
* * *
* *
*
```

Question 6 - Like

Go to `like` directory. Complete the following **Part A**.

Page `like.php` has TWO (2) submit buttons and ONE checkbox.

People say that nothing is impossible but I do nothing all the time

Like

Haha

☐ Bookmark

Part A (**)

Complete `like.php` file so that the following are performed:

- 1. The **FORM** inside `like.php` submits back to this same file.
- 2. If the user clicks 'Like' button, `like.php` must display 'You like it!'.
- 3. If the user clicks 'Haha' button, `like.php` must display 'You find it funny.'
- 4. If the user checks 'Bookmark' checkbox, `like.php` must also display ' Bookmarked!'.

like.php	like.php (after clicking a submit button)
<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>Click on 'Like' submit button</div>	<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>You like it!</div>
<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>Click on 'Haha' submit button</div>	<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>You find it funny.</div>
<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input checked="" type="checkbox"/> Bookmark</div> <div>Click on 'Like' submit button</div>	<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>You like it! Bookmarked!</div>
<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input checked="" type="checkbox"/> Bookmark</div> <div>Click on 'Haha' submit button</div>	<div>People say that nothing is impossible but I do nothing all the time</div> <div>LikeHaha<input type="checkbox"/> Bookmark</div> <div>You find it funny. Bookmarked!</div>

Question 7 - Calculator

Go to `calculator` directory. Complete the following **Parts A, B, and C**.

Part A (*)

`two.html` page has TWO form input fields (TEXT) and ONE (1) drop-down menu. Complete `two.php` file so that it takes TWO (2) numbers and ONE (1) operator (e.g. plus, minus, multiply, divide). The page then performs the calculation and displays the calculation result. You don't have to perform form input validation at this time. You may assume that `two.php` is guaranteed to have all form input fields passed to it in correct format.

Test Cases

<code>two.html</code>	<code>two.php</code>
<input type="text" value="2"/> + <input type="text" value="3"/> <input type="button" value="Calculate"/>	<u>Result: 5</u>
<input type="text" value="2"/> * <input type="text" value="3"/> <input type="button" value="Calculate"/>	Result: 6
<input type="text" value="2"/> / <input type="text" value="0"/> <input type="button" value="Calculate"/>	Result: Undefined

Part B (**)

Modify `two.php` file so that it performs **Form Input Validation** as shown below.

<code>two.html</code>	<code>two.php</code>
<input type="text"/> + <input type="text"/> <input type="button" value="Calculate"/>	<u><ul style="list-style-type: none">• num1 is missing• num2 is missing</u>
<input type="text" value="2"/> + <input type="text"/> <input type="button" value="Calculate"/>	<ul style="list-style-type: none">• num2 is missing
<input type="text"/> + <input type="text" value="3"/> <input type="button" value="Calculate"/>	<ul style="list-style-type: none">• num1 is missing
<input type="text" value="abc"/> + <input type="text" value="2"/> <input type="button" value="Calculate"/>	<ul style="list-style-type: none">• num1 is non-numeric
<input type="text" value="2"/> + <input type="text" value="hello"/> <input type="button" value="Calculate"/>	<ul style="list-style-type: none">• num2 is non-numeric
<input type="text" value="hello"/> + <input type="text" value="layfoo"/> <input type="button" value="Calculate"/>	<ul style="list-style-type: none">• num1 is non-numeric• num2 is non-numeric

Part C (**)

`three.html` page has THREE form input fields (TEXT) and TWO (2) drop-down menus. Complete `three.php` file so that it takes all form input fields and their values. The page then performs the calculation

and displays the calculation result. You don't have to perform form input validation at this time. You may assume that `three.php` is guaranteed to have all form input fields passed to it in correct format.

IMPORTANT: Please make use of `calculate2()` function you completed in **Part A**.

Test Cases

three.html	three.php
<div>2 + 4 + 2</div> <div>Calculate</div>	Result: 8
<div>2 + 4 * 2</div> <div>Calculate</div>	Result: 10
<div>2 + 4 / 2</div> <div>Calculate</div>	Result: 4
<div>2 + 4 - 2</div> <div>Calculate</div>	Result: 4
<div>3 * 4 + 2</div> <div>Calculate</div>	Result: 14
<div>6 / 2 + 2</div> <div>Calculate</div>	Result: 5
<div>6 / 2 * 5</div> <div>Calculate</div>	Result: 15
<div>6 * 2 / 4</div> <div>Calculate</div>	Result: 3

Question 8 - Temperature Conversion

Go to `temp` directory. Complete the following **Part A**.

Part A (**)

`temp.php` page allows the user to perform temperature conversion. The page has ONE form input field (TEXT), ONE (1) drop-down menu and a SUBMIT button ("equals").

<code>temp.php</code>		
<input type="text"/>	degrees Fahrenheit	<input type="button" value="equals"/>

Complete `temp.php` page so that:

- The user can key in a numeric value in the form input field (temperature);
- The user can select the temperature scale (Fahrenheit or Celsius);
- The user can click SUBMIT, and the user is taken back to `temp.php` page. This time, `temp.php` page must perform:
 - a. Temperature conversion
 - If the user's input was "Fahrenheit", `temp.php` must convert the temperature to Celsius.
 - If the user's input was "Celsius", `temp.php` must convert the temperature to Fahrenheit.
 - b. Display the converted temperature value in HTML (next to "equals" SUBMIT button)
 - Converted temperature must be displayed to FIVE (5) decimal places
 - c. Remember the user's original input (temperature value **and** drop-down menu selection) and display them

Test Cases

<code>temp.php</code> (user input)	<code>temp.php</code> (after form submission)
98 degrees Fahrenheit	98 degrees Fahrenheit 36.66667 degrees Celsius.
24 degrees Celsius	24 degrees Celsius 75.20000 degrees Fahrenheit.
<input type="text"/> degrees Fahrenheit	<input type="text"/> degrees Fahrenheit
// empty input for temperature input field	// Display as above in case of empty input

Question 9 - Select

Go to `select` directory. Complete the following **Part A**.

Part A (*)

`select.php` page ONE (1) drop-down menu and a SUBMIT button (“equals”).

<code>select.php</code>
School: <input type="text" value="Business"/> <input type="button" value="Submit"/>

The page has a pre-populated Associative Array `$schools`:

<code>select.php</code>
<pre>\$schools = ['LKCSB' => 'Business', 'SOE' => 'Economics', 'SIS' => 'Information systems', 'SOL' => 'Law', 'SOA' => 'Accountancy', 'SOSS' => 'Social Sciences'];</pre>

Complete `select.php` page so that it displays a drop-down menu with the following options:

<div>School: <input type="text" value="Business"/> <input type="button" value="Submit"/></div> <div><div>Business</div><div>Economics</div><div>Information systems</div><div>Law</div><div>Accountancy</div><div>Social Sciences</div></div>	<pre>// EXAMPLE // <option value='LKCSB'>Business</option> // <option value='SOE'>Economics</option> // and so on... // DO NOT HARDCODE // Make use of the Associative Array \$schools</pre>
---	--

Part B (**)

Modify `select.php` page. After the user selects a **school** (e.g. Law) and clicks on **SUBMIT** button, the page performs the following:

- Use the option value (e.g. SOL) as **key** and retrieve the corresponding **value** from Associative Array `$messages`
- Display this **value** (message) BELOW the drop-down menu and ABOVE the SUBMIT button.
- Remember the user's original input (drop-down menu selection) and display it.
 - e.g. If the user selected “Law” and clicked SUBMIT, `select.php` page must pre-select Law option in the drop-down menu.
- See next page for **Test Cases**.

<pre><select name='school'> <option value='LKCSB'>Business</option> <option value='SIS'>Information Systems</option> <option value='SOE' selected>Economics</option> </select></pre>

Test Cases

select.php (user input)	select.php (after form submission)
First time visiting select.php	
School: <input type="text" value="Law"/> <input type="button" value="Submit"/>	School: <input type="text" value="Law"/> See you in court <input type="button" value="Submit"/>
Next, choose a different school in the drop-down menu	
School: <input type="text" value="Economics"/> <input type="button" value="Submit"/>	School: <input type="text" value="Economics"/> Inflation Time <input type="button" value="Submit"/>
Next, choose a different school in the drop-down menu	
School: <input type="text" value="Social Sciences"/> <input type="button" value="Submit"/>	School: <input type="text" value="Social Sciences"/> We Love People <input type="button" value="Submit"/>

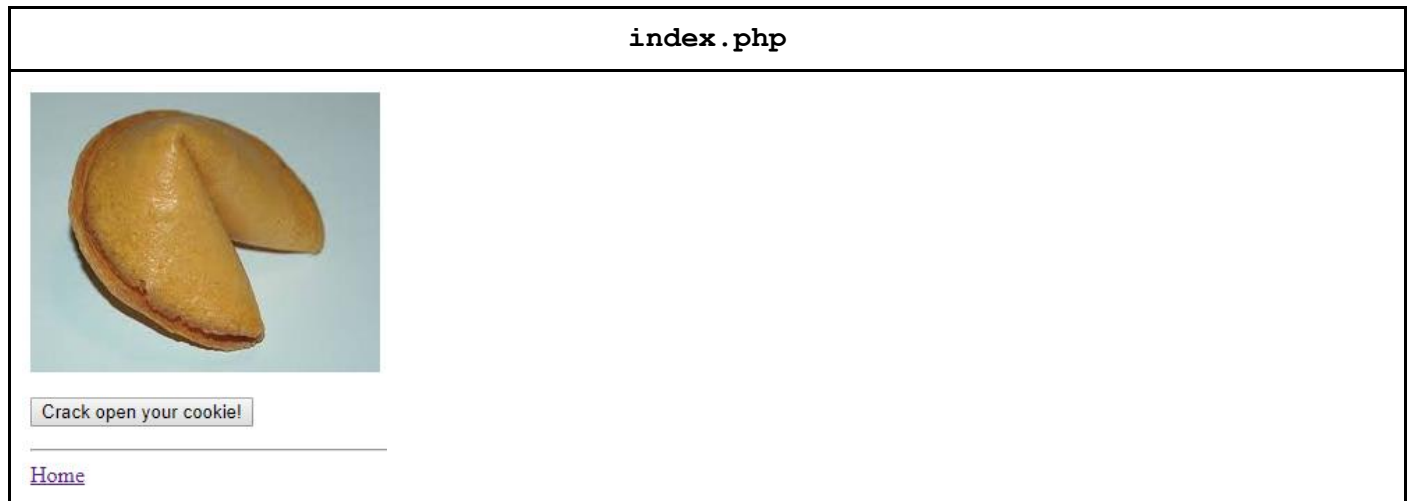
Question 10 - Fortune Cookie

Go to **fortune** directory. Complete the following **Part A**.

What makes cracking open a fortune cookie fun? You don't know what **quote** you'll get. In this exercise, you are to develop a webpage that will allow users to crack open a fortune cookie.

Part A (**)

index.php page, when loaded in a web browser for the first time, looks like below.









- The page displays a **closed** fortune cookie image.
- Below, the page displays a **SUBMIT** button ("Crack open your cookie!").
- Below a Horizontal Line, there is a Hyperlink back to itself (**index.php**).

Complete **index.php** page so that it performs the following:

- When the page loads for the first time, display a **closed** fortune cookie image (as shown above).
- When the user clicks **SUBMIT** button, **index.php** must:
 - **Randomly select** a **quote** from Associative Array **\$quotes** and display the **quote** in Size-1 Heading in HTML.
 - Below the **quote**, display an **open** fortune cookie image.
- At any point of time, when the user clicks on "**Home**" **hyperlink** at the bottom of the page, the page must treat it as if the page is loaded for the first time such that:
 - The page displays a **closed** fortune cookie image.
 - Below, the page displays a **SUBMIT** button ("Crack open your cookie!").
 - Below a Horizontal Line, there is a Hyperlink back to itself (**index.php**).
- See next page for **Test Cases**

Test Cases

index.php (user input)	index.php (after form submission)
First time visiting index.php	
<div><div>Crack open your cookie!</div><hr/>Home</div> <p>// Click SUBMIT button</p>	<div><p>If you judge people, you have no time to love them. -- Mother Teresa</p><div>Crack open your cookie!</div><hr/>Home</div> <p>// The quote has been randomly selected</p>
<div><p>If you judge people, you have no time to love them. -- Mother Teresa</p><div>Crack open your cookie!</div><hr/>Home</div> <p>// Click SUBMIT button</p>	<div><p>Great things in business are never done by one person. They're done by a team of people. - Steve Jobs</p><div>Crack open your cookie!</div><hr/>Home</div> <p>// The quote has been randomly selected</p>
<div><p>Great things in business are never done by one person. They're done by a team of people. - Steve Jobs</p><div>Crack open your cookie!</div><hr/>Home</div> <p>// Click on “Home” hyperlink at the bottom</p>	<div><div>Crack open your cookie!</div><hr/>Home</div>

Question 11 - Validate & Forward

Go to `validate` directory. Complete the following **Part A**.

Part A (**)

`signup.php` page allows the user to specify two kinds of input.

<code>signup.php</code>
<h2>Krazy Gym</h2> <p>Reason for joining (must select at least ONE):</p> <p><input type="checkbox"/> Lose Weight <input type="checkbox"/> Find Love <input type="checkbox"/> Build Muscles</p> <p>Gym type (must select at least ONE):</p> <p><input type="radio"/> Men only <input type="radio"/> Women only <input type="radio"/> Anything</p> <p><input type="button" value="Submit"/></p>

The user **MUST** specify **BOTH** the 1) reason (for joining) and 2) gym type. Complete `signup.php` page so that it performs **form validation**.

Test Cases

<code>signup.php</code> (user input)	<code>signup.php</code> (after form submission)
<h2>Krazy Gym</h2> <p>Reason for joining (must select at least ONE):</p> <p><input type="checkbox"/> Lose Weight <input type="checkbox"/> Find Love <input type="checkbox"/> Build Muscles</p> <p>Gym type (must select at least ONE):</p> <p><input type="radio"/> Men only <input type="radio"/> Women only <input type="radio"/> Anything</p> <p><input type="button" value="Submit"/></p> <p>// NOTE that the first time this page loads, it must show the form as shown above. // The error messages must NOT be shown.</p>	<h2>Krazy Gym</h2> <ul style="list-style-type: none">• Must select reason(s)• Must select type <p>Reason for joining (must select at least ONE):</p> <p><input type="checkbox"/> Lose Weight <input type="checkbox"/> Find Love <input type="checkbox"/> Build Muscles</p> <p>Gym type (must select at least ONE):</p> <p><input type="radio"/> Men only <input type="radio"/> Women only <input type="radio"/> Anything</p> <p><input type="button" value="Submit"/></p>

Krazy Gym

Reason for joining (must select at least ONE):

- ☒ Lose Weight
- ☐ Find Love
- ☐ Build Muscles

Gym type (must select at least ONE):

- ☐ Men only
- ☐ Women only
- ☐ Anything

Krazy Gym

- Must select type

Reason for joining (must select at least ONE):

- ☒ Lose Weight
- ☐ Find Love
- ☐ Build Muscles

Gym type (must select at least ONE):

- ☐ Men only
- ☐ Women only
- ☐ Anything

Krazy Gym

Reason for joining (must select at least ONE):

- ☐ Lose Weight
- ☐ Find Love
- ☐ Build Muscles

Gym type (must select at least ONE):

- ☐ Men only
- ☒ Women only
- ☐ Anything

Krazy Gym

- Must select reason(s)

Reason for joining (must select at least ONE):

- ☐ Lose Weight
- ☐ Find Love
- ☐ Build Muscles

Gym type (must select at least ONE):

- ☐ Men only
- ☒ Women only
- ☐ Anything

Krazy Gym

Reason for joining (must select at least ONE):

- ☐ Lose Weight
- ☒ Find Love
- ☒ Build Muscles

Gym type (must select at least ONE):

- ☐ Men only
- ☐ Women only
- ☒ Anything

Forwarded to **thankyou.php**

(Hint: use `header("Location: thankyou.php")`; More on week 12.)

Thank you for signing up!

Question 12 - Missing index.html (*)

This question is to recap HTML and basic form processing concepts.

Go to `missing_index` directory.

The file `index.html` has a form that submits to `receive.php`.

However, the file `index.html` is lost. Fortunately, we know that `index.html` looks like this when it is loaded in a web browser.

Title / Name	<input type="text"/>		
Language	<input type="radio"/> English <input type="radio"/> Chinese <input type="radio"/> Others	Media Format	<input type="checkbox"/> Hardcopies (e.g. Book)
Suitable for	<div>All ages All ages Above 12 year old 18 years old and above</div>		<input type="checkbox"/> DVD
			<input type="checkbox"/> Streaming (e.g. Spotify, YouTube)
			<input type="button" value="Send the form!"/>

The displayed choices for the respective fields are:

Language

- English
- Chinese
- Others

Suitable for

- All ages
- Above 12 year old
- 18 years old and above

Media Format

- Hardcopies (e.g. Book)
- DVD
- Streaming (e.g. Spotify, YouTube)

And we also have `receive.php` (available in the resource folder 'missing_index').

To do:

You are to create `index.html`

- Based on the above screenshot and
- Such that the given `receive.php` will process the submitted form values successfully.
 - You may assume `receive.php` is correct.
 - You must NOT edit `receive.php`.

Question 13 - Text (*)

This question is to recap HTML and basic form processing concepts.

Go to **text** directory.

The file `index.html` has a form that submits to `text.php`.

User submits with the following values	<code>text.php</code>								
<p>When <code>index.html</code> is loaded in a web browser for the first time.</p> <table><tr><td>Message</td><td><input type="text" value="Enter your message"/></td></tr><tr><td>Style</td><td><input type="checkbox"/> Italic <input type="checkbox"/> Bold</td></tr><tr><td>Colour</td><td><input type="text" value="Black"/></td></tr><tr><td>Background color</td><td><input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.</td></tr></table> <p><input type="button" value="Go!"/></p> <p>User submits without entering/changing any value.</p>	Message	<input type="text" value="Enter your message"/>	Style	<input type="checkbox"/> Italic <input type="checkbox"/> Bold	Colour	<input type="text" value="Black"/>	Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.	<p>Message cannot be empty</p> <p>Displays error: Message cannot be empty</p>
Message	<input type="text" value="Enter your message"/>								
Style	<input type="checkbox"/> Italic <input type="checkbox"/> Bold								
Colour	<input type="text" value="Black"/>								
Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.								
<table><tr><td>Message</td><td><input type="text" value="I love PHP!"/></td></tr><tr><td>Style</td><td><input checked="" type="checkbox"/> Italic <input checked="" type="checkbox"/> Bold</td></tr><tr><td>Colour</td><td><input type="text" value="Red"/></td></tr><tr><td>Background color</td><td><input type="radio"/> Cyan <input type="radio"/> Magenta <input checked="" type="radio"/> Yellow If none is selected, background color is white.</td></tr></table> <p><input type="button" value="Go!"/></p> <ul style="list-style-type: none">• Message is 'I love PHP!'• Select both styles 'Italic' and 'Bold'• Select 'Red' for color and 'Yellow' for background	Message	<input type="text" value="I love PHP!"/>	Style	<input checked="" type="checkbox"/> Italic <input checked="" type="checkbox"/> Bold	Colour	<input type="text" value="Red"/>	Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input checked="" type="radio"/> Yellow If none is selected, background color is white.	<p><i>I love PHP!</i></p> <p>The message is italics, bold and red in color.</p> <p>The background of the whole page is yellow.</p>
Message	<input type="text" value="I love PHP!"/>								
Style	<input checked="" type="checkbox"/> Italic <input checked="" type="checkbox"/> Bold								
Colour	<input type="text" value="Red"/>								
Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input checked="" type="radio"/> Yellow If none is selected, background color is white.								
<table><tr><td>Message</td><td><input type="text" value="I love PHP!"/></td></tr><tr><td>Style</td><td><input type="checkbox"/> Italic <input type="checkbox"/> Bold</td></tr><tr><td>Colour</td><td><input type="text" value="Blue"/></td></tr><tr><td>Background color</td><td><input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.</td></tr></table> <p><input type="button" value="Go!"/></p> <ul style="list-style-type: none">• Message is 'I love PHP!'• Select 'Blue' for color	Message	<input type="text" value="I love PHP!"/>	Style	<input type="checkbox"/> Italic <input type="checkbox"/> Bold	Colour	<input type="text" value="Blue"/>	Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.	<p>I love PHP!</p> <p>The message is blue in color.</p> <p>The background of the whole page is white..</p>
Message	<input type="text" value="I love PHP!"/>								
Style	<input type="checkbox"/> Italic <input type="checkbox"/> Bold								
Colour	<input type="text" value="Blue"/>								
Background color	<input type="radio"/> Cyan <input type="radio"/> Magenta <input type="radio"/> Yellow If none is selected, background color is white.								

The file `index.html` is given in the resource folder '`text`'. Do **NOT** edit this file!

To do:

Create `text.php` according to the above requirements. You may use either CSS or HTML attributes/tags for the styling.

If you intend to use CSS (out-of-scope of IS113-WAD1),

```
<html>  
<body style='font-style: italic; font-weight: bold; color: red; background-  
color: yellow;' >
```

*The above CSS code will make
The background of the web page yellow
Text in italics, bold and red color*

```
</body>  
</html>
```

For more information

- https://www.w3schools.com/cssref/pr_font_font-style.asp
- https://www.w3schools.com/cssref/pr_font_weight.asp
- https://www.w3schools.com/Css/css_colors.asp

Question 14 - Guild

This question is to recap form re-population and array concepts.

Go to `guild` directory.

You are to edit the given `index.php` according to the instructions below.

- Do NOT change lines 3 to 19 of `index.php`.
- The elements in the arrays `$guilds` and `$heroes` may change.
 - Name of guilds and its members may change.
 - Name of heroes and values of their attributes may change. Each hero will always have attributes 'Strength' and ' Magic'; no additional attributes.
- You may assume there is at least one hero per guild.

Part A (**)

Create a form has a list of checkboxes that allow user to select one or more guilds.

For each checkbox, display the guild's name and the number of members. The guilds' names and list of members is from the array `$guilds`.

-
- When user clicks on the displayed text, the checkbox is ticked.

E.g. guild 'Black Star' has 3 members, thus, the display text is 'Black Star(3)'.

The form submits back to `index.php` via HTTP POST.

Guilds

☐ Black Star (3)

☐ Iconic (2)

☐ Crazy World (3)

Get

If user submits the form without selecting any guilds, display a message 'No guild selected' below the form.

Guilds

☐ Black Star (3)

☐ Iconic (2)

☐ Crazy World (3)

Get

No guild selected

If user selects one or more guilds then submits the form,

- Re-populate the form with the user's selection; i.e. the check boxes for the selected guilds remains ticked.
- Display the selected guilds' names and its members' names as an ordered list.

Guilds

☒ Black Star (3)

☒ Iconic (2)

☐ Crazy World (3)

Get

Black Star

1. Jam

2. Nisa

3. Rose

Iconic

1. Soo

2. Nisa

Part B (*)

Upgrade to display each hero's attributes which is obtainable from `$heroes`.

Guilds ☐ Black Star (3) ☒ Iconic (2) ☒ Crazy World (3)

Iconic

1. Soo : Strength 1 Magic 5
2. Nisa : Strength 3 Magic 3

Crazy World

1. Jam : Strength 4 Magic 1
2. Soo : Strength 1 Magic 5
3. Nisa : Strength 3 Magic 3

Part C (**)

Upgrade to display the guilds' details in a table.

For each guild, display its hero on each row.

- For each hero, display stars for strength and magic. E.g. Jam's strength and magic are 4 and 1, thus, 4 stars and 1 star are displayed in the corresponding cells.
- Display the total value for all the guild's heroes' strength and magic.
- For guild 'Black Star', the total strength of all its heroes' is 11 and total magic is 6.

Guilds ☒ Black Star (3) ☐ Iconic (2) ☒ Crazy World (3)

Guild	Hero	Strength	Magic
Black Star	Jam	★★★★	★
	Nisa	★★★★	★★★★
	Rose	★★★★	★★
	Total	11	6
Crazy World	Jam	★★★★	★
	Soo	★	★★★★★
	Nisa	★★★★	★★★★
	Total	8	9

Question 15 - Guild-Alliance (***)

Go to `guild_alliance` directory.

You are to edit the given `index.php` according to the instructions below.

- Do NOT change lines 3 to 42 of `index.php`.
 - The elements in the arrays `$alliances`, `$guilds` and `$heroes` may change.
 - Name of alliances and its members may change.
 - Name of guilds and its members may change.
 - Name of heroes and values of their attributes may change. Each hero will always have attributes 'Strength' and ' Magic'; no additional attributes.
 - You may assume there is at least one guild per alliance and one hero per guild.
1. Create a form has a list of checkboxes that allow user to select one or more alliances.
 - a. For each checkbox, display the alliance's name
 - i. The alliances' names and list of member guilds is from the array `$alliances`.
 - ii. When user clicks on the displayed text, the checkbox is ticked.
 - b. The form submits back to `index.php` via HTTP POST.

Alliances <input type="checkbox"/> All Stars <input type="checkbox"/> Iconic <input type="checkbox"/> Duel <input type="checkbox"/> <input type="button" value="Get"/>
--

2. If user submits the form without selecting any alliance, display a message 'No alliance selected' below the form.

Alliances <input type="checkbox"/> All Stars <input type="checkbox"/> Iconic <input type="checkbox"/> Duel <input type="checkbox"/> <input type="button" value="Get"/>
No alliance selected

3. If user selects one or more alliances then submits the form,
 - a. Re-populate the form with the user's selection; i.e. the check boxes for the selected alliances remains ticked.
 - b. Display the selected alliances' names and details of its member guilds in a table.
 - c. For each guild, displays its names and its member heroes' details.
 - i. Details of the guilds' member heroes are from the array `$guilds`.
 - ii. For each hero, display his/her name, and stars for strength and magic.
 1. Each hero's attributes which is obtainable from `$heroes`.
 2. E.g. Jam's strength and magic are 4 and 1, thus, 4 stars and 1 star are displayed in the corresponding cells.
 - iii. Display the minimum and maximum strength and magic among the guild's heroes'.
 1. E.g. For guild 'Black Star',
 - a. The minimum strength is 1 (hero 'Soo Ya').
 - b. The maximum strength is 4 (heroes 'Jam' and 'Rose').
 - c. The minimum magic is 1 (hero 'Jam').
 - d. The maximum magic is 5 (hero 'Soo Ya').

Alliances <input checked="" type="checkbox"/> All Stars <input type="checkbox"/> Iconic <input checked="" type="checkbox"/> Duel <input type="button" value="Get"/>				
Alliance	Guild	Hero	Strength	Magic
All Stars	Black Star	Jam	★★★★	★
		Ni Sa	★★★	★★★★
		Rose	★★★★	★★
		Soo Ya	★	★★★★★★
		Min	1	1
		Max	4	5
Duel	Ak Family	Ak Hyuk	★★	★★★★★
		Ak Hyun	★	★★★★★★
		Ak Pa	★★★	★★★★★★
		Ak Ma	★★★★★	★★★★★★
		Min	1	4
		Max	5	5
	Ukulele	Terry	★★★	★★★★★★
		Lee Chi	★★★★	★★★★★
		Min	3	4
		Max	4	5

Hints:

- Breakdown the problem into smaller parts.**
Do question 14 'Guild' first if you have not done so.
- Think through the logic.**
 - For each user's selected alliance
 - Calculate the number of rows that the alliance's table cell will span across.
 - For each member guild of the alliance
 - Calculate the number of rows that the guild's table cell will span across.
 - For each member hero of the guild
 - Display the hero's details
 - Display the heroes' statistics for this guild

3. Solve the smaller parts.

You may wish to create the following functions.

- function getAllianceRowspan(\$guilds, \$member_guilds)**

Calculates the number of rows that an alliance will span across.

Parameters

- `$guilds`: Dictionary of all guilds; key: name of guild. value: list of members aka heroes' name
- `$member_guilds`: List of names of guilds that are member of the alliance.

Return the number of rows that the alliance table cell should span across.

b. **function genGuildStats(\$heroes, \$guild_heroes)**

Calculate the heroes' min and max attributes for a guild.

Parameters

- i. `$heroes`: Dictionary of all heroes;
 1. key: member's name.
 2. value: dictionary of member's attributes; E.g. ['Strength' => 3, 'Magic' => 3]
- ii. `$guild_heroes`: List of names of heroes that are members of guild.

Return a nested array representing the statistics . Structure is as follows:

```
[
  'Min' => [
    'Strength' => int,
    'Magic' => int,
  ],
  'Max' => [
    'Strength' => int,
    'Magic' => int
  ]
]
```


Question 16 - Equation

Go to `equation` directory.

Complete `equation.html` (contains a form that uses method post) and `process_equation.php` (form handler) so that the following pages would be rendered:

This form will compute the following equation: $Z = X^2 + Y$

Please enter the value of X

Please enter the value of Y

`equation.html`

The value of Z is 37

`process_equation.php`

```
<!DOCTYPE html>
<html>
  <head>
    <title>Equation</title>
  </head>
  <body>
    This form will compute the following equation:  $Z = X^2 + Y$ <br/>
    <!-- Please enter code here -->

    <!-- End of code -->
  </body>
</html>
```

`equation.html`

```
<html>
  <head>
    <title>Equation</title>
  </head>
  <body>
    <?php
      // Please enter code here

      // End of code
    ?>
  </body>
</html>
```

process_equation.php

Question 17 - Bus Fare (*)

Go to **bus_fare** directory.

Complete bus_fare.php below so that it can compute the bus fare of a passenger given information about distance (in km), age group (<12, 12-60, or >60), time (peak or non-peak). During non-peak hours, the fare of a bus ride is 10 cents per km. During peak-hours, the fare is 15 cents per km. Children (<12) receives a 50% discount, while senior citizens (>60) receives a 25% discount.

```
<!DOCTYPE html>
<html>
  <head>
    <title>Compute Bus Fare in Far Far Away Land</title>
  </head>
  <body>
    <h2>Enter details</h2>
    <form action="bus_fare.php" method="post">
      Distance (in km):
      <input type="text" name="distance" size="5" required/><br/>
      Age group:
      <select name="age_group">
        <option value="children"><12</option>
        <option value="others" selected>12-60</option>
        <option value="senior">>60</option>
      </select><br/>
      Time:
      <input type="radio" name="time" value="Peak" checked/>Peak
      <input type="radio" name="time" value="Non-Peak"/>Non-Peak<br/>
      <input type="submit" value="Compute Fare"/>
    </form>
  </body>
</html>
```

bus.html

```
<!DOCTYPE html>
<html>
  <head>
    <title>Compute Bus Fare in Far Far Away Land</title>
  </head>
  <body>
    <?php
      $fare = 0.0;

      // Please enter code here

      // End of code

      echo "Fare is $" . $fare;
    ?>
  </body>
</html>
```

bus_fare.php

Example:

Enter details

Distance (in km):

Age group: ▼

Time: ☒ Peak ☐ Non-Peak

ex1.html

Fare is \$11.25

bus_fare.php

Question 18 - Prime Number (**)

Go to **prime_number** directory.

Implement a PHP program that checks whether a given number is a prime number. A partially completed php page is shown below:

```
<!DOCTYPE html>
<html>
<head>
    <title>Check Prime</title>
</head>
<body>
    <p>Please enter a whole number.</p>
    <form> <!-- If action is unspecified, form submits to itself -->
        <input name="number" type="text">
        <input type="submit" value="Go!">
    </form>

<?php
    if (isset($_GET['number'])) { // Loaded for first time?
        if (isValid($_GET['number'])) {
            $isPrime = true;

            // Please enter code here

            // End of code

            if ($isPrime) {
                echo "<p>".$_i." is a prime number!</p>";
            } else {
                echo "<p>".$_i." is not prime.</p>";
            }
        } else {
            // User submitted something which is not a positive whole number
            echo "<p>Please enter a positive whole number.</p>";
        }
    }

    // check if the given number is a valid numeric value
    // round() rounds a floating point value
```

```
function isValid($number) {  
    if(is_numeric($number) && $number > 0  
        && $number == round($number, 0)) {  
        return true;  
    } else {  
        return false;  
    }  
}  
?  
  
</body>  
</html>
```

prime.php

Question 19 - Binary Search (***)

Go to `binary_search` directory.

Implement a PHP program that performs a **binary search**

(https://en.wikipedia.org/wiki/Binary_search_algorithm) to compute the length of a “hidden” string stored in the system. A partially completed php page is shown below:

```
<?php
    if
(isset($_GET["submit"]) && $_GET["submit"]=="BinarySearch!") {
    // predefine the search range [1, 100]
    $hi = 100;
    $lo = 1;

    /* add your code that performs binary search to find the length of
    password string here. [Hint: You have to invoke the provided
    isEqual() and/or isLess() in a loop.]
    */

    /* end of code */
    echo "<p> Length of the password is: " . $lo . "</p>";
}

//----- Do not modify the following two functions -----
function isEqual($length) {
    $password = "This is a secret password!";
    if(strlen($password) == $length) {
        return true;
    } else {
        return false;
    }
}

function isLess($length) {
    $password = "This is a secret password!";
    if($length < strlen($password)) {
        return true;
    } else {
        return false;
    }
}
```

```
    }  
  }  
?>  
<form>  
  <input type="submit" name="submit" value="BinarySearch!">  
</form>
```

binary.php

The following output will be shown if the inserted code is correct:

Length of the password is: 26

BinarySearch!