



HTML Forms – Lab 5

IMPORTANT! Save all your work to a safe location such as OneDrive.

Create a folder for GUI & Web Development into which you will save all your work for this module, arranged how you wish. Ideally you should create a folder each week for your lab exercises. Note that you should create a separate file for each exercise.

This week we will work with forms and input elements. Form elements allow us to create different types of input elements such as text fields, password fields, checkboxes, radio buttons, submit buttons, and many more. They provide a way of allowing the user to input information.

Exercise 1: Create a basic input option

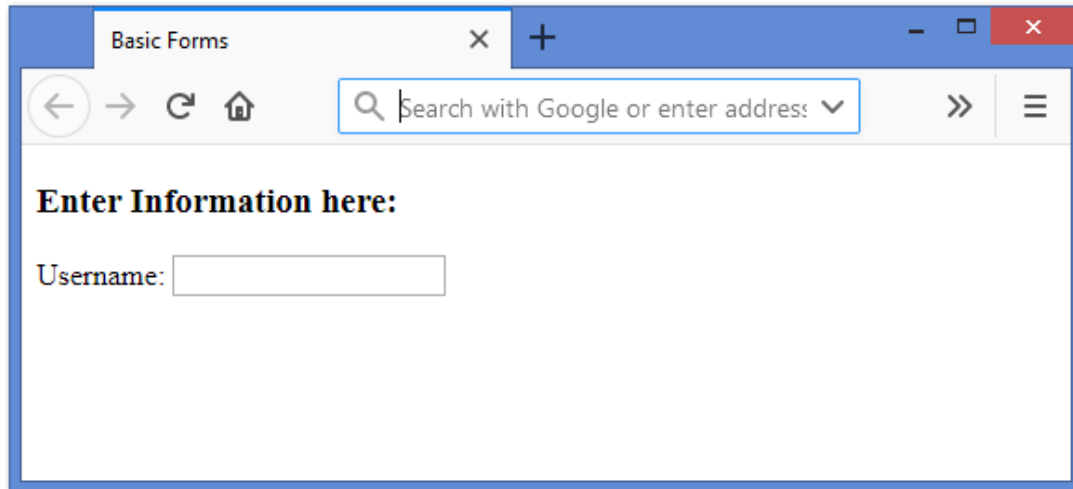
1. Create the following page that has a single input that allows the user to input text:

```
1 <!doctype html>
2 <html lang="en">
3 <head>
4     <title>Basic Forms</title>
5 </head>
6 <body>
7     <input type="text">
8 </body>
9 </html>
```

2. Save your page and view it in a browser. There should be a single box on the where you enter text. Note that the `<input>` tag is self-closing.
3. Amend your page so that you have text on the page above the input box and beside the input box as shown:

```
6 <body>
7     <h3>Enter Information here:</h3>
8     <p>Username: <input type="text"></p>
9 </body>
```

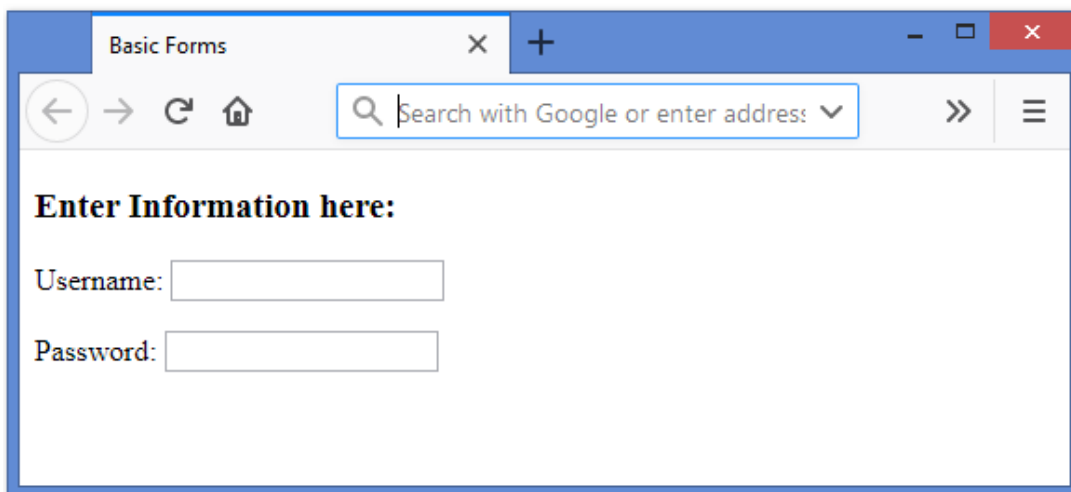
Your finished result should look similar to as shown below:



4. Add a password input using the input tag – this will also use the “type” attribute but will have a value of “password”:

`<p>Password: <input type="password"></p>`

It should also have the text “Password” before the password input box, so that your page should look similar to as shown here:



5. Finally, to add a button, use the Button tag `<button>`. Whatever text appears between the opening and closing button tags is what will appear in the button as text:

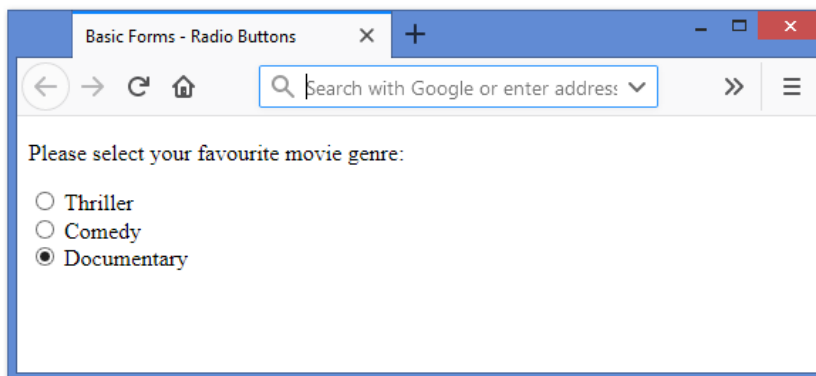
```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <title>Basic Forms</title>
5 </head>
6 <body>
7   <h3>Enter Information here:</h3>
8   <p>Username: <input type="text"></p>
9   <p>Password: <input type="password"></p>
10  <button>Click here to login!</button>
11 </body>
12 </html>
```

Exercise 2: Radio Buttons

1. Create the following page using with radio buttons. Radio buttons are created using the input tag, but the attribute “type” value is “radio”, as shown. Note that the name attribute is important to use in radio buttons. The radio group must have the same name (the value of the “name” attribute) *to be treated as a group*. Once the radio group is created, selecting any radio button in that group automatically deselects any other selected radio button in the same group. You can have as many radio groups on a page as you want, as long as each group has its own name.

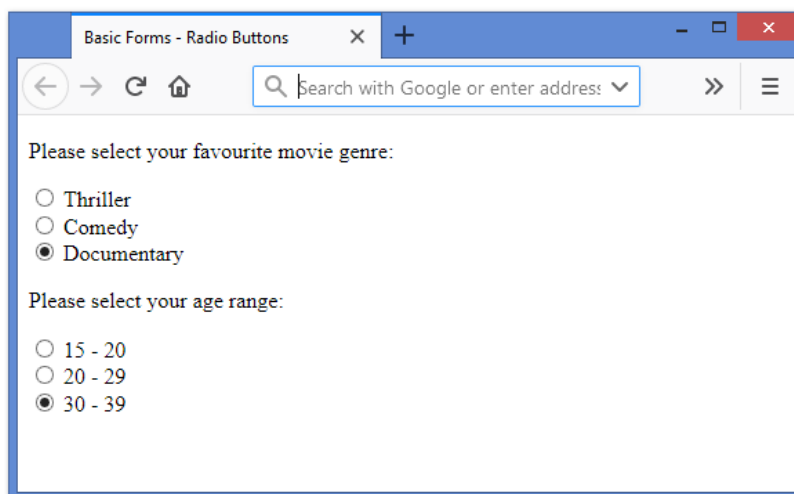
```
1 <!doctype html>
2 <html lang="en">
3 <head>
4   <title>Basic Forms - Radio Buttons</title>
5 </head>
6 <body>
7   <p>Please select your favourite movie genre:</p>
8   <input type="radio" name="genre"> Thriller<br>
9   <input type="radio" name="genre"> Comedy<br>
10  <input type="radio" name="genre"> Documentary<br>
11 </body>
12 </html>
```

Your finished page should look similar to as shown here:



The screenshot shows a web browser window titled "Basic Forms - Radio Buttons". The page content is: "Please select your favourite movie genre:". Below this text are three radio button options: "Thriller", "Comedy", and "Documentary". The "Documentary" option is selected, indicated by a filled circle next to it.

2. Amend your page to include an additional radio button selection option for an age range as shown below:

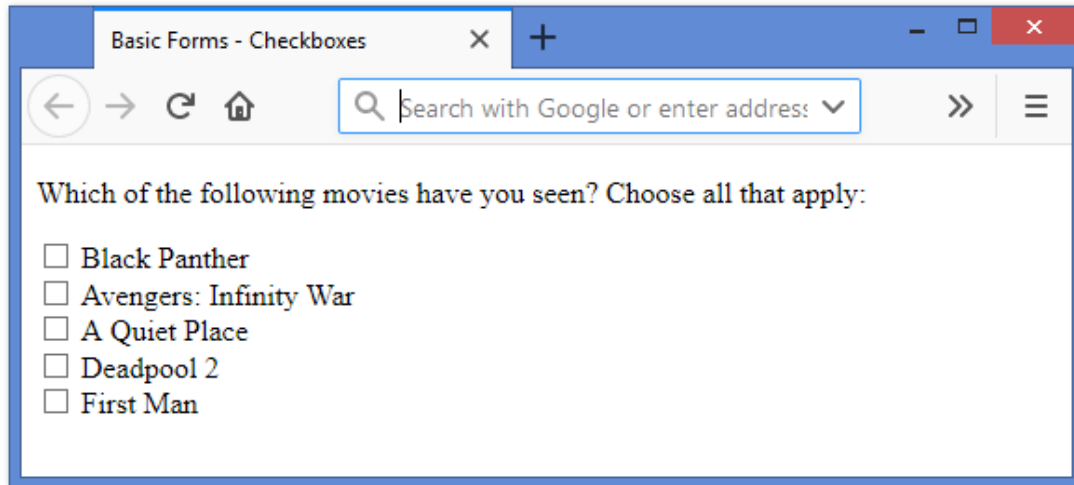


The screenshot shows the same web browser window, but the form has been updated. It now includes a second section: "Please select your age range:". Below this text are three radio button options: "15 - 20", "20 - 29", and "30 - 39". The "30 - 39" option is selected, indicated by a filled circle next to it.

3. Check that page works properly so that it's not possible to select any more than one option in the movie genre set and no more than one option in the age range set. Selecting an option in the genre set should have no effect in the age set.

Exercise 3: Create a checkbox

1. Create web page that contains a checkbox that allows the user select films they have seen. Your finished page should look similar to as shown below. Note that the only attribute required for the checkbox in this exercise is the type attribute.



The screenshot shows a web browser window titled "Basic Forms - Checkboxes". The address bar contains the text "Search with Google or enter address:". The main content area displays the text "Which of the following movies have you seen? Choose all that apply:" followed by a list of five movies, each preceded by an unchecked checkbox:

- ☐ Black Panther
- ☐ Avengers: Infinity War
- ☐ A Quiet Place
- ☐ Deadpool 2
- ☐ First Man

2. Amend your code so that the movie "First Man" is checked by default when the page opens by using the "checked" attribute.

Exercise 4: Multi-input form

You have been asked to create an entry form for the Forcefield Sneakers “Pimp My Shoes” contest. The copy editor had handed you a screenshot of the form’s content. Recreate this form using appropriate form controls:

Want to trade in your old sneakers for a custom pair of Forcefields? Make a case for why your shoes have *got* to go and you may be one of the ten lucky winners

Name:

Email:

Phone no.:

My shoes are so old...

Design your custom Forcefields:

Custom Shoe Design

Color (*choose one*)

☐ Red ☐ Blue ☐ Black ☐ Silver

Features (*Choose as many as you want:*)

- ☐ Sparkly laces
- ☐ Metallic Logo
- ☐ Light-up heels
- ☐ MP3-enabled

Exercise 5: Create a credit card payment form

1. Recreate the following form using appropriate HTML form controls, such as text boxes, textareas, radio buttons and drop-down boxes.

The options for the months should be from January to December but with a generic Month option which is selected by default. The options for the years for the credit card details should be from current year to 5 years from now but with a generic Year option which is selected by default

HINT: To put colour on the entire page, you can set a *bgcolor* attribute on the <body> tag:

First Name:

Last Name:

Address 1:

Address 2:

County:

Credit card types:

☐ VISA

☐ VISA DEBIT

☐ MASTERCARD

☐ AMERICAN EXPRESS

Credit card number

Expiry year:

Expiry month:

Additional Notes

Exercise 6: Form Validation

1. In this exercise the form tags will be used to enforce input validation. Validation can check that users enter the correct type of information into a form and displays a notification if there is an input error.

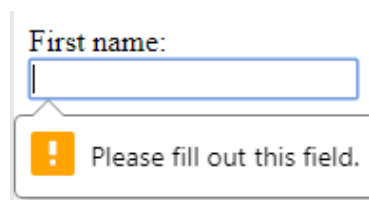
Create a web page using the code shown below:

```
1  <!DOCTYPE html>
2  <html>
3  <head>
4      <title>HTML5 Validation</title>
5  </head>
6  <body>
7      <form>
8          First name: <br/>
9          <input type="text" required /> <br/><br/>
10         <input type="submit" value="Submit" />
11     </form>
12 </body>
13 </html>
```

This page has 2 input types; a text box and a submit button. Note that both of these items are inside a pair of <form> tags, shown opening on line 7 and closing on line 11 above. For validation to work, inputs must be enclosed inside form tags.

In addition to this, the input tag on line 9 is using the “required” attribute. This means that the user is required to provide information here.

When this page is viewed in a browser, if nothing is entered into the input box and the “Submit” button is pressed, the user will be notified to provide information, for example:



Once there is data inside the input box, the submit button can be clicked and the data validated.

NOTE: After clicking the submit button, the page will be refreshed, so it will return to a blank input box – this is expected behaviour.

2. Amend your page so that it is as shown below.

Note:

- Ensure that all input fields are required
- That telephone number input box has a placeholder value set and checks the number input by the user conforms to expected pattern – (HINT: type="tel"). See http://www.w3schools.com/html/html_form_input_types.asp.
- That the email input box requires a properly formatted email address with an "@". (HINT: type="email")

First name:

Surname:

Telephone Number:

Email Address:

Submit

Exercise 7

1. Amend your form from exercise 5 to include images instead of text for the credit cards. Images are available along with this lab on Moodle – see [exercise7_Images](#) file.

First name:

Last name:





Address 1:

Address 2:

County:

Credit Card Information

Credit card type:

☐ 
☐ 
☐ 
☒ 

Credit card number

Expiry year:

Expiry month:

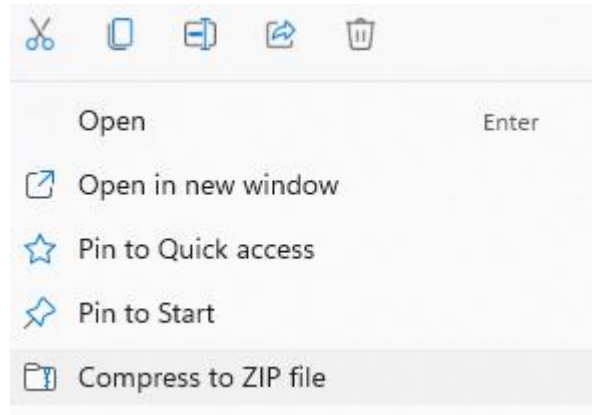
Additional notes:

Enter additional information here...

2. Add a field set and legend to create the border as shown around the “Credit Card Information” section as above. See https://www.w3schools.com/tags/tag_fieldset.asp for example
3. Code your page so that clicking on the appropriate credit card image will select the relevant radio button (HINT: <label>). See https://www.w3schools.com/tags/tryit.asp?filename=tryhtml5_input_type_radio for example.
4. Add form validation so that the user is required to input the appropriate information for **all** areas of the form.

Upload your work to Moodle

1. Navigate to the location of the folder where you saved all your work for today's lab.
2. Right-click on the folder and select "Compress to ZIP file". This will create a compressed version of any files you have worked on for the lab.



3. There should be a new compressed file created. This is the file that you will need to upload to Moodle.
4. In Moodle, navigate to the "Submissions" section, and click the **upload** link for current lab.
5. Click "add submission" and add the ZIP file you created here. Make sure you complete the submission process. Your lab work has been submitted.