# Exercise 1: Creating Web Pages with HTML

## 1 Introduction

Web pages consist of text documents that contain plain text formatted with ***HTML*** (***HyperText Markup Language***) tags embedded within the text. HTML is a computer language used to format the content displayed in Web pages. The formatting consists of configuring the foreground and background color, adding white spaces between text, aligning text, configuring font, creating lists, tables, and forms. In this exercise, we will learn how to use HTML to format plain text into Web pages.

### 1.1 Topics

- Creating Web content with the HyperText Markup Language (HTML)
- Formatting Web content with HTML tags
- Interacting with Web pages with HTML form tags
- Navigating between Web pages with HTML anchor tags

## 2 Exercises

This section walks you through several exercises to familiarize yourself with HTML. Copy the examples into an HTML document as instructed and confirm that they render as intended. After practicing with the exercises, you will be asked to apply the skills to create ***Tuiter***, a Web application inspired by a popular social networking site.

### 2.1 Heading Tags

Text documents are often broken up into several sections and subsections. Each section is usually prefaced with a short title or ***heading*** that attempts to summarize the topic of the section it precedes. For instance, this paragraph is preceded by the heading ***Heading Tags***. The font of the section headings is usually larger and bolder than their subsection headings. This document uses headings to introduce topics such as HTML

Documents, HTML Tags, Heading Tags, etc. HTML ***heading tags*** can be used to format plain text so that it renders in a browser as large headings. There are 6 heading tags: ***h1***, ***h2***, ***h3***, ***h4***, ***h5***, and ***h6***. Tag ***h1*** is the largest heading and ***h6*** is the smallest heading.

To practice using the heading tags we are going to create several headings and subheadings to introduce the topics we will cover in this exercise. Under the ***public*** directory, create directory ***exercises/e1*** where you will practice several HTML exercises for exercise 1. Under the ***exercises/e1*** directory, create an HTML file called ***index.html***. Copy the HTML below into the ***<body>*** tag of this new file.

Then, after the ***Heading Tags*** heading (highlighted in red here on the right), copy and paste the first paragraph of this section highlighted in yellow. To see the content of the Webpage, open it in the browser.

| *index.html* | How the browser renders |
|---|---|
| `<h1>HTML Examples</h1>`<br><br>`<h2>Heading Tags</h2>` | **HTML Examples**<br>**Heading Tags** |

The file will open in a browser window and the content should look similar to the content highlighted yellow at the beginning of this section. Note how the text surrounded by the ***<h1>*** tag is larger and bolder than the text surrounded by the ***<h2>*** tag, and both are larger than the text that has no tags around it. The ***index.html*** file should now look as shown below.

*index.html*

```
<!DOCTYPE html>
<html lang="en">

<head>
   <meta charset="UTF-8">
   <title>Title</title>
</head>

<body>
   <h1>HTML Examples</h1>
```

```
   <h2>Heading Tags</h2>

   Text documents are often broken up into several sections and subsections. Each section is
usually prefaced with a short title or heading that attempts to summarize the topic of the
section it precedes. For instance, this paragraph is preceded by the heading Heading Tags. The
font of the section headings is usually larger and bolder than their subsection headings. This
document uses headings to introduce topics such as HTML Documents, HTML Tags, Heading Tags, etc.
HTML heading tags can be used to format plain text so that it renders in a browser as large
headings. There are 6 heading tags: h1, h2, h3, h4, h5, and h6. Tag h1 is the largest heading
and h6 is the smallest heading.

</body>
</html>
```

## 2.2 Paragraph Tag

Browsers ignore white spaces such as tabs and newlines. To add space between different paragraphs we can use the paragraph tag ***<p>***. Wrap text with the paragraph tag to add vertical spacing. To practice using the paragraph tag, copy the code on the right at the end of the ***index.html***, but still within the ***<body>*** tag. Below is another example of how the browser renders HTML text on the left column. Note how the browser ignores line breaks and other white space formatting like tabs and content just flows from left to right and then wraps when there's no more horizontal space. This style of rendering is referred to as ***inline***. Inline content flows from left to right horizontally the whole width of its parent container and then wraps vertically when there's no more space.

***index.html***

```
<h2>Paragraph Tag</h2>
<p>
   This is a paragraph. We often separate a long
set of sentences with vertical spaces to make
the text easier to read. Browsers ignore
vertical white spaces and render all the text as
one single set of sentences. To force the
browser to add vertical spacing, wrap the
paragraphs you want to separate with the
paragraph tag
</p>
```

| *index.html* | How the browser renders |
| --- | --- |
| This is the first paragraph. The paragraph tag is used to format vertical gaps between long pieces of text like this one.<br><br>This is the second paragraph. Even though we added a deliberate gap between the paragraph above and this paragraph, by default browsers render them as one contiguous piece of text as shown here on the right.<br><br>This is the third paragraph. Wrap each paragraph with the paragraph tag to tell browsers to render the gaps. | This is the first paragraph. The paragraph tag is used to format vertical gaps between long pieces of text like this one. This is the second paragraph. Even though we added a deliberate gap between the paragraph above and this paragraph, by default browsers render them as one contiguous piece of text as shown here on the right. This is the third paragraph. Wrap each paragraph with the paragraph tag to tell browsers to render the gaps. |

Applying the paragraph tags below lets the browser know we want to keep the vertical spacing.

| *index.html* | How the browser renders |
| --- | --- |
| <br>This is the first paragraph. The paragraph tag is used to format vertical gaps between long pieces of text like this one.<br> <br> <br>This is the second paragraph. Even though there is a deliberate white gap between the paragraph above and this paragraph, by default browsers render them as one contiguous piece of text as shown here on the right.<br> <br> <br>This is the third paragraph. Wrap each paragraph with the paragraph tag to tell browsers to render the gaps.<br> | This is the first paragraph. The paragraph tag is used to format vertical gaps between long pieces of text like this one.<br><br>This is the second paragraph. Even though there is a deliberate white gap between the paragraph above and this paragraph, by default browsers render them as one contiguous piece of text as shown here on the right.<br><br>This is the third paragraph. Wrap each paragraph with the paragraph tag to tell browsers to render the gaps. |

Copy the HTML above on the left to the end of the ***index.html*** document in the ***Paragraph Tag*** section. Remember to keep all your content within the ***body*** tag. Refresh the Webpage and confirm it renders as shown on the right. Note how the paragraphs are now spaced vertically from one another. Both the paragraph and heading tags add vertical space and we refer to this style of rendering as ***block***. By controlling the inline and block styles of laying out content, we can achieve all sorts of useful layouts.

## 2.3 List Tags

List tags are used to create lists of related items. There are two types of lists: ***ordered*** and ***unordered***.

### 2.3.1 Ordered List Tag

Ordered list tags are useful for listing items in a particular order. Here's a list of steps to make a cup of tea.

| *index.html* | How the browser renders |
| --- | --- |
| `<h2>List Tags</h2>`<br>`<h3>Ordered List Tag</h3>`<br>`How to make a cup of tea?`<br>`1. Boil water.`<br>`2. Place a tea bag or tea leaves in a cup.`<br>`3. Pour hot water over the tea.`<br>`4. After 3 minutes, remove the tea bag or strain loose leaves.`<br>`5. Add milk, sugar, honey, or lemon if desired.`<br>`6. Enjoy your tea!` | **List Tags**<br>**Ordered List Tag**<br>How to make a cup of tea?1. Boil water.2. Place a tea bag or tea leaves in a cup.3. Pour hot water over the tea.4. After 3 minutes, remove the tea bag or strain loose leaves.5. Add milk, sugar, honey, or lemon if desired.6. Enjoy your tea! |

Note that in the HTML text on the left we explicitly wrote the numbers 1., 2., etc., but the nice formatting is lost when the browser renders it on the right. Instead of rendering a list of items, each in its own line, they are instead all rendered on the same line. To achieve the desired format, we'll use the ordered list tag. The ordered list tag actually consists of a pair of tags

- <ol> declares the beginning of the list
- <li> declares an item in the list

Here's the same example from earlier, but now applying the ordered list tags to achieve the intended formatting.

| *index.html* | How the browser renders |
|---|---|
| `<h2>List Tags</h2>`<br>`<h3>Ordered List Tag</h3>`<br>`How to make a cup of tea?`<br>`<ol>`<br>`  <li>Boil water.</li>`<br>`  <li>Place a tea bag or tea leaves in a cup.</li>`<br>`  <li>Pour hot water over the tea.</li>`<br>`  <li>After 3 minutes, remove the tea bag or strain loose leaves.</li>`<br>`  <li>Add milk, sugar, honey, or lemon if desired.</li>`<br>`  <li> Enjoy your tea!</li>`<br>`</ol>` | List Tags<br><br>Ordered List Tag<br><br>How to make a cup of tea?<br><br>1. Boil water.<br>2. Place a tea bag or tea leaves in a cup.<br>3. Pour hot water over the tea.<br>4. After 3 minutes, remove the tea bag or strain loose leaves.<br>5. Add milk, sugar, honey, or lemon if desired.<br>6. Enjoy your tea! |

Copy the HTML above to the end of ***index.html*** file and confirm it renders as shown on the right.

## 2.3.2 Unordered List Tag

The unordered list tag is similar to its ordered version with the difference that the items are not numbered and instead bullets decorate each line item. The unordered list tag is ***<ul>***, but the list item tag is still ***<li>*** as shown below. Unordered lists are great for displaying a list of items in no particular order. Here's an example of an unordered list of my favourite books in no particular order.

| *index.html* | How the browser renders |
| --- | --- |
| `<h3>Unordered List Tag</h3>`<br>`My favourite books (in no particular order)`<br>`<ul>`<br>`  <li>Dune</li>`<br>`  <li>Lord of the Rings</li>`<br>`  <li>Ender's Game</li>`<br>`  <li>Red Mars</li>`<br>`  <li>The Forever War</li>`<br>`</ul>` | **Unordered List Tag**<br>My favourite books (in no particular order)<br>• Dune<br>• Lord of the Rings<br>• Ender's Game<br>• Red Mars<br>• The Forever War |

## 2.4 Table Tags

HTML began as a tool for sharing research results between scientists. These documents often consisted of data points captured as a result of some experiment. Each data point might have several attributes associated. An effective way to display or visualize these results were formatted in a data table with a row for each data point and a column for each attribute. The ***<table>*** tag allows formatting data into a table with rows and columns. For instance, consider capturing grade results for several quiz exams you might have taken over the semester. These might be captured using the following table.

| Quiz | Topic | Date | Grade |
| --- | --- | --- | --- |
| Q1 | HTML | 12/09/23 | 85 |
| Q2 | CSS | 02/10/23 | 90 |
| Q3 | JavaScript | 21/11/23 | 95 |
| Average | | | 90 |

Several things to note:

1. The first row is formatted as headings for each column
2. There are 3 data points, one for each quiz, one in each row
3. Each data point has the same data types for each of the columns, e.g, Quiz, Topic, Date, Grade
4. The last row is formatted as a footer
5. The three first columns of the last row are merged into a single cell and unlike the 3 data rows

HTML ***table*** tag can be used to format the data with the following tags:

- ***table*** - declares the start of a table
- ***tr*** - declares the start of a row
- ***td*** - declares a table data cell
- ***thead*** - declares a row of headings
- ***tbody*** - declares the main data content rows of the table
- ***tfoot*** - declares a row as a footer
- ***th*** - declares a table cell as a heading

To practice using ***table*** tag, copy the HTML below to the end of index.html. The code implements the table shown earlier. You can ignore the comments on the right.

```
<h2>Table Tag</h2>
<table border="1" width="100%">      <!-- declares the table, sets border and width -->
   <thead>                           <!-- declares the table heading section -->
   <tr>                              <!-- declares the headings row -->
       <th>Quiz</th>                 <!-- declares heading for first column -->
       <th>Topic</th>                <!-- declares heading for second column -->
       <th>Date</th>                 <!-- declares heading for third column -->
       <th>Grade</th>                <!-- declares heading for fourth column -->
   </tr>
   </thead>
   <tbody>                           <!-- declares the table's main content -->
   <tr>                              <!-- declares the first row -->
       <td>Q1</td>                   <!-- declares data for first row, first column -->
       <td>HTML</td>                 <!-- declares data for first row, second column -->
```

```
        <td>12/09/23</td>              <!-- declares data for first row, third column -->
        <td>85</td>                    <!-- declares data for first row, fourth column -->
    </tr>
    <tr>                               <!-- declares the second row -->
        <td>Q2</td>                    <!-- declares data for second row, first column -->
        <td>CSS</td>                   <!-- declares data for second row, second column -->
        <td>02/10/23</td>              <!-- declares data for second row, third column -->
        <td>90</td>                    <!-- declares data for second row, fourth column -->
    </tr>
    <tr>                               <!-- declares the third row -->
        <td>Q3</td>                    <!-- declares data for third row, first column -->
        <td>JavaScript</td>            <!-- declares data for third row, second column -->
        <td>21/11/23</td>              <!-- declares data for third row, third column -->
        <td>95</td>                    <!-- declares data for third row, fourth column -->
    </tr>
    </tbody>
    <tfoot>
    <tr>                               <!-- declares the table footer section -->
        <td colspan="3">Average</td>   <!-- declares the footer row -->
        <td>90</td>                    <!-- declares data spans 3 columns -->
    </tr>                              <!-- declares data for fourth column -->
    </tfoot>
</table>
```

## 2.5 Image Tag

Use the image tag to render pictures in your HTML documents. The images can be anywhere on the internet, or a local image document in your local file system.

```
<img                        <!-- Use img tag to embed pictures in HTML documents.
src="chandrayan.jfif"       The src attributes references the image file either locally
    width="200px"           or remotely. The width and height attributes configure the image size. If only width or height is
    height="300px"/>        provided, the other scales proportionally    -->
```

To practice using the image tag, copy the code below at the end of ***index.html***. The first image tag embeds an image from a remote server. The second one assumes you have a local image file called ***teslabot.jfif***. Search for Tesla Bot on the internet, and download an image that looks similar to the one shown below. Name the image ***teslabot*** keeping the original file extension.

```
<h2>Image tag</h2>

Loading an image from the internet:<br/>

<img width="400px"
    src="https://www.staradvertiser.com/wp-
content/uploads/2021/08/web1_Starship-gap2.jpg"/>
<br/>

Loading a local image:<br/>

<img src="teslabot. jfif "
    height="200px"/>
```

## 2.6 Form Tags

Form tags are useful for entering data. Let's take a look at the most common ones: ***form***, ***input***, ***select***, ***textarea***, ***radio***, ***checkbox***.

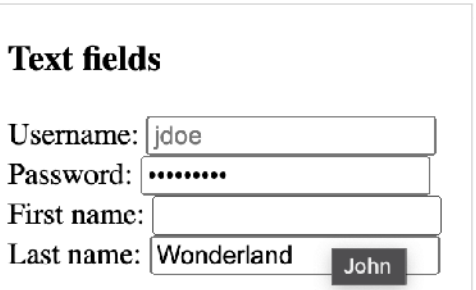
### 2.6.1 Text fields

Text fields are the most common form elements allowing entering a single line of text.

```
<input type="text"         <!--  use input tag's text type to declare a single line input field text is default if type is
       placeholder="hint"  left out. Use placeholder and title to give a hint of what information you're expecting.
       title="tooltip"     Optionally initialize the value of the field with value attribute-->
       value="COMEDY"/>
```

To practice using text fields, add the following example at the end of ***index.html***. It creates a set of input fields for entering some personal information. The ***label*** tags below associate descriptive text with each form element. The is established by setting a ***label***'s ***for*** attribute to the ***id*** attribute of the related form field.

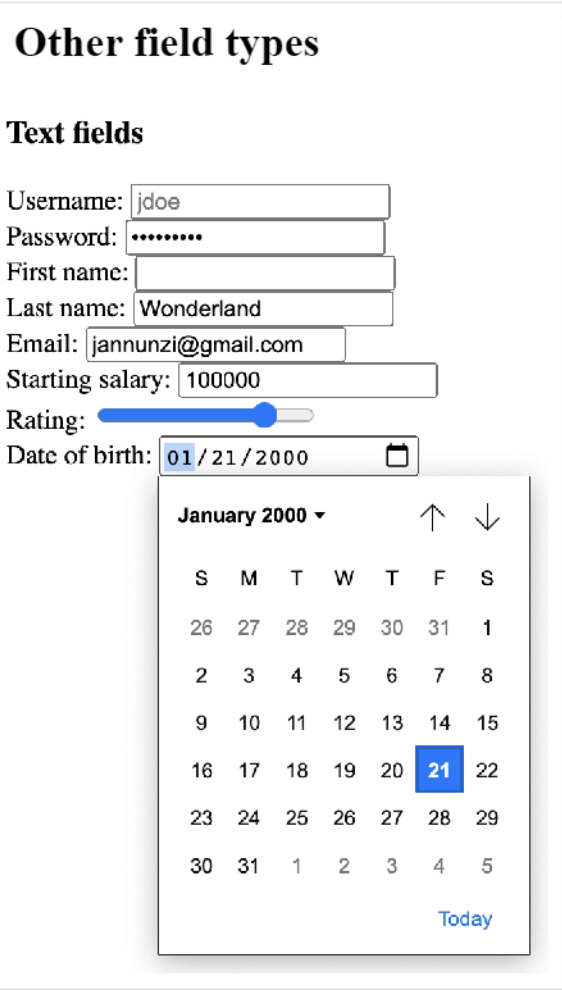
```
<h2>Text fields</h2>
<form id="text-fields">
   <label for="text-fields-username">Username:</label>
   <input id="text-fields-username" placeholder="jdoe"/><br/>
   <label for="text-fields-password">Password:</label>
   <input type="password" id="text-fields-password"
          value="123@#$asd"/><br/>
   <label for="text-fields-first-name">First name:</label>
   <input type="text" id="text-fields-first-name"
          title="John"/><br/>
   <label for="text-fields-last-name">Last name:</label>
   <input type="text" id="text-fields-last-name"
          placeholder="Doe" value="Wonderland"/>
   <!-- copy rest of form elements here -->
</form>
```



## 2.6.2 Date, email, number, and range fields

The input tag's ***type*** attribute has several other possible values: ***date***, ***email***, ***number***, and ***range***. They can be used to enter text information with a specific format. To practice these other formats add the following example under the last input field you worked on earlier, but inside the ***form*** tag. The fields should look as shown below on the right.

```
<h2>Other field types</h2>
<label for="text-fields-email">
   Email:
</label>
<input type="email"
       placeholder="jdoe@somewhere.com"
       id="text-fields-email"/><br/>
<label for="text-fields-salary-start">
   Starting salary:
</label>
<input type="number"
       id="text-fields-salary-start"
       placeholder="1000"
       value="100000"/><br/>
<label for="text-fields-rating">
   Rating:
</label>
<input type="range" id="text-fields-rating"
       placeholder="Doe"
       max="5"
       value="4"/><br/>
<label for="text-fields-dob">
   Date of birth:
</label>
<input type="date"
       id="text-fields-dob"
       value="2000-01-21"/><br/>
```



## 2.6.3 Text boxes

The ***textarea*** tag is useful for entering long form text such as someone's biography data, or a blog post.

```
<textarea cols="20"              <!--  use textarea tag for long form text configure its width and
        rows="25"                height with attributes cols and rows. Use placeholder and tooltip to
        placeholder="Biography"  give hints. Note default value is in tag's body -->
        title="tooltip">Some
text
</textarea>
```

To practice using the ***textarea*** tag, add the following example to the end of index.html. It creates a ***textarea*** useful for entering your biography. You can get a sample of the dummy text at https://www.lipsum.com/.

```
<h2>Text boxes</h2>
<form id="textarea">
   <label>Biography:</label><br/>
   <textarea cols="30" rows="10">Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed do eiusmod tempor incididunt ut
labore et dolore magna aliqua. Ut enim ad minim veniam, quis nostrud
exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat.
Duis aute irure dolor in reprehenderit in voluptate velit esse cillum
dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat
non proident, sunt in culpa qui officia deserunt mollit anim id est
laborum.</textarea>
</form>
```

## 2.6.4 Buttons

Buttons allow invoking actions executed by the browser. To practice creating buttons, copy the code below at the end of ***index.html***.

```
<h3>Buttons</h3>
<button type="button">Click me!</button>
```

## 2.6.5 Dropdowns

Dropdowns are useful for selecting one or more options from a list of possible values. The default version displays a set of values from which you can choose a single value.

```
<select>                    <!--  Wrap several option tags in a select tag. Optionally provide option's value,
   <option                  otherwise the option's text is the value of the select element. Optionally use selected
value="VAL1">Value          attribute to select default.  -->
1</option>
   <option value="VAL2"
selected>
       Value 2</option>
   <option
value="VAL3">Value
3</option>
</select>
```

Adding the optional ***multiple*** attribute converts the dropdown into a list of options that can be selected.

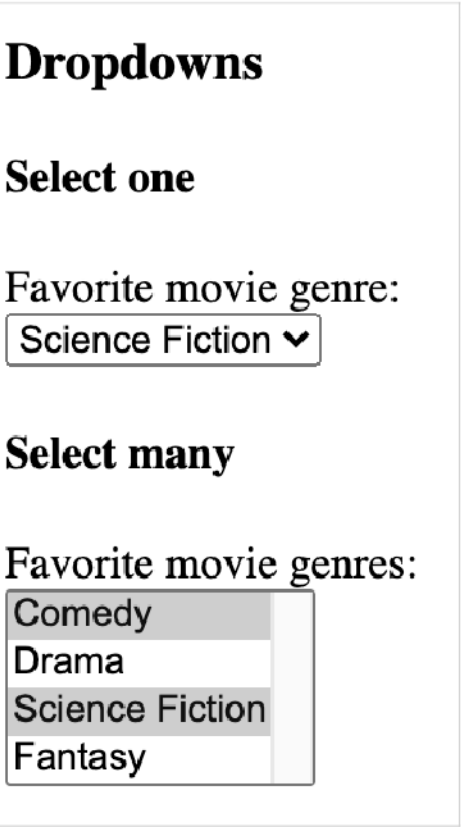
```
<select multiple>                   <!--  Alternatively use attribute multiple to allow selecting more than one
   <option value="VAL1"             option.     Use ctrl+click to select more than one option -->
selected>Value 1</option>
   <option value="VAL2">Value
2</option>
   <option value="VAL3"
selected>Value 3</option>
</select>
```

To practice using the ***select*** tag, add the following example to the end of ***index.html***. It creates a dropdown and a list of options.

```
<h2>Dropdowns</h2>

<h3>Select one</h3>
<label for="select-one-genre">
   Favorite movie genre:
</label><br/>
<select id="select-one-genre">
   <option value="COMEDY">Comedy</option>
   <option value="DRAMA">Drama</option>
   <option selected value="SCIFI">
       Science Fiction</option>
   <option value="FANTASY">Fantasy</option>
</select>

<h3>Select many</h3>
<label for="select-many-genre">
   Favorite movie genres:
</label><br/>
<select id="select-many-genre" multiple>
   <option selected value="COMEDY">Comedy</option>
   <option value="DRAMA">Drama</option>
   <option selected value="SCIFI">
       Science Fiction</option>
   <option value="FANTASY">Fantasy</option>
</select>
```




## 2.6.6 File upload button

Use the file type for the input tag to choose a file for upload. We won't be able to upload just yet until later in the course, but for now let's practice adding a file upload tag as shown below. Clicking the button pops up a file choose where you can navigate to the file you want to upload. To practice using the file selector, copy the code below to the end of ***index.html***. We'll learn how to upload files later in the course.

```
<h2>File upload</h2>
<input type="file"/>
```

## 2.6.7 Radio buttons

Radio buttons allow selecting a single choice from multiple alternative options

```
<input type="radio"
       name="NAME1"
       value="OPTION1"/>
<input type="radio"
checked
       name="NAME1"
       value="OPTION2"/>
```

```
<!--  use the input tag's checkbox type to declare a checkbox give the checkbox a value
      -->
```

To practice using radio buttons, add the following example at the end of ***index.html***.

```
<h2>Radio buttons</h2>

<label>Favorite movie genre:</label><br/>

<input type="radio" value="COMEDY"
      name="radio-genre" id="radio-comedy"/>
<label for="radio-comedy">Comedy</label><br/>
<input type="radio" value="DRAMA"
      name="radio-genre" id="radio-drama"/>
<label for="radio-drama">Drama</label><br/>
<input type="radio" value="SCIFI"
      name="radio-genre" id="radio-scifi" checked/>
<label for="radio-scifi">Science Fiction</label><br/>
<input type="radio" value="FANTASY"
      name="radio-genre" id="radio-fantasy"/>
<label for="radio-fantasy">Fantasy</label>
```

## 2.6.8 Checkboxes

Checkboxes allow selecting multiple choices

```
<input type="checkbox"    <!--  use the input tag's checkbox type to declare a checkbox give the checkbox a value
       name="NAME2"             -->
       checked
       value="OPTION1"/>
<input type="checkbox"
       name="NAME2"
       value="OPTION2"/>
<input type="checkbox"
       checked
       name="NAME2"
       value="OPTION3"/>
```

To practice using checkboxes, add the following example to the end of ***index.html***. It creates a set of checkbox buttons to select all your favorite movie genres, which there might be more than one.

```
<h2>Checkboxes</h2>
<label>Favorite movie genre:</label>
<br/>
<input type="checkbox" value="COMEDY"
      name="check-genre" id="chkbox-comedy" checked/>
<label for="chkbox-comedy">Comedy</label>    <br/>
<input type="checkbox" value="DRAMA"
      name="check-genre" id="chkbox-drama"/>
<label for="chkbox-drama">Drama</label>    <br/>
<input type="checkbox" value="SCIFI"
      name="check-genre" id="chkbox-scifi" checked/>
<label for="chkbox-scifi">Science Fiction</label> <br/>
<input type="checkbox" value="FANTASY"
      name="check-genre" id="chkbox-fantasy"/>
<label for="chkbox-fantasy">Fantasy</label>
```

## 2.7 Anchor Tag

The anchor tag allows navigating to other websites or other pages within the same website.

```
<a href="aa.com">
American
Airlines</a>
```

```
<!--  Use the href attribute to refer to the location of the website or other page in the same
website. Click on the body text to navigate -->
```

To practice using anchor tags, add the following example at the end of ***index.html***. It creates two hyperlinks. One navigates to ***lipsum.com***, a website that contains dummy text, and the other link navigates to another document located in the same website. Create the ***other-page.html*** document in the same directory as ***index.html*** and fill it with some dummy text. Click the ***other page*** link and confirm the navigation works.

```
<h2>Anchor tag</h2>

Please
<a href="https://www.lipsum.com">click here</a>
to get dummy text<br/>

Checkout my <a href="other-page.html">other page</a>
```

**Anchor tag**

Please click here to get dummy text
Checkout my other page