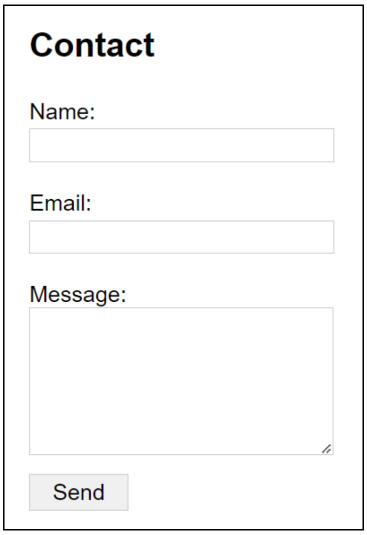
HTML/CSS

Basic Online Form, Usability and Accessibility



Contact

Name:

Email:

Message:

Preparation

1 Copy the **basic online form** folder to your disk space

2 Open **contact.html** (you will link to the css file later in this task)

Structural coding

1 Place the cursor below the <**main**> tag, insert a <**section**> tag with an **id** of **contact-form** then insert a closing tag, eg </section>

2 Place the cursor between these <**section**> tags and insert the following **<form>** coding

```
<form action="contact.php" method="post">
</form>
```

Note 1: the action attribute typically has a php submission process, eg contact.php

Note 2: the method attribute can be post or get – post is more secure

Basic form elements

Input for Text – end-user's name

1 Start with a **label** and **input** code for the end-user to type their name into the form

2 Place the cursor between the <**form**> tags, insert the following <**label**> and <**input**> coding

```
<label for="name" >Name:</label>
<input type="text" id="name" name="name" maxlength="30" required>
```

Note 1: the <input> tag is self-closing – it is similar to the <img> tag

Note 2: the value of **for** and **id** are the same. They provide accessibility options by creating an association between the <label> and the <input> codes

Note 3: the **name** attribute and value are **required** for the data to be submitted to the database – the data will not be sent to the database if you omit this coding!!!

Note 4: the **maxlength** attribute is strongly recommended as it limits the amount of data that is submitted to the database – otherwise an end-user could copy and paste many pages of text with dire consequences for the website – DoS etc

Note 5: the **placeholder** attribute and value is optional – it displays instructions inside the input box **however** this text is replaced when the end-user types data into the input box – IS THIS USER-FRIENDLY?

Input for Text – end-user's email address

1 Copy and paste the <**label**> and <**input**> coding from the previous step

2 Amend the attribute values to suit

```
<label for="email" >Email:</label>

<input type="text" id="email" name="email" maxlength="40">
```

Textarea – for end-user's message

1 Place the cursor below the <**input**> tag for the email address

2 Insert the following <**label**> and <**textarea**> coding

```
<label for="message" >Message:</label>

<textarea id="message" name="message" maxlength="1500"></textarea>
```

Note 1: the <textarea> tag requires a closing tag

Note 2: the maxlength attribute for <textarea> is new in HTML5

Input for Submit Button

1 Place the cursor below the **</textarea>** tag, insert coding for the submit button

```
<input type="submit" value="Send">
```

Note 1: the **name** attribute and value are **not required** for this code because no data is attached to this coding

Note 2: the **value** attribute changes the default button text from **"**Submit**"** to your choice of text, eg value=**"**Send**"**

2 Ensure the </form> tag is located after the submit button

3 Ensure the </section> and </main> tags are located after the </form> tag

Apply CSS

- link the css file
- the form elements should be displayed vertically – apply display: flex and flex-direction: column to **form** then add a top margin

- apply a suitable width to the form
- add a top margin to label
- the border is missing from the input boxes – apply a suitable border
- the border is missing from the textarea element – apply a suitable border
- the width of the textarea box doesn't match the input elements. Apply a width to the textarea box to match the input boxes then use the handle on the bottom right to expand and contract the textarea size – amend the textarea width to a **min-width** and **add a max-width** of the same value OR apply no resize options (https://www.w3schools.com/howto/howto_css_disable_resize_textarea.asp)
- apply a suitable height to the textarea element – this is one of the rare instances where you will apply a height to an html element
- apply a suitable font hierarchy to body
- enter some text into each form input/textarea element and note that the text in the textarea is a different font and a different size – to correct this: add a font family hierarchy to the textarea rule then add a font size of 16 px. Refresh the browser, add text again to confirm the text entries are displayed in the same font and font size
- apply suitable padding to the input and textarea elements
- target the submit button with an id OR input [type="submit"] { } – apply padding/margin rules
- apply a top margin and a left margin to main – to move the form in from the browser edges
- apply/adjust css rules for presentation – widths, margins, padding, colour, etc

Finalise the html code for the form

1 Place the cursor before the **<section>** tag, insert a major heading **Contact** and apply suitable css for font size, etc

2 Research the **autofocus** and **required** attributes

– the **autofocus** attribute can only be applied to one element in the form. Why?

____

– identify at least two (2) benefits the **autofocus** and **required** attributes provide

____

____

3 Consider placing an indicator for the "required" information that is necessary in the online form

– identify at least two (2) benefits for the end-user

____

____

4 **Validate at W3C** and amend your coding if required