Assignment: Dating Web Site Assessment

Submit by December 13, 11:59 PM PT

Ready to start your next assignment?
You'll find instructions and a place to submit below.

Instructions (/learn/html-css-javascript/peer/biy2Z/dating-web-site-assessment)

My submission (/learn/html-css-javascript/peer/biy2Z/dating-web-site-assessment/submit)

Discussions (/learn/html-css-javascript/peer/biy2Z/dating-web-site-assessment/discussions)
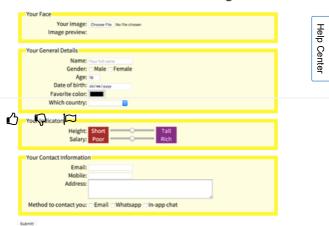
Instructions

By making a web page for a startup dating company, you will gain experience in HTML and CSS.

Instructions	less ∧

The Task

A startup company wishes to build a web site for their online dating business. They have employed a designer who has produced the following design using desktop publishing software. The software used by the designer only produces an image, and does not produce HTML.

The company has now contracted you to implement the design i.e. build the web page. This includes a working *form* component. The company thinks the designer has done a great job and wants you to implement the design without altering the style.

Technical Overview

From a technical point of view, the web page is constructed using HTML and CSS. Most of the HTML content are form elements. All the style parameters are handled in a style section, at the top of the same page.

Interactive Features

The company requires you to have two interactive features in the web page.

1. When the mouse is moved over any *label* or *input* or *select* element, the text in that element is shown larger. When the mouse moves away from the element, the text returns to its normal size.

2. Immediately after the user selects the image of his/her face, the image is shown in the web page. A small piece of JavaScript will be made available to you to help with this functionality. However, you are not expected to understand the JavaScript code at this stage in the course and should not create any further JavaScript for this assessment task.

Discussion Video

Please watch the accompanying video, which shows and discusses the completed task.

What to Submit

There are 3 parts of this assessment. You will need to submit 3 files in total, one for each part. Each subsequent part builds upon the previous part.

For Part 1, you need to create the web page content without style. Make sure you save a copy of this as, for example, part1.html, before you move on to the next part.

For Part 2, take your part 1 file and extend it by applying style. Make sure you save a copy of this as, for example, part2.html, before you move on to the next part.

For Part 3, take your part 2 file and extend it to include the interactive features. Save this as part3.html.

For Parts 2 and 3, please note that all style rules are to be included in the style section, and there should be no use of inline style.

Here are the requirements and further information for each of the 3 parts.

Further Notes

You can only submit one single HTML file. Note that all style rules are to be included in the *style* section, and there is no use of inline style for this assessment.

Part 1: Web page content without style

- For part 1 of this assessment task, you need to assemble the components together in a web page (without style).The result of part 1 will look like this when viewed in a browser.

The result of this stage is a web page which contains a form. That is, after the various form elements are selected/data is entered, the submit button can be pressed. Although it is not an absolute requirement of this assessment task, it would be good if your form is sent to an appropriate server program after the submit button is selected. For this to work the data needs to be sent to a server using the post method with enctype="multipart/form-data". The destination program used in the instructional videos is http://ihome.ust.hk/~rossiter/cgi-bin/show_everything.php (http://ihome.ust.hk/~rossiter/cgi-bin/show_everything.php) .

Here is a summary of the elements needed inside the web page.

At the top

- An appropriate large heading e.g. `h1` or `h2` or `h3`

The HTML elements to be contained within the form are listed by the company as follows.

First fieldset, with legend 'Your face'

- `input type="file"` [required]
- An `img` element with `id="preview"`

Second fieldset, with legend 'Your general details'

- `input type= "text"` ... [required]
- `input type="radio"` ... (two of these) [required]
- `input type="number"` ... [required]
- `input type="date"` ...
- `input type="color"` ...
- `select` , with 6 `option` , to handle 'no selection' plus 5 different countries

Third fieldset, with legend 'Your indicators'

- `input type="range" min="0" max="100"` ...
- `input type="range" min="0" max="100"` ...

Fourth fieldset, with legend 'Your Contact Information'

- `input type="email"` ... [required]
- `input type="tel"` ...
- `textarea` ...
- `input type="checkbox"` ... , three of these are needed for the three types of contact method

At the end (not in any fieldset):

- `input type="submit"`

Important notes

- Each of the four `fieldset` must have a `legend`, as indicated above.
- [required] shown in the list above means that the input must be entered by the user before the form can be successfully submitted, so you must indicate this by adding the appropriate text in the HTML i.e. `required`
- For everything in the form except the submit button, there must be an appropriate `label` which has an appropriate `for` e.g.

```
<label for="avatar">Your image:</label>
<input type="file" id="avatar" name="avatar" required>
```

- You can see the text for each label from the designer's image provided to you. Alternatively, you can use your own text, as long as it hasthe same meaning.
- Every element which is used to provide data that gets sent to the server e.g. the *input* and *textarea* elements, need to have a `name`. They are sent to the server when the submit button is clicked. You are welcome to choose appropriate names.
- Where appropriate, add a break after each input i.e. `<br>` so the web page does not look cluttered.
- Some of the input types such as `input type="email"` have not been introduced in previous instructional videos, but the meaning of this type of input is obvious.
- You may find one or two of the form elements are not supported in certain browsers. You are recommended to use the *Chrome* browser.

For Part 2: Applying visual style

- For this part, style rules are applied appropriately so that the web page has the visual style required by the company. The result of part 2 will look like this when viewed in a browser.

Here are important notes from the designer passed to you by the company. You must use this information in your work.
- The background of each *fieldset* is **lightyellow** e.g. **background:lightyellow**.
- The border of each *fieldset* is **yellow**, with **10px** width e.g. **border:10px solid yellow**
- The *fieldset* needs to have **margin-bottom:10px**
- The width of each *fieldset* is **720px**
- The width of each *label* is **180px**
- The *label* text needs to have **display:inline-block**
- The *label* text needs to be **right-align** and **top-align**
- The *labels* of the checkbox and the radio needs to have **width:auto**
- For the words on both sides of the two *range*, you can use any appropriate style e.g. **text-align:center**
- The *textarea* needs to be **width:360px** and **height:50px**

You can use *class* wherever you think it is appropriate. However, remember that there is no use of inline styles anywhere in this assessment. So that means you cannot do something like this:

```
<input type="email" style="background:pink" ...
```

Instead, all style must be appropriately handled in the *style* section.

For Part 3: Adding interactive features

For this part, two types of interactive behavior need to be added. These interactive behaviors are demonstrated in the accompanying video.

- Style rules are added so that the label and input elements become significantly larger when the mouse is moved over them (and return to normal size when the mouse moves away from them). This can be easily achieved by using the hover *pseudo-class* for all *label* and *input* elements e.g. label:hover and input:hover. For example, `label:hover { font-size:40px }`. Add these in the *style* section of the web page.
- The following <script> instruction needs to be added before `</body>`, near the end of the file. It is a link to some JavaScript code. After this is added the face image will be immediately shown in the web page after the user selects a file using the file selector (the first *input* element). The JavaScript code shows the selected image in an *img* element with `id="preview"` (which should be shown under the file selector). Therefore, please ensure you use this id for your img.

```
<script src="https://www.cse.ust.hk/~rossiter/dating_web_site.js" (https://www.cse.ust.hk/~rossiter/dating_web_site.js")></script>
```