Web Programming Name: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_\_

Activity 1.1 – Build a Cat Phot App

Points: \_\_\_\_\_\_\_\_\_\_ / 25

HTML tags give a webpage its structure. You can use HTML tags to add photos, buttons, and other elements to your webpage.

In this activity, you'll learn the most common HTML tags by building your own cat photo app.

1. Download, extract, and rename the linked project from GitHub using the link provided in Google Classroom. When you rename the project folder, make sure you follow proper naming practices.

2. Next, open the file named *index.html*. Every webpage should have a file called index.html, which will generally contain your homepage content, that is, the text and images that people see when they first go to your site.

3. HTML elements have opening tags like <h1> and closing tags like </h1>. The text for an element goes between its opening and closing tags. Find the h1 element and change its text to: CatPhotoApp.

4. The h1 through h6 heading elements are used to signify the importance of content below them. The lower the number, the higher the importance, so h2 elements have less importance than h1 elements. Only use one h1 element per page and place lower importance headings below higher importance headings. Below the h1 element, add an h2 element with this text: Cat Photos.

5. The p element is used to create a paragraph of text on websites. Create a p element below your h2 element and give it the following text: See more cat photos in our gallery.

6. Commenting allows you to leave messages without affecting the browser display. It also allows you to make code inactive. A comment in HTML starts with <!--, contains any number of lines of text, and ends with -->. For example, the comment <!-- TODO: Remove h1 --> contains the text TODO: Remove h1. Add a comment above the p element with this text: TODO: Add link to cat photos. In addition, add a comment at the top of the document with the project name, your name, today’s date, and the title of course on separate lines.

7. HTML5 has some elements that identify different content areas. These elements make your HTML easier to read and help with Search Engine Optimization (SEO) and accessibility. Identify the main section of this page by adding a <main> opening tag before the h1 element, and a </main> closing tag after the p element.

8. In the previous step, you put the h1, h2, comment, and p elements inside the main element. This is called nesting. Nested elements should be placed two spaces further to the right of the element they are nested in. This spacing is called indentation and it is used to make HTML easier to read.

The h1 element, h2 element and the comment are indented two spaces more than the main element in the code below. Use the space bar on your keyboard to add two more spaces in front of the p element so that it is indented properly as well.

9. You can add images to your website by using the img element. img elements have an opening tag without a closing tag. A tag for an element without a closing tag is known as a self-closing tag. Add an img element below the p element. At this point, no image will show up in the browser.

10. HTML *attributes* are special words used inside the opening tag of an element to control the element's behavior. The src attribute in an img element specifies the image's URL or path (where the image is located). Here is an example of an img element with a src attribute pointing to the freeCodeCamp logo located in the images folder:

<img src="images/fcc\_secondary.svg">

Inside the existing img element, add a src attribute with this path:

images/relaxing-cat.jpg

11. All img elements should have an alt attribute. The alt attribute's text is used for screen readers to improve accessibility and is displayed if the image fails to load. For example,   
<img src="cat.jpg" alt="A cat"> has an alt attribute with the text A cat. Inside the img element, add an alt attribute with this text: A cute orange cat lying on its back

12. You can link to another page with the anchor (a) element. For example,   
<a href="https://freecodecamp.org"></a> would link to freecodecamp.org. Add an anchor element after the paragraph that links to https://freecatphotoapp.com. At this point, the link won’t show up in the preview.

13. A link's text must be placed between the opening and closing tags of an anchor (a) element. For example, <a href="https://www.freecodecamp.org">click here to go to freeCodeCamp.org</a> is a link with the text click here to go to freeCodeCamp.org. Add the anchor text link to cat pictures to the anchor element. This will become the link's text.

14. You can turn any text into a link, such as the text inside of a p element.

<p>I think <a href="https://www.freecodecamp.org">freeCodeCamp</a> is great.</p>

In the text of your p element, turn the words cat photos into a link by adding opening and closing anchor (a) tags around these words. Then set the href attribute to https://freecatphotoapp.com.

15. Now that you turned the text cat photos inside the p element into a link, you don't need the second link below the p element. Delete the entire anchor element below the p element.

16. Add a target attribute with the value \_blank to the anchor (a) element's opening tag, so that the link opens in a new tab.

17. In previous steps you used an anchor element to turn text into a link. Other types of content can also be turned into a link by wrapping it in anchor tags. Turn the image into a link by surrounding it with necessary element tags. Use https://freecatphotoapp.com as the anchor's href attribute value.

18. Before adding any new content, you should make use of a section element to separate the cat photos content from the future content. Take your h2, comment, p, and anchor (a) elements and nest them in a section element.

19. It is time to add a new section. Add a second section element below the existing section element.

20. Within the second section element, add a new h2 element with the text Cat Lists.

21. When you add a lower rank heading element to the page, it's implied that you're starting a new subsection. After the last h2 element of the second section element, add an h3 element with this text: Things cats love:

22. After the h3 element with the Things cats love: text, add an unordered list (ul) element. Note that nothing will be displayed at this point.

23. Use list item (li) elements to create items in a list. Here is an example of list items in an unordered list:

<ul>

<li>milk</li>

<li>cheese</li>

</ul>

Within the ul element nest three list items to display three things cats love: cat nip, laser pointers, and lasagna.

24. After the unordered list, add a new image with a src attribute value set to: images/lasagna.jpg.

And its alt attribute value to: A slice of lasagna on a plate.

25. The figure element represents self-contained content and will allow you to associate an image with a caption. Nest the image you just added within a figure element.

26. A figure caption (figcaption) element is used to add a caption to describe the image contained within the figure element. For example, <figcaption>A cute cat</figcaption> adds the caption A cute cat. After the image nested in the figure element, add a figcaption element with text set to: Cats love lasagna.

27. Emphasize the word love in the figcaption element by wrapping it in an emphasis em element.

28. After the figure element, add another h3 element with the text: Top 3 things cats hate:

29. The code for an ordered list (ol) is similar to an unordered list, but list items in an ordered list are numbered when displayed. After the second section element's last h3 element, add an ordered list with these three list items: flea treatment, thunder, and other cats.

30. After the ordered list, add another figure element.

31. Inside the figure element you just added, nest an img element with a src attribute set to images/cats.jpg.

32. To improve accessibility of the image you added, add an alt attribute with the text: Five cats looking around a field.

33. After the last img element, add a figcaption element with the text Cats hate other cats.

34. The strong element is used to indicate that some text is of strong importance or urgent. In the figcaption you just added, indicate that hate is of strong importance by wrapping it in a strong element.

35. It is time to add a new section. Add a third section element below the second section element.

36. Inside the third section element, add an h2 element with the text: Cat Form

37. Now you will add a web form to collect information from users. After the Cat Form heading, add a form element.

38. The action attribute indicates where form data should be sent. For example,   
<form action="/submit-url"></form> tells the browser that the form data should be sent to the path /submit-url. Add an action attribute with the value https://freecatphotoapp.com/submit-cat-photo to the form element.

39. The input element allows you several ways to collect data from a web form. Like img elements, input elements are *self-closing* and do not need closing tags. Nest an input element in the form element.

40. There are many kinds of inputs you can create using the type attribute. You can easily create a password field, reset button, or a control to let users select a file from their computer. Create a text field to get text input from a user by adding the type attribute with the value text to the input element.

41. In order for a form's data to be accessed by the location specified in the action attribute, you must give the text field a name attribute and assign it a value to represent the data being submitted. For example, you could use the following syntax for an email address text field:   
<input type="text" name="email">. Add the name attribute with the value catphotourl to your text field.

42. Placeholder text is used to give people a hint about what kind of information to enter into an input. For example, <input type="text" placeholder="Email address">. Add the placeholder text cat photo URL to your input element.

43. To prevent a user from submitting your form when required information is missing, you need to add the required attribute to an input element. There's no need to set a value to the required attribute. Instead, just add the word required to the input element, making sure there is space between it and other attributes.

44. Use the button element to create a clickable button. For example,   
<button>Click Here</button> creates a button with the text Click Here. Add a button element with the text Submit below the input element. The default behavior of clicking a form button without any attributes submits the form to the location specified in the form's action attribute.

45. Even though you added your button below the text input, they appear next to each other on the page. That's because both input and button elements are inline elements, which don't appear on new lines. The button you added will submit the form by default. However, relying on default behavior may cause confusion. Add the type attribute with the value submit to the button to make it clear that it is a submit button.

46. You can use radio buttons for questions where you want only one answer out of multiple options. Here is an example of a radio button with the option of cat: <input type="radio"> cat. Remember that input elements are *self-closing*. Before the text input, add a radio button with the option set as: Indoor

47. label elements are used to help associate the text for an input element with the input element itself (especially for assistive technologies like screen readers). For example, <label><input type="radio"> cat</label> makes it so clicking the word cat also selects the corresponding radio button. Nest your radio button inside a label element.

48. The id attribute is used to identify specific HTML elements. Each id attribute's value must be unique from all other id values for the entire page. Add an id attribute with the value indoor to the radio button. When elements have multiple attributes, the order of the attributes doesn't matter.

49. Create another radio button below the first one. Nest it inside a label element with Outdoor as the label text. Give the radio button an id attribute with outdoor as the value.

50. Notice that both radio buttons can be selected at the same time. To make it so selecting one radio button automatically deselects the other, both buttons must have a name attribute with the same value. Add the name attribute with the value indoor-outdoor to both radio buttons.

51. If you select the Indoor radio button and submit the form, the form data for the button is based on its name and value attributes. Since your radio buttons do not have a value attribute, the form data will include indoor-outdoor=on, which is not useful when you have multiple buttons. Add a value attribute to both radio buttons. For convenience, set the button's value attribute to the same value as its id attribute.

52. The fieldset element is used to group related inputs and labels together in a web form. fieldset elements are *block-level elements*, meaning that they appear on a new line. Nest the Indoor and Outdoor radio buttons within a fieldset element, and don't forget to indent the radio buttons.

53. The legend element acts as a caption for the content in the fieldset element. It gives users context about what they should enter into that part of the form. Add a legend element with the text Is your cat an indoor or outdoor cat? above both of the radio buttons.

54. Next, you are going to add some new form input elements, so add another fieldset element directly below the current fieldset element.

55. Add a legend element with the text What's your cat's personality? inside the second fieldset element.

56. Forms commonly use checkboxes for questions that may have more than one answer. For example, here's a checkbox with the option of tacos: <input type="checkbox"> tacos. Under the legend element you just added, add an input with its type attribute set to checkbox and give it the option of: Loving

57. Add an id attribute with the value loving to the checkbox input.

58. There's another way to associate an input element's text with the element itself. You can nest the text within a label element and add a for attribute with the same value as the input element's id attribute. Associate the text Loving with the checkbox by nesting only the text Loving in a label element and giving it an appropriate for attribute.

59. Add the name attribute with the value personality to the checkbox input element. While you won't notice this in the browser, doing this makes it easier for a server to process your web form, especially when there are multiple checkboxes.

60. Add another checkbox after the one you just added. The id attribute value should be lazy and the name attribute value should be the same as the last checkbox. Also add a label element to the right of the new checkbox with the text Lazy. Make sure to associate the label element with the new checkbox using the for attribute.

61. Add a final checkbox after the previous one with an id attribute value of energetic. The name attribute should be the same as the previous checkbox. Also add a label element to the right of the new checkbox with text Energetic. Make sure to associate the label element with the new checkbox.

62. Like radio buttons, form data for selected checkboxes are name / value attribute pairs. While the value attribute is optional, it's best practice to include it with any checkboxes or radio buttons on the page. Add a value attribute to each checkbox. For convenience, set each checkbox's value attribute to the same value as its id attribute.

63. In order to make a checkbox checked or radio button selected by default, you need to add the checked attribute to it. There's no need to set a value to the checked attribute. Instead, just add the word checked to the input element, making sure there is space between it and other attributes. Make the first radio button and the first checkbox selected by default.

64. Now you will add a footer section to the page. After the main element, add a footer element.

65. Nest a p element with the text No Copyright - freeCodeCamp.org within the footer element.

66. Turn the existing freeCodeCamp.org text into a link by enclosing it in an anchor (a) element. The href attribute should be set to <https://www.freecodecamp.org>.

67. Notice that everything you've added to the page so far is inside the body element. All page content elements that should be rendered to the page go inside the body element. However, other important information goes inside the head element. Add a head element above the body element.

68. The title element determines what browsers show in the title bar or tab for the page. Add a title element within the head element using the text below: CatPhotoApp

69. Notice that the entire contents of the page are nested within an html element. All other elements must be descendants of this html element. Add the lang attribute with the value en to the opening html tag to specify that the language of the page is English.

70. All pages should begin with <!DOCTYPE html>. This special string is known as a *declaration* and ensures the browser tries to meet industry-wide specifications. Add this declaration as the first line of the code.

71. You can set browser behavior by adding self-closing meta elements in the head. Here's an example:

<meta attribute="value">

Inside the head element, nest a meta element with an attribute named charset. Set to the value to utf-8 which tells the browser how to encode characters for the page. Note that meta elements are self-closing.

At this point, your website is complete. To wrap up this assignment, please do the following…

1. Compress the project folder and submit it to Google Classroom.

2. Print a copy of *index.html* and add it to your engineering notebook. Highlight and annotate your code by identifying the various HTML elements recording the purpose and function of each element. If you are stuck, use the references provided on Google Classroom under this assignment.

3. Finally, record the file structure for this project in your engineering notebook.

|  |  |
| --- | --- |
| **Project Criteria** | |
| Criteria | Point(s) |
| Website contains a <head> element with all necessary elements to reconstruct the desired website. | /3 |
| Website contains a <body> element with all necessary elements to reconstruct the desired website. | /10 |
| Website contains a <footer> element with all necessary elements to reconstruct the desired website. | /2 |
| **Development Mechanics** | |
| Criteria | Point(s) |
| All elements are structured using best practices. | /2 |
| Project folder/director structure follows industry best practices. | /2 |
| All folders/directories and files use proper naming style (i.e., file-name). | /2 |
| All documents use whitespace and comments to help organize the code. | /2 |
| Project root folder/directory has an appropriate name. | /1 |
| All documents have a comment at the top that includes the program’s title, the student’s name, the date, and the course’s title. | /1 |

/ 25