POINTS TO PONDER

HTML

- 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.

- 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 (where the image is located). An example of an `img` element using an `src` attribute: `<img src="https://www.example.com/the-image.jpg">`.

- 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`.

- You can link to another page with the anchor (`a`) element. For example, `<a href='https://freecodecamp.org'></a>` would link to `freecodecamp.org`.

- 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`.

- a `target` attribute with the value `_blank` to the anchor (`a`) element's opening tag, is for that the link opens in a new tab.

- Before adding any new content, you should make use of a `section` element to separate the cat photos content from the future content.

- unordered list (`ul`) element.
- 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>
```

- The `figure` element represents self-contained content and will allow you to associate an image with a caption.

- 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`.

-   is for italics

- The code for an ordered list (`ol`) is similar to an unordered list, but list items in an ordered list are numbered when displayed.
  Example-
  ```
  <ol>
      <li>flea treatment </li>
      <li>thunder </li>
      <li>other cats </li>
  </ol>
  ```

- The `strong` element is used to indicate that some text is of strong importance or urgent.

- 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`.

- 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.

- 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.

- 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">`.

- 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">`.

- 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.

- Use the `button` element to create a clickable button. For example, `<button>Click Here</button>` creates a button with the text `Click Here`.

- The default behavior of clicking a form button without any attributes submits the form to the location specified in the form's `action` attribute.

- 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 `type` attribute with the value `submit` to the `button` to make it clear that it is a submit button.

- You can use radio buttons for questions where you want only one answer out of multiple options.

- `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.

- 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.

- 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.

- 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.

- 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.

- 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.

- 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`.

- 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.
- 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.
- Now you will add a footer section to the page.
- The `title` element determines what browsers show in the title bar or tab for the page.

- Add the `lang` attribute with the value `en` to the opening `html` tag to specify that the language of the page is English.
- 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.
- One more thing. You should allow people to use their native language. Tell the browser to encode multiple languages by adding a `meta` element as a child of the `head` element. Set its `charset` attribute to `UTF-8`.