Understanding the Code: A Breakdown

Let's dissect the HTML code in the Lando Camps page to understand its structure and functionality.

HTML Structure

Below is the breakdown of the HTML code:

Image:

```
    <img class="alignnone size-full wp-image-5808" src="https://educ-elvlc-2021.sites.olt.ubc.ca/files/2022/12/shutterstock_145708730-1-
 scaled.jpg" alt="Kids jumping on the field" width="1877" height="740" />
```

The `<img>` tag is used to display an image on the webpage. In this case, it displays an image of kids jumping on a field. The `src` attribute specifies the URL of the image, while the `alt`, `width`, and `height` attributes provide alternative text and dimensions for the image.

Paragraph:

```
    <p style="padding-top: 10px; padding-bottom: 10px; font-size: 20px; line-height: 1.3;">
        Check out our summer camp offerings below! From programming to environmental science, there's truly something suitable for
 everyone.
    </p>
```

The `<p>` tag defines a paragraph of text. In this case, it provides an introductory message about the summer camp offerings. Inline styles are used to adjust the padding, font size, and line height of the paragraph.

Current Offerings Section:

```
    <h3>Current Offerings</h3>
    <div class="flexboxCurrent">
        [loop query="posts_per_page=15&post_type=page&meta_key=currentCamp&order=ASC"]
```

```
            <div class="flexbox-box">
                [the_post_thumbnail]
                <h3><strong>[link_with_content]</strong></h3>
                [cf campInfo]
                [cf nextSession]
                [cf registration]
                [cf learnmore]
            </div>
            [/loop]
        </div>
```

This section displays the current summer camp offerings. It starts with a `<h3>` tag, indicating a subheading for the section. Then, a `<div>` element with the class `flexboxCurrent` is used to contain the offerings. Within this container, a loop retrieves and displays information about each offering. Each offering is wrapped in a `<div>` with the class `flexbox-box`. Various Custom Field-based shortcodes (more info below) such as `[cf campInfo]` are used to display information about each camp.

Past Offerings Section:

```
        <h3>Past Offerings</h3>
        <div class="flexbox">
            [loop query="posts_per_page=15&post_type=page&meta_key=summerCamp&order=ASC"]
            <div class="flexbox-box">
                [the_post_thumbnail]
                <h3><strong>[link_with_content]</strong></h3>
                [cf campInfo]
                [cf nextSession]
                [cf registration]
                [cf learnmore]
            </div>
            [/loop]
        </div>
```

This section displays the past summer camp offerings, following a similar structure to the current offerings section. It includes a `<h3>` tag for the section heading and a `<div>` container with the class `flexbox`. Inside the loop, information about each past offering is displayed within `<div>` elements with the class `flexbox-box`.

Button:

```
        <a href="https://educ-elvlc-2021.sites.olt.ubc.ca/camps/gallery/">
            <button class="button-blue-box" role="button"><b>View Photos from Previous Camps</b></button>
        </a>
```

This section includes a button that links to a gallery of photos from previous camps. The `<a>` tag creates the link, and the `<button>` tag creates the button itself. It has a class of `button-blue-box` for styling purposes.

CSS Section:

```
        <style>
            /* CSS styles can be included here */
        </style>
```

The `<style>` element allows you to define CSS styles directly within the HTML document. This is useful for customizing the appearance of HTML elements. Alternatively, you can use the CSS editor provided in WordPress.

Custom Fields and Loops in WordPress:

Custom fields, such as `[cf campInfo]`, `[cf nextSession]`, `[cf registration]`, and `[cf learnmore]`, are utilized to dynamically display information about each summer camp offering. These fields are populated with data specific to each camp.

Loops, indicated by `[loop query="..."]`, are used to iterate through each post (or camp) and display its corresponding information. The loop retrieves posts based on certain criteria, such as the post type and custom meta keys (e.g., `currentCamp` and `summerCamp`). The parameters within the loop query, such as `posts_per_page`, `post_type`, `meta_key`, and `order`, define how the posts are queried and displayed. For example, `posts_per_page` specifies the maximum number of posts to display per page, `meta_key` defines the custom field to query, and `order` determines the sorting order of the retrieved posts (e.g., ascending or descending).