

CHAPTER 2 :- HTML Forms and Inputs

HTML Forms and Inputs – User Interaction: Theory

Forms are a fundamental part of HTML, enabling user interaction by collecting and sending data to a server. They consist of various elements such as `<form>`, `<input>`, `<textarea>`, `<select>`, and `<button>`. Each element has a specific role, and understanding them is crucial for building interactive and accessible web pages.

1. `<form>` Element

The `<form>` tag acts as a container for all input elements. It defines how and where the data collected by the form will be sent.

Attributes of `<form>`

1. **action:** Specifies the URL where form data is sent.
 - Example: `action="/submit_form"` sends data to `/submit_form`.
 2. **method:** Defines the HTTP method for submitting data.
 - **GET:** Appends data as query parameters in the URL.
 - **POST:** Sends data in the body of the HTTP request, more secure and suitable for sensitive data.
 3. **target:** Determines where the response will be displayed.
 - `_self` (default): Same tab.
 - `_blank`: New tab or window.
-

2. `<input>` Element

The `<input>` tag is used to create interactive controls for collecting data from users.

Common Input Types

- **text:** For single-line text input.
- **email:** Validates input as an email format.
- **number:** Allows only numeric input, often with `min` and `max` attributes.
- **password:** Hides user input for sensitive fields like passwords.
- **checkbox:** Allows selection of multiple options.
- **radio:** For selecting one option from a group.

Attributes of `<input>`

1. **name:** Identifies the data when it is sent to the server.
 2. **placeholder:** Displays example text inside the field.
 3. **required:** Ensures the field must be filled before submission.
 4. **value:** Specifies the default value for the input field.
-

3. `<textarea>` Element

The `<textarea>` tag is used for multi-line text input, such as comments or messages.

Example:

html

Copy code

```
<label for="feedback">Feedback:</label>
```

```
<textarea id="feedback" name="user_feedback" rows="5" cols="30"></textarea>
```

4. `<select>` and `<option>` Elements

The `<select>` tag creates a dropdown list, and `<option>` defines the choices.

Example:

html

Copy code

```
<label for="city">Choose a city:</label>
<select id="city" name="user_city">
  <option value="mumbai">Mumbai</option>
  <option value="kolhapur">Kolhapur</option>
</select>
```

5. `<button>` Element

The `<button>` tag creates a clickable button.

Types of Buttons:

1. **submit:** Submits the form (default behavior).
2. **button:** A generic button that requires JavaScript to function.
3. **reset:** Resets all form fields to their initial values.

Example:

html

Copy code

```
<button type="submit">Submit</button>
<button type="reset">Reset</button>
```

6. Accessibility Considerations

1. **Labels:** Always use `<label>` for each input to describe its purpose. Connect it to the input using the `for` attribute.
 2. **Keyboard Navigation:** Ensure users can navigate the form using the Tab key.
 3. **Error Messages:** Provide clear feedback for invalid inputs.
-

Key Concepts:

- Forms are the bridge between user input and the server.
- The `<form>` element organizes inputs and defines submission behavior.
- Input validation can be done with built-in HTML attributes like `required` and `pattern`.
- Accessibility ensures forms are usable by everyone, including people using assistive technologies.