Coding Bootcamp

HTML

Forms in HTML

Learning Objectives

By the end of this lesson, you will be able to:

◉ Illustrate the concept of forms in HTML for creating interactive and user-friendly web applications

◉ Identify the HTTP methods to ensure the data is transmitted correctly and securely

◉ Analyze the concept of checkboxes, radio buttons, and iframes to enable the integration of dynamic content within web pages

◉ Define the usage of input text fields to create accessible and user-friendly forms

Forms

Forms: Overview

HTML forms collect user input and send it to a server for processing. They are crucial for web applications, enabling actions like signing up, logging in, submitting feedback, and conducting searches.

An HTML document contains:

Normal content

Markup

Controls

Labels

Forms: Example

Here is an example of a form:

First Name:

Last Name:

Date of Birth:

Email ID:

Mobile Number:

SUBMIT   RESET

Forms: Syntax

An HTML form is defined using the <form> element, which can contain various form controls such as text fields, checkboxes, radio buttons, and submit buttons. The basic syntax is as follows:

```
<form>

Text fields, checkboxes, radio buttons,
submit buttons.

</form>
```

Forms: Action Attribute

The action attribute of **<form>** defines the process to be performed on the form.

The form can be processed using .php, .jsp, .asp, or any other URL.
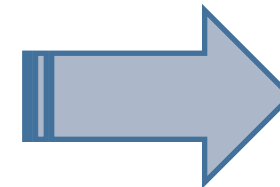
Input type

The input type="text" specifies a single-line text input field.

Label

Form elements have a label defined by the label tag.
Therefore, the attribute is the ID of the element that the label is associated with.

Forms: Example

```
<!DOCTYPE html>
<html>
<body>
<h2>HTML Forms</h2>
<form>
  <label for="fname">First name:</label><br>
  <input type="text" id="fname" name="fname"
value="Rehmat"><br>
  <label for="lname">Last name:</label><br>
  <input type="text" id="lname" name="lname"
value="kaur"><br><br>
  <input type="submit" value="Submit">
</form>
</body>
</html>
```

Output

HTML Forms

First name:
Rehmat

Last name:
kaur

Submit

Creating Forms Using HTML

Problem Statement:

You have been asked to create a form using HTML.

Outcome:

By completing this task, you created a form using HTML. You started by creating the index.html file, then added forms to the web page. Finally, you viewed the results in your local browser.

Note: Refer to the demo document for detailed steps:
01_Creating_Forms_Using_HTML

ASSISTED PRACTICE

Assisted Practice: Guidelines

Steps to be followed:

1. Create the index.html file
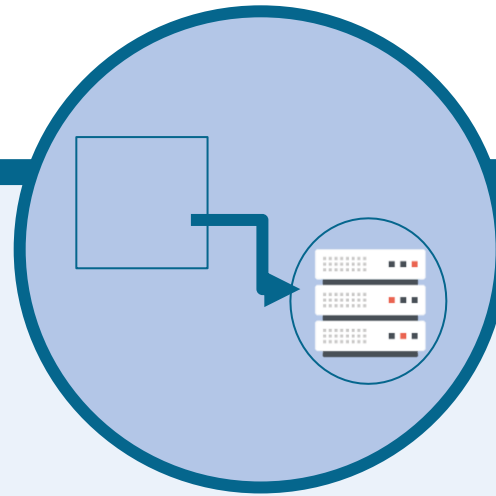2. Add forms to the web page
3. View the results in your local browser

HTTP Methods

HTTP Methods

HTTP methods specify the action to be performed on the server when a form is submitted. The two most common methods are GET and POST.
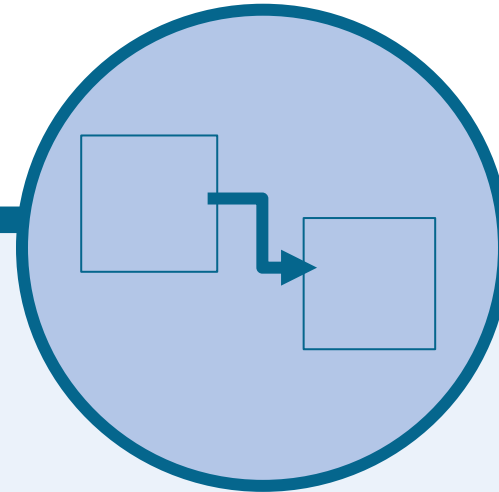
To send data to a server, utilize the POST method, since it:

- Is never cached

- Does not remain in the browser history

- Cannot be bookmarked

- Has no restrictions on data length

HTTP Methods

With the GET method, one may retrieve data from a specified resource since it:

- Remains in the browser history
- Can be bookmarked
- Is never used when dealing with sensitive data
- Has length restrictions
- Is used only to request data

Input Text Fields

Input Text Fields

An input text field in HTML is a form element that allows users to enter text. It is created using the <input> tag with the type attribute set to text.
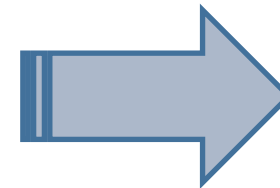
A single-line text input field is defined by **input type="text">.**

The text field's default width is 20 characters.

Input Text Fields: Example

```html
<!DOCTYPE html>
<html>
<body>
<h3>Text Fields</h3>
<form>
<label for="fname">First Name:</label><br>
<input type="text" id="fname" name="fname"><br>
<label for="lname">Last Name:</label><br>
<input type="text" id="lname"
name="lname"><br><br>
</form>
</body>
</html>
```

Output

Text Fields

First Name:

Last Name:

Select Menus

Select Menus

These are used to create drop-down lists in HTML forms. Here are the following select menu elements:

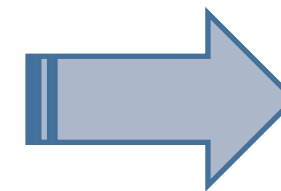<select>

Defines a drop-down list

<option>

Defines an option that can be selected

To define a pre-selected option, the user must add the selected attribute to the option.

Select Menus: Example

```
<!DOCTYPE html>
<html>
<body>
<h3>Select Menus</h3>
<form>
<label for="colors">Choose a color:</label>
<select id="colors" name="colors">
<option value="red">Red</option>
<option value="green">Green</option>
<option value="yellow">Yellow</option>
<option value="blue">Blue</option>
<option value="black">Black</option>
</select>
</form>
</body>
</html>
```

Output

Select Menus

Choose a color: Red

Red
Green
Yellow
Blue
Black

Checkboxes and Radio Buttons

Checkboxes

Checkboxes are commonly used to record inputs.

A checkbox allows one or more selections from a list.

Checkboxes: Example

The **<input type="checkbox">** defines a checkbox.

```
<!DOCTYPE html>
<html>
<body>
<form>
Choose languages you know:
<br>
<input type="checkbox" name="HTML"
value="yes">HTML<br>
<input type="checkbox" name="C#"
value="yes">C#<br>
<input type="checkbox" name="Java"
value="yes">Java<br>
</form>
</body>
</html>
```
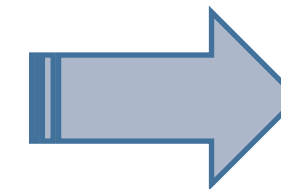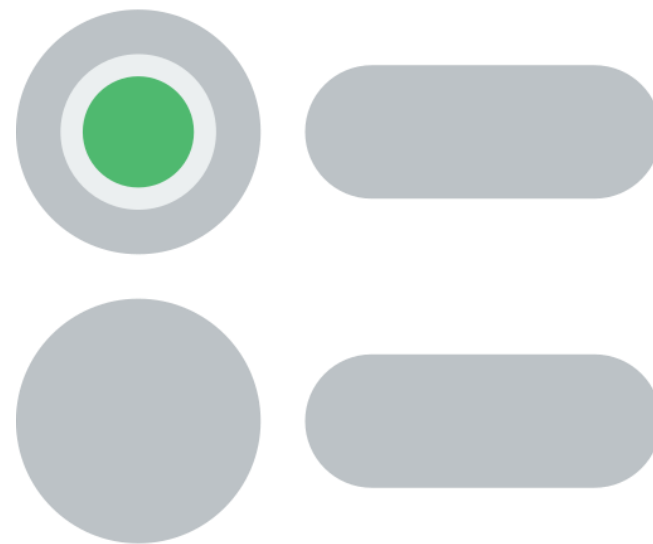
Output:

Choose languages you know:
☑ HTML
☐ C#
☑ Java

Radio Buttons

Radio buttons allow choosing one option.

A radio button is defined with the **<input type="radio">**.

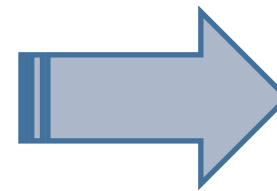# Radio Buttons: Example

```
<!DOCTYPE html>
<html>
<body>
<form>
Select the right option:
<br>
<input type="radio" name="male"
value="male">Male<br>
<input type="radio" name="female"
value="female">Female<br>
        </form>
</body>
</html>
```
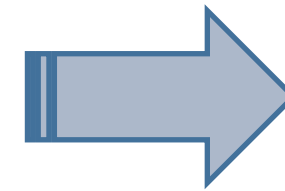
Output

Select the right option:
○ Male
○ Female

Text Areas, Buttons, and Iframes

Text Areas

The **<textarea>** tag defines multi-line text input control.

This element is used to collect the user inputs.

Text Areas: Example

```html
<!DOCTYPE html>
<html>
<body>
<h2>The textarea element</h2>
<form>
<textarea id="message" name="message" rows="4"
cols="50">
Our world is an amazing place. Let's keep our
planet neat and clean. Remember to use trash
bins and maintain hygiene in your surroundings.
</textarea>
<br>
</form>
</body>
</html>
```
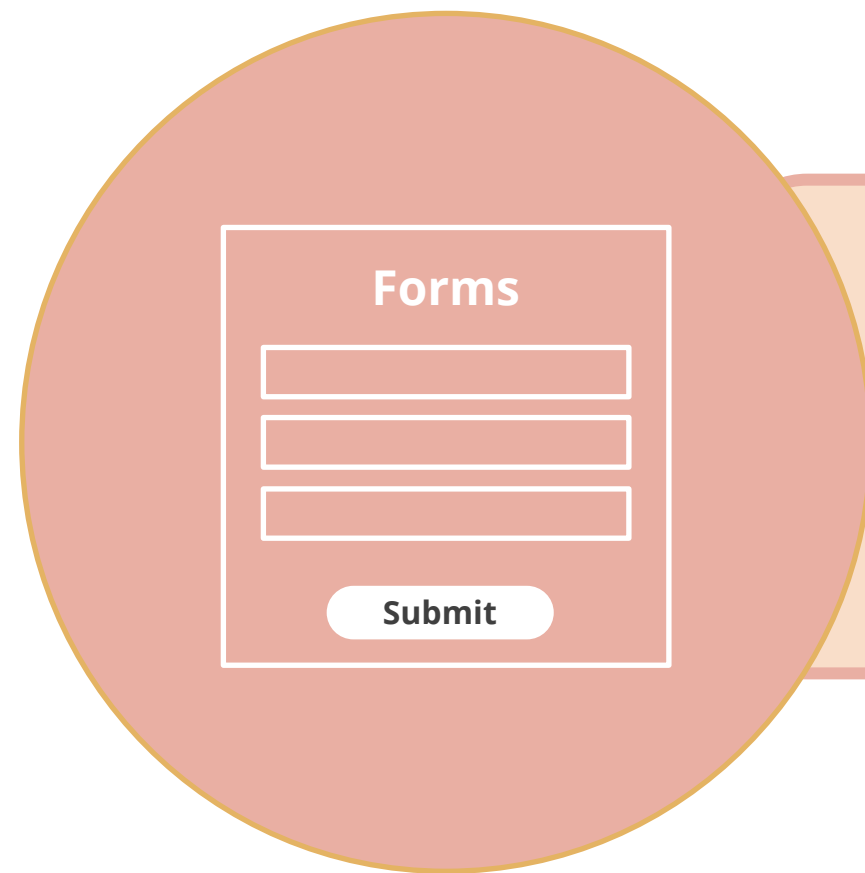
Output:

The textarea element

```
A text area can hold
an unlimited number
```

Button Tag

The button tag is used to create a button within a clickable form.

Forms

Submit

A user can submit information in a form by clicking the button.

Button Tag

The **<button>** tag is used for control.

Syntax:

```
<button type="button|submit|reset">
```
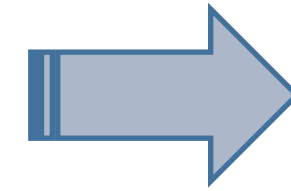
Clickable button

Submit form-data

Reset the form-data to its initial values

Button Tag: Example

```
<!DOCTYPE html>
<html>
<body>
<h2>Buttons</h2>
<form>
<label for="fname">First Name:</label><br>
<input type="text" id="fname" name="fname"><br>
<label for="lname">Last Name:</label><br>
<input type="text" id="lname"
name="lname"><br><br>
<button type="submit"
value="Submit">Submit</button>
<button type="reset"
value="Reset">Reset</button>
</form>
</body>
</html>
```

Output

Buttons

First Name:

Last Name:

Submit   Reset

Iframes

An iframe allows multiple HTML documents to be displayed independently.
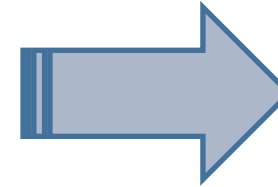
Each window shows a separate HTML page or web page. The syntax is as follows:

```
iframe src="url"
title="description"></iframe>
```
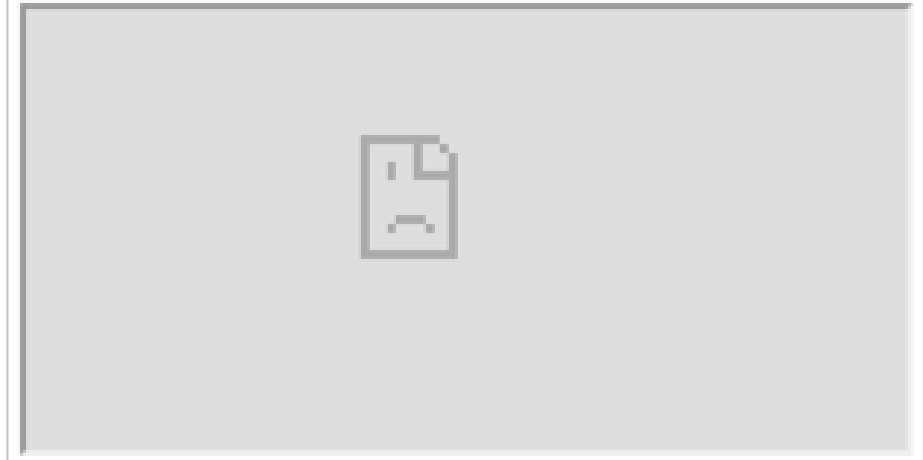
Iframes: Example

```
<!DOCTYPE html>
<html>
<body>
<h2>Iframes </h2>
<iframe src="" height="200" width="300"
title="Iframes">
</iframe>
</body>
</html>
```

Output

Iframes

Key Takeaways

- Input components are contained within the **<form>** element.

- The POST method is used to communicate data to a server.

- The GET method is used to obtain data from a given resource.

- Input text refers to the text that the user has entered in the text field.

- The select tag is used for the selected option that needs to be selected.

- The **<textarea>** tag defines multi-line text input control.

TECHNOLOGY

Thank You

simplilearn