HTML Essentials

Table & Form

Overview

1. HTML Table
2. HTML Form
3. Structure HTML Form
4. Sending form data
5. Form validation (HTML5)

➢ **Table:** A structured set of data made up of rows and columns (**tabular data**).

```
TABLE _____                    <table>
Row 1 _____          ____       <tr>
  cell 1 _____       |    _____  <td>
                               |    |          A
                               |    |_____  </td>
                               |    _____  <td>
  cell 2 _____       |    |          B
                               |    |_____  </td>
                               |____          </tr>

Row 2 _____          ____       <tr>
  cell 1 _____       |    _____  <td>
                               |    |          C
                               |    |_____  </td>
                               |    _____  <td>
  cell 2 _____       |    |          D
                               |    |_____  </td>
                               |____          </tr>

                                           </table>
```
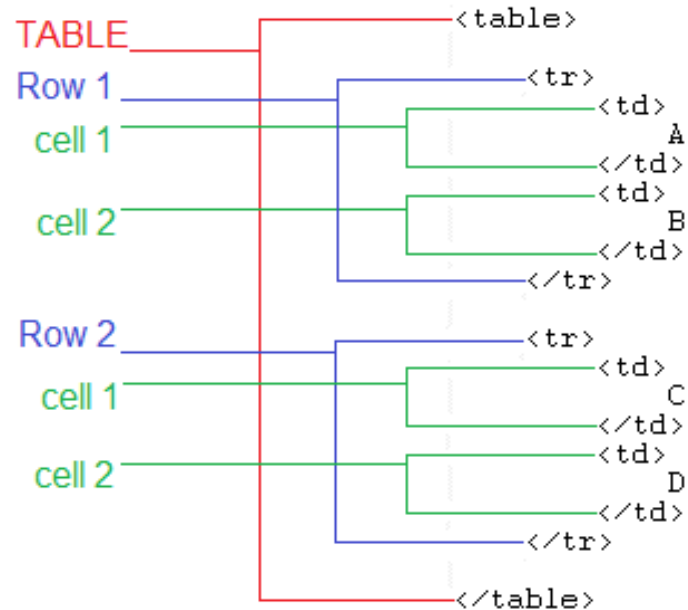
1. HMTL Table

> **Attributes:**
> - **Align:** This enumerated attribute indicates how the table must be aligned inside the containing document. It may have the following values: *(letf, center, right)*
> - **Bgcolor:** Specifies the background color of the table.
> - **Border:** This integer attribute defines, in pixels, the size of the frame surrounding the table. If set to 0, the frame attribute is set to void.

```
table, th, td {
    border: 1px solid black;
}
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

1. HMTL Table

> **Attributes:**
> - **border-collapse:** Sets whether the table borders should be collapsed into a single border

```
table {
    border-collapse: collapse;
}
```

Firstname	Lastname
Peter	Griffin
Lois	Griffin

> - **padding:** Control the space between the border and the content in a table

2. HTML Form

➢ **Overview:** The HTML <form> element defines a form that is used to collect user input. An HTML form contains **form elements**.

➢ **Form elements:** Are different types of input elements, like: text fields, checkboxes, radio buttons, submit buttons, and more.

First name:
John
Last name:
Doe

Submit

3. Structure HTML Form

➢ **Structure HTML Form:** The tags you use to manage the general structure of the form include:

- ▪ **<form>:** Defines a form and attributes that determine the form's behavior. Form element can contain one or more of the following form elements

=> Some attributes of form:

- ○ *action*: Specifies where to send the form-data when a form is submitted.
- ○ *method*: Specifies the HTTP method to use when sending form-data.
- ○ *target*: Specifies where to display the response that is received after submitting the form

3. Structure HTML Form

➢ **Structure HTML Form:** The tags you use to manage the general structure of the form include:

- ▪ **<fieldset>:** Groups a series of input elements together

- ▪ **<legend>:** Describes the purpose of the <fieldset>

- ▪ **<lable>:** Defines a label for an HTML form widget

- ▪ **<button>:** Defines a clickable button

3. Structure HTML Form

> **Structure HTML Form:** The tags you use to manage the general structure of the form include:

- **<input>:** Can be displayed in several ways, depending on the type attribute.

| Number: | → type="number" |
Date: mm/dd/yyyy	→ type="date"
Color:	→ type="color"
Range:	→ type="range"

3. Structure HTML Form

- ➢ **Structure HTML Form:** The tags you use to manage the general structure of the form include:
    - ▪ **<select>:** Defines a **drop-down list,** the <option> tags inside the <select> element define the available options in the list.
    - ⇒ Some attributes of select:
        - o *disabled*: Specifies that a drop-down list should be disabled
        - o *form*: Defines which form the drop-down list belongs to
        - o *multiple*: Specifies that multiple options can be selected at once
        - o *required*: Specifies that the user is required to select a value before submitting the form

➢ **Sending form data**

Let remember client-server architecture

An HTML form on a web page is a way to configure an HTTP request to send data to a server. This enables the user to provide information to be delivered in the HTTP request.
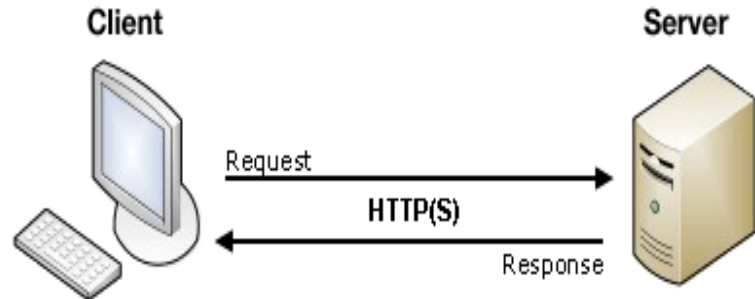
4. Sending form fata

➢ **Sending form data:**
- ▪ **On client side:** Defining how to send the data
  - o Action attribute: Defines where the data gets sent. Its value must be a valid relative or absolute URL

```
1 | <form action="https://example.com">
```

  - o Method attribute: Defines how data is sent. HTML form data can be transmitted via Post method or Get method.

```
1 | <form action="http://www.foo.com" method="GET">
```

4. Sending form fata

➢ **GET method:** The method used by the browser to ask the server to send back a given resource**.** In this case, the browser sends an empty body. Because the body is empty, if a form is sent using this method the data sent to the server is appended to the URL.

➢ **POST method:** The method the browser uses to talk to the server when asking for a response that takes into account the data provided in the body of the HTTP request. When the form is submitted using the POST method, you get no data appended to the URL

5. Form validation

- ➢ **Form validation (HTML5)**
    - ▪ '**required**': Specifies whether a form field needs to be filled in before the form can be submitted.
    - ▪ '**minlength**' and '**maxlength**': Specifies the minimum and maximum length of textual data (strings).
    - ▪ '**min**' and '**max**': Specifies the minimum and maximum values of numerical input types.
    - ▪ '**type**': Specifies whether the data needs to be a number, an email address, or some other specific preset type.
    - ▪ '**pattern**': Specifies a regular expression that defines a pattern the entered data needs to follow.

5. Form validation

➤ **Form validation (HTML5)**
   ✓ When an element is valid: The element matches the :**valid** CSS pseudo-class, which lets you apply a specific style to valid elements.
   ✓ When an element is invalid: The element matches the :**invalid** CSS pseudo-class, and sometimes other UI pseudo-classes (e.g., :out-of-range) depending on the error, which lets you apply a specific style to invalid elements.

5. Quiz 1

➤ Fill in the blanks:

```
1  <html>
2    <body>
3       <____ method="POST">
4          <____ src="image.jpg" /> <br />
5          <input ____="text" name="name" />
6          <input type="submit" name="Submit" />
7       </form>
8    </body>
9  </html>
```

➢ Fill in the blanks and align the table to the center of the page:

```
1  ∨ <html>
2    <head>
3    <title>My Page</title>
4    <____>
5    <body>
6      <table align="center">
7  ∨     <tr><td>
8          <ul>
9            <li>item</li>
10           <____>
11         </td></tr>
12       <____>
13       ____
14   </html>
```

5. Quiz 3

➤ Which of these tags are table tags?

    A. <table><tr><td>

    B. <table><tr><tt>

    C. <thead><body><tr>

    D. <tr><tt><dd>

Thank you