

HTML LEVEL 3

1. Tables in HTML

Tables are used to display data in rows and columns, like spreadsheets.

```
<table>
  <tr>
    <th>Name</th>
    <th>Age</th>
  </tr>
  <tr>
    <td>John</td>
    <td>25</td>
  </tr>
</table>
```

- `<table>` - starts the table.
 - `<tr>` - table row.
 - `<th>` - table header (bold by default).
 - `<td>` - table data (normal cell).
-

2. Semantics in Table

Semantics = meaning behind the tags. In tables:

- `<table>` = table structure.
- `<thead>` = table header section.
- `<tbody>` = main body of the table.
- `<tfoot>` = footer of the table (for totals or summary).

```
<table>
  <thead>
    <tr><th>Product</th><th>Price</th></tr>
  </thead>
  <tbody>
    <tr><td>Pen</td><td>$2</td></tr>
  </tbody>
  <tfoot>
    <tr><td>Total</td><td>$2</td></tr>
  </tfoot>
</table>
```

✅ **Extra Info:** Semantic tags improve **SEO** (search engines can read your table better) and **accessibility** (screen readers for visually impaired people).

3. Colspan and Rowspan attributes

Used when you want a cell to *stretch over multiple columns or rows*.

- **colspan** = merges columns.
- **rowspan** = merges rows.

```
<table border="1">
  <tr>
    <td colspan="2">Merged 2 columns</td>
  </tr>
  <tr>
    <td rowspan="2">Merged 2 rows</td>
    <td>Row 1</td>
  </tr>
  <tr>
    <td>Row 2</td>
  </tr>
</table>
```

✅ **Tip:** Useful when creating invoice tables or calendars.

4. Forms in HTML

Forms are used to *collect user input* (login forms, search bars, etc.).

html

CopyEdit

```
<form>
  <!-- form inputs will go here -->
</form>
```

5. Action Attribute

Tells where (which URL) to send the form data when submitted.

```
<form action="submit.php">
  <!-- form content -->
</form>
```

- If empty (`action=""`), it will submit to the same page.
-

6. Form Element

The `<form>` element wraps all input fields, buttons, etc.

Attributes you can add:

- **method="get" or "post"** (how to send data).
- **action** (where to send data).
- **autocomplete** (on/off for browser to remember previous inputs).

7. Input and Types of Input

The `<input>` tag is used to get different types of user input.

Common types:

```
<input type="text">
```

```
<input type="email">
```

```
<input type="password">
```

```
<input type="number">
```

```
<input type="date">
```

```
<input type="file">
```

```
<input type="submit">
```

✅ **Tip:** Always match input types to expected data for better UX and validation.

8. Placeholder Attribute

Shows **temporary text inside input** as a hint.

```
<input type="text" placeholder="Enter your name">
```

Once user types, it disappears.

9. Label

Used to **name or describe an input**.

```
<label for="email">Email:</label>
```

```
<input type="email" id="email">
```

✅ **Bonus:** Clicking on the label will focus the input (makes form more accessible).

10. Button Element

Adds a clickable **button** inside forms or pages.

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

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

```
<button type="button">Click Me!</button>
```

- **submit** = submits form.
- **reset** = clears form inputs.
- **button** = just a clickable button (add JS for action).

11. Button Type Input

Another way to create buttons:

```
<input type="submit" value="Submit">
```

```
<input type="reset" value="Reset">
```

```
<input type="button" value="Click Me">
```

Both `<button>` and `<input type="button">` do similar jobs.

12. Name Attribute

Used to identify form fields when sending data.

```
<input type="text" name="username">
```

✓ **Tip:** Backend (PHP, Node, etc.) uses name to access input values.

13. Input Element Checkbox

Used to select multiple options.

```
<input type="checkbox" name="skills" value="HTML">HTML
```

```
<input type="checkbox" name="skills" value="CSS">CSS
```

✓ Users can select one or many.

14. Input Element Radio Button

Used to select only one option from a group.

```
<input type="radio" name="gender" value="male">Male
```

```
<input type="radio" name="gender" value="female">Female
```

✓ Same name = only one can be selected.

15. Select Element

Creates a dropdown menu.

```
<select name="country">
```

```
  <option value="india">India</option>
```

```
  <option value="usa">USA</option>
```

```
</select>
```

✓ **Extra:** Add multiple attribute to allow multiple selections.

16. Range

For sliders (numeric values in a range).

```
<input type="range" min="0" max="100" value="50">
```

✓ Can style it with CSS.

17. Textarea

For multi-line text input (comments, messages).

```
<textarea rows="4" cols="30" placeholder="Your message here"></textarea>
```

✓ More space than `<input type="text">`.

18. Fieldset and Legend

Used to **group related form fields** together.

```
<fieldset>
  <legend>Personal Info</legend>
  <label for="name">Name:</label>
  <input type="text" id="name">

  <label for="age">Age:</label>
  <input type="number" id="age">
</fieldset>
```

- `<fieldset>` = groups related fields with a box.
- `<legend>` = title/caption for the group.

✅ **Tip:** Useful in long forms (e.g., registration forms).

19. Hidden Input

Used to send **invisible data** with a form.

```
<input type="hidden" name="userid" value="12345">
```

✅ Backend can use this data (e.g., user ID, tokens).

20. Disabled and Readonly Attributes

- `disabled` = cannot interact or submit the input.
- `readonly` = can see but cannot edit.

```
<input type="text" value="Locked" readonly>
<input type="text" value="Disabled" disabled>
```

✅ disabled input won't be sent to server when form submits!

21. Autofocus Attribute

Automatically **focuses the input** when page loads.

```
<input type="text" autofocus>
```

✅ Improves user experience by guiding the user to the first input.

22. Datalist Element

Gives a **list of suggestions** (auto-complete dropdown).

```
<input list="browsers" name="browser">
<datalist id="browsers">
  <option value="Chrome">
  <option value="Firefox">
  <option value="Safari">
</datalist>
```

- ✓ Similar to Google Search autocomplete behavior.
-

23. Form Validation Attributes

HTML5 has built-in form validation, no JavaScript needed:

- **required**
- **minlength / maxlength**
- **min / max** (for numbers)
- **pattern** (regex)
- **type validation** (e.g., email must have @)

Example:

```
<input type="email" required>
```

```
<input type="text" pattern="[A-Za-z]+" title="Letters only">
```