

Lesson:

List



List

In HTML, a list is a structured way to present a collection of related items. Lists are commonly used to organize and display information in a structured and readable format. HTML provides several tags specifically designed for creating different types of lists.

The main types of lists in HTML are:

- ul
- ol
- dl

Let us now discuss each one of them and look at examples

- **ul:** It stands for an unordered list, which means every list item will have a bullet point in front of it.

Ex:

```
Unset
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <ul>
      <li>Item 1</li>

      <li>Item 2</li>
      <li>Item 3</li>
      <li>Item 4</li>
    </ul>
  </body>
</html>
```

In the above example code, we have a ul with 4 li elements inside the body tag. A li represents a single list item.

The output of the above code would look as shown in the image below

- Item 1
- Item 2
- Item 3
- Item 4

You see every list item is displayed with a bullet point in front of it. This indicates it is an unordered list. We can also nest the list as shown below

Ex:

```
Unset
<ul>
  <li>item</li>
  <li>item</li>
  <li>item</li>
  <ul>
    <li>Sub item</li>
    <li>Sub item</li>
    <ul>
      <li>Sub sub item</li>
      <li>Sub sub item</li>
    </ul>
  </ul>
</ul>
```

The output of the above code would look like this

```
1. Item 1
2. Item 2
3. Item 3
4. Item 4
  1. Sub Item 1
  2. Sub Item 2
    .
      1. Sub sub Item 1
      2. Sub sub Item 2
```

We have type attribute also available on the ul tag but it is not to be used anymore as it has been deprecated. We will learn how we can achieve this using CSS later in the course. But for reference let us look at them

- **disc:** This is the default behavior as we have seen in the previous example
- **circle:** This is almost similar to disc but is unfilled

Ex:

```
JavaScript
<ul type="circle">
    <li>Item 1</li>

    <li>Item 2</li>
    <li>Item 3</li>
    <li>Item 4</li>
</ul>
```

Output:

- Item 1
- Item 2
- Item 3
- Item 4

square: This makes the shape of the bullet points square

Ex:

```
JavaScript
<ul type="square">
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
    <li>Item 4</li>
</ul>
```

Output:

- Item 1
- Item 2
- Item 3
- Item 4

NOTE: There is also a triangle, but not all the browsers support it.

Similar to ul tag the li tag also supports type attribute with the following values

Similar to **ul** tag the **li** tag also supports type attribute with the following values

```
JavaScript
type="circle"
type="square"
type="disc"
```

Ex:

```
JavaScript
<ul>
  <li type="circle">item</li>
  <li type="square">item</li>
  <li type="disc">item</li>
</ul>
```

Output:

- item
 - item
 - item

Note: As these technologies progressed and matured, it was decided that all styling changes should be controlled solely by **CSS**, resulting in the deprecation of similar attributes. This deprecation extends to both the **li** tag within **ul** and the **ul** tag itself. Instead, we will delve into achieving these effects using **CSS** later in the course.

ol: It stands for ordered list, which means every list item will have a number in front of it.

Ex:

```
Unset
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <ol>
      <li>Item 1</li>
      <li>Item 2</li>
      <li>Item 3</li>
      <li>Item 4</li>
    </ol>
  </body>
</html>
```

In the above example code, we have a **ol** with 4 **li** elements inside the **body** tag. A **li** represents a single list item similar to what we have seen inside the **ul**.

The output of the above code would look as shown in the image below

1. Item 1
2. Item 2
3. Item 3
4. Item 4

We can also nest the list as shown below

Ex:

```
Unset
<ol>
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
  <li>Item 4</li>
  <ol>
    <li>Sub Item 1</li>
    <li>Sub Item 2</li>.
    <ol>
      <li>Sub sub Item 1</li>
      <li>Sub sub Item 2</li>
    </ol>
  </ol>
</ol>
```

The output of the above code would look like this

- 1. Item 1
- 2. Item 2
- 3. Item 3
- 4. Item 4
 - 1. Sub Item 1
 - 2. Sub Item 2
 - 1. Sub sub Item 1
 - 2. Sub sub Item 2

We also have a start attribute which we can use to define the start number

Ex:

```
JavaScript
<!DOCTYPE html>
<html lang="en">

<head>
  <meta charset="UTF-8" />
  <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
  <title>Document</title>
</head>

<body>
  <ol start="3">
    <li>Item 1</li>
    <li>Item 2</li>
    <li>Item 3</li>
    <li>Item 4</li>
  </ol>
</body>

</html>
```

Output:

- 3. Item 1
- 4. Item 2
- 5. Item 3
- 6. Item 4

There is another handy attribute called type, which lets us change the numbering style of the items list

JavaScript

```
type="1"  
type="A"  
type="a"  
type="I"  
type="i"
```

- **type="1"** is the default one.
- type="A": This will change the numbers to capital A, B, C, D

- A. Item 1
- B. Item 2
- C. Item 3
- D. Item 4

- type="a": This will change the numbers to lowercase a, b, c, d

- a. Item 1
- b. Item 2
- c. Item 3
- d. Item 4

- type="I": This will change the numbers to capital roman numbers

- I. Item 1
- II. Item 2
- III. Item 3
- IV. Item 4

- type="i": This will change the numbers to lowercase roman numbers

- i. Item 1
- ii. Item 2
- iii. Item 3
- iv. Item 4

You can also use the start attribute with any of the type attributes

Ex:

```
JavaScript
<ol start="3" type="A">
  <li>Item 1</li>
  <li>Item 2</li>
  <li>Item 3</li>
  <li>Item 4</li>
</ol>
```

Output:

- C. Item 1
- D. Item 2
- E. Item 3
- F. Item 4

As you can see the first 2 (**A, B**) are omitted and the list starts with **C**

The above **type** values also apply to **li** tag

```
JavaScript
type="1"
type="A"
type="a"
type="I"
type="i"
```

Ex:

JavaScript

```
<ol>
  <li type="a">item</li>
  <li type="A">item</li>
  <li type="I">item</li>
  <li type="i">item</li>
  <li type="1">item</li>
</ol>
```

Output:

- a. item
- B. item
- III. item
- iv. item
- 5. item

Note: As these technologies progressed and matured, it was decided that all styling changes should be controlled solely by **CSS**, resulting in the deprecation of similar attributes. This deprecation extends to both the **li** tag within **ol** and the **ol** tag itself. Instead, we will delve into achieving these effects using **CSS** later in the course.

- **dl:** It stands for Definition List. It represents a list of terms and their corresponding definitions

Definition list consists of **<dt>** (Definition Term) and **<dd>** (Definition Description) pairs.

Ex:

```
Unset
<!DOCTYPE html>
<html lang="en">
  <head>
    <meta charset="UTF-8" />
    <meta name="viewport" content="width=device-width,
initial-scale=1.0" />
    <title>Document</title>
  </head>
  <body>
    <dl>
      <dt>Term 1</dt>
      <dd>Definition 1</dd>
      <dt>Term 2</dt>
      <dd>Definition 2</dd>
    </dl>
  </body>
</html>
```

In the above example code, we have a dl with 2 dt elements and 2 dd elements inside the body tag.

The output of the above code would look as shown in the image below

Term 1

Definition 1

Term 2

Definition 2

Now, you must be thinking, is there a way I can change the ul bullet list from circles to something else? In short, yes, we can change them to something different with the help of CSS (We will learn this later in the course). There is a way to change them in HTML using type attributes but it is not preferred