

What are Lists in HTML

Lists exist to group related items in a meaningful order. That order (or lack of it) is not visual decoration — it's semantic information. Browsers, screen readers, SEO crawlers all care about this.

If items have sequence, priority, or steps → ordered list

If items are just a collection → unordered list

That's the first-principles rule. Everything else is implementation.

1. Unordered List ()

Use this when order does NOT matter.

```
<ul>
  <li>HTML</li>
  <li>CSS</li>
  <li>JavaScript</li>
</ul>
```

Key points

- Default bullets (•), but bullets are not the point
- Meaning: “These items are equal; sequence is irrelevant”
- Common use cases:
 - Navigation menus
 - Feature lists
 - Ingredients
 - Tags, categories

Strategic mistake beginners make

Using just because they want bullets.

That's wrong thinking — semantics first, styling second.

2. Ordered List ()

Use this when order matters.

```
<ol>
  <li>Install code editor</li>
  <li>Write HTML</li>
  <li>Style with CSS</li>
</ol>
```

Key points

- Default numbering (1, 2, 3...)
- Meaning: “This sequence matters”
- Common use cases:
 - Step-by-step tutorials
 - Rankings
 - Instructions
 - Timelines

Extra control

```
<ol type="A">
  <li>Plan</li>
  <li>Design</li>
  <li>Build</li>
</ol>
```

```
<ol start="5">
  <li>Step Five</li>
  <li>Step Six</li>
</ol>
```

These exist because HTML cares about logic, not just looks.

3. — the non-negotiable rule

 must always live inside or .

Never alone. Never as a shortcut.

```
<ul>
  <li>Item</li>
</ul>
```

4. Nesting Lists (real-world usage)

Lists can contain other lists — this is how menus and outlines work.

```
<ul>
  <li>Frontend
    <ul>
      <li>HTML</li>
      <li>CSS</li>
    </ul>
  </li>
  <li>Backend</li>
</ul>
```