



Mastering List Styling

A step-by-step guide from raw HTML to polished design with CSS.

Our Canvas: A Simple Set of Lists

Throughout this guide, we will transform a standard set of HTML lists. We'll start with this unstyled 'Hummus Recipe' example, which includes an unordered list (), an ordered list (), and a description list (<dl>).

```
<h2 style="color: #6A7A8F;">Shopping (unordered list)</h2>
<ul style="color: #D4A056;">
  <li style="color: #D4A056;">Hummus</li>
  <li style="color: #D4A056;">Pita</li>
  <li style="color: #D4A056;">Green salad</li>
  <li style="color: #D4A056;">Halloumi</li>
</ul>
```

```
<h2 style="color: #6A7A8F;">Recipe (ordered list)</h2>
<ol style="color: #D4A056;">
  <li style="color: #D4A056;">Toast pita, leave to cool...</li>
  <li style="color: #D4A056;">Fry the halloumi...</li>
  <li style="color: #D4A056;">Wash and chop the salad.</li>
  <li style="color: #D4A056;">Fill pita with salad...</li>
</ol>
```

```
<h2 style="color: #6A7A8F;">Ingredients (description list)</h2>
<dl style="color: #D4A056;">
  <dt style="color: #D4A056;">Hummus</dt>
  <dd style="color: #D4A056;">A thick dip/sauce...</dd>
  <dt style="color: #D4A056;">Pita</dt>
  <dd style="color: #D4A056;">A soft, slightly leavened flatbread.</dd>
  <dt style="color: #D4A056;">Halloumi</dt>
  <dd style="color: #D4A056;">A semi-hard, unripened cheese...</dd>
  <dt style="color: #D4A056;">Green salad</dt>
  <dd style="color: #D4A056;">That green healthy stuff...</dd>
</dl>
```




1.
2.

<dl>

Term
Definition

The Browser's Default Styling

Without any CSS, browsers apply their own default styles. Notice the standard bullets, numbers, and inconsistent indentation. This is our starting point.

Shopping (unordered list)

- Hummus
- Pita
- Green salad
- Halloumi

Recipe (ordered list)

1. Toast pita, leave to cool, then slice down the edge.
2. Fry the halloumi in a shallow, non-stick pan, until browned on both sides.
3. Wash and chop the salad.
4. Fill pita with salad, hummus, and fried halloumi.

Ingredients (description list)

Hummus

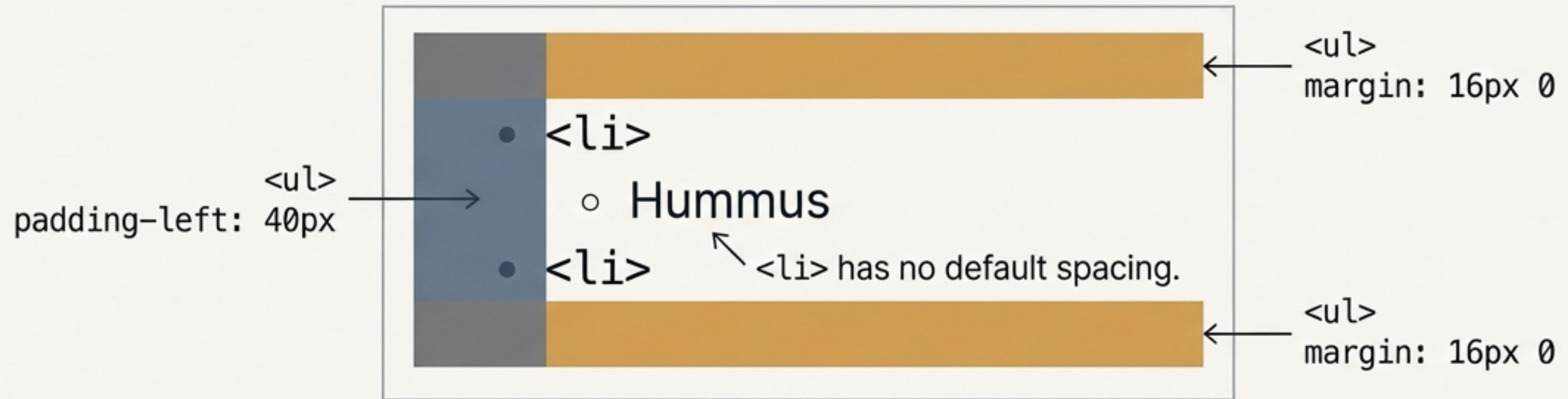
A thick dip/sauce generally made from chick peas blended with tahini, lemon juice, salt, garlic, and other ingredients.

Pita

A soft, slightly leavened flatbread.

Deconstructing the Default Spacing

The default appearance is controlled by built-in margins and padding. Understanding these is the first step to taking control.



- `<ul>` & `<ol>`: Have a top and bottom margin of 16px and a padding-left of 40px.
- `<dd>`: Have a margin-left of 40px.
- `<li>`: Have no default spacing.

The Foundation: Establishing Vertical Rhythm

Before targeting list-specific styles, we must ensure our lists share the same vertical spacing and typographic scale as surrounding elements. This creates a harmonious, professional layout.

✘ Before

Shopping (unordered list)

- Hummus
- Pita
- Green salad
- Halloumi

Recipe (ordered list)

1. Toast pita, leave to cool, then slice down the edge.
2. Fry the halloumi in a shallow, non-stick pan, until browned on both sides.
3. Wash and chop the salad.
4. Fill pita with salad, hummus, and fried halloumi.

✔ After

Shopping (unordered list)

- Hummus
- Pita
- Green salad
- Halloumi

Recipe (ordered list)

1. Toast pita, leave to cool, then slice down the edge.
2. Fry the halloumi in a shallow, non-stick pan, until browned on both sides.
3. Wash and chop the salad.
4. Fill pita with salad, hummus, and fried halloumi.

```
html {  
  font-family: Helvetica, Arial, sans-serif;  
  font-size: 10px;  
}  
  
ul, ol, dl {  
  font-size: 1.5rem;  
}  
  
li, dd, dt {  
  line-height: 1.5;  
}  
  
dt {  
  font-weight: bold;  
}
```

The List Styling Toolkit

CSS provides three core properties for controlling the appearance of list markers on `<ul>` and `<ol>` elements. We will master each one in turn.



`list-style-type`

Sets the type of marker to use.



`list-style-position`

Sets whether markers appear inside or outside the text flow.



`list-style-image`

Allows a custom image to be used as the marker.

Customizing Markers with `list-style-type`

This property changes the bullet or numbering style. For example, we can switch our ordered list from default numbers to uppercase Roman numerals.

☰ Before

1. Toast pita, leave to cool, then slice down the edge.
2. Fry the halloumi...
3. Wash and chop the salad.

↓☰ After

- I. Toast pita, leave to cool, then slice down the edge.
- II. Fry the halloumi...
- III. Wash and chop the salad.

```
ol {  
  list-style-type: upper-roman;  
}
```

Controlling Flow with `list-style-position`

The `list-style-position` property determines if the marker sits outside the list item (the default) or inside, as part of the text flow.

Value: `outside` (Default Behavior)

- I. Toast pita, leave to cool, then slice down the edge so you can easily stuff it with delicious fillings.

Value: `inside`

- I. Toast pita, leave to cool, then slice down the edge so you can easily stuff it with delicious fillings.

```
ol {  
  list-style-type: upper-roman;  
  list-style-position: inside;  
}
```

A Basic Approach: Using `list-style-image`

For custom markers, the `list-style-image` property lets you specify a URL for your bullet. It's a quick and simple solution.

```
ul {  
  list-style-image: url("star.svg");  
}
```



A Word of Caution

This property is limited. You have no direct control over the image's size or precise position, which can lead to alignment issues.

PRO-TIP

The Professional Technique: Full Control with Backgrounds

For pixel-perfect custom bullets, the best practice is to remove the default marker and apply a `background-image` to each `<li>` element. This gives you complete control over size, position, and repetition.

Before	After
• Hummus • Pita • Green salad	★ Hummus ★ Pita ★ Green salad

Deconstructing the Background Image Method

This technique involves three key steps: removing the original bullet, creating space for the new one, and then placing the background image precisely.

1

Hide the Default Bullet

First, we remove the original disc from the `<ul>`.

```
ul { list-style-type: none; }
```

2

Adjust Spacing

Next, we add padding to both the `<ul>` and the `<li>`. This creates a clean space for our custom bullet, preventing it from overlapping the text.

```
ul { padding-left: 2rem; }  
ul li { padding-left: 2rem; }
```

3

Apply the Background

Playfair Display

Finally, we add the background properties to the `<li>` to place our custom image.

```
ul li {  
  background-image: url(...);  
  background-position: 0 0;  
  background-size: 1.6rem 1.6rem;  
  background-repeat: no-repeat;  
}
```

The Background Properties Explained

These four properties work together to position our custom bullet.

``background-image``

References the path to the image file (`url("star.svg")`).

``background-position``

Defines where the image appears. We use `0 0` to place it in the top-left corner of the `<li>`'s padding.

``background-size``

Sets the image dimensions. We use `1.6rem` to fit nicely within our padding.

``background-repeat``

We use `no-repeat` because we only want one bullet per list item.

```
ul {  
  padding-left: 2rem;  
  list-style-type: none;  
}  
ul li {  
  padding-left: 2rem;  
  background-image: url("https://mdn.github.io/learning-area/css/styling-text/styling-lists/star.svg");  
  background-position: 0 0;  
  background-size: 1.6rem 1.6rem;  
  background-repeat: no-repeat;  
}
```

The Final Polish: Writing Cleaner Code with `list-style`

To save time and space, you can combine `list-style-type`, `list-style-position`, and `list-style-image` into a single shorthand property: `list-style`.

The Long Way

```
ul {  
  list-style-type: disc;  
  list-style-position: inside;  
  list-style-image: url(example.png);  
}
```



The Shorthand Way

```
ul {  
  list-style: disc inside url(example.png);  
}
```

Mastering the `list-style` Shorthand

Remember these key rules when using the **`list-style`** shorthand for maximum efficiency.



Any Order: The values can be listed in any order.



Omit as Needed: You can use one, two, or all three values. The browser will use the default for any property you don't include (`disc``, `outside``, `none``).



Smart Fallback: If you specify both a ``type`` and an ``image``, the ``type`` will be used as a fallback if the image fails to load.

```
ul {  
  list-style: square url("custom-bullet.svg");  
}
```

This will show ``custom-bullet.svg``, but fall back to a simple ``square`` if the image is not available.