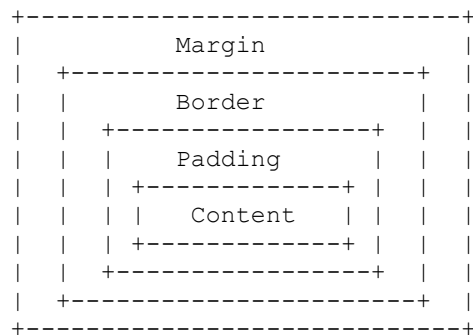


# CSS Box Model

## What is the Box Model?

- Every HTML element is treated as a **rectangular box** in CSS.
- The box consists of **4 layers**:
  1. **Content** → The actual text, image, or content inside.
  2. **Padding** → Space between content and border (inside the box).
  3. **Border** → Surrounds padding and content.
  4. **Margin** → Space between this element and other elements (outside the box).

## Box Model Structure



## Properties of Box Model

### 1. Content

```
p {  
  width: 200px;  
  height: 100px;  
}
```

### 2. Padding

```
p {  
  padding: 20px;           /* all sides */  
  padding: 10px 15px;      /* top-bottom | left-right */  
  padding: 5px 10px 15px 20px; /* top | right | bottom | left */  
}
```

### 3. Border

```
p {  
  border: 2px solid black;  
  border-radius: 10px; /* rounded corners */  
}
```

### 4. Margin

```
p {  
  margin: 20px;           /* all sides */  
  margin: 10px auto;      /* top-bottom | left-right (auto = center) */  
}
```

```
}
```

## Outline vs Border

- **Border** → part of the box (affects layout).
- **Outline** → drawn outside border (does not affect layout).

```
p {  
  border: 2px solid red;  
  outline: 2px dashed blue;  
}
```

## Box-sizing

- Controls how width and height are calculated.

```
p {  
  width: 200px;  
  padding: 20px;  
  border: 5px solid black;  
  box-sizing: content-box; /* default (width = content only) */  
}  
p {  
  box-sizing: border-box; /* width includes padding + border */  
}
```

# 6. Display and Positioning

## Display Property

The `display` property defines how an element is shown on the page.

### 1. Block

- Takes full width available.
- Starts on a new line.
- Examples: `<div>`, `<p>`, `<h1>`

```
div {  
  display: block;  
}
```

### 2. Inline

- Takes only required width.
- Does not start on a new line.
- Examples: `<span>`, `<a>`, `<strong>`

```
span {  
  display: inline;  
}
```

### 3. Inline-block

- Behaves like inline (same line) but allows **width & height**.

```
button {  
  display: inline-block;  
  width: 100px;  
  height: 40px;  
}
```

#### 4. None

- Hides the element (not displayed at all).

```
p {  
  display: none;  
}
```

### Visibility vs Display

- `display: none;` → element is removed from layout.
- `visibility: hidden;` → element is invisible but still occupies space.

## Position Property

The `position` property defines how an element is placed in the document.

#### 1. Static (default)

- Normal flow of the page.

```
div {  
  position: static;  
}
```

#### 2. Relative

- Positioned relative to its **normal position**.
- Can be moved using `top`, `left`, `right`, `bottom`.

```
div {  
  position: relative;  
  top: 20px;  
  left: 30px;  
}
```

#### 3. Absolute

- Positioned relative to the **nearest positioned ancestor**.
- If no ancestor, then relative to `<html>`.

```
div {  
  position: absolute;  
  top: 50px;  
  left: 100px;  
}
```

```
}
```

#### 4. Fixed

- Positioned relative to the **viewport** (browser window).
- Does not move when scrolling.

```
nav {  
  position: fixed;  
  top: 0;  
  width: 100%;  
}
```

#### 5. Sticky

- Acts like `relative` until a scroll point, then becomes `fixed`.

```
h1 {  
  position: sticky;  
  top: 10px;  
}
```

### Z-Index

Controls the **stack order** (which element is on top).

```
.box1 {  
  position: absolute;  
  z-index: 1;  
}  
.box2 {  
  position: absolute;  
  z-index: 2; /* on top of box1 */  
}
```

### Overflow

Defines what happens when content is larger than the container.

```
div {  
  width: 200px;  
  height: 100px;  
  overflow: hidden; /* hidden, scroll, auto, visible */  
}
```

### Example

```
<!DOCTYPE html>  
<html>  
<head>  
  <style>  
    .container {  
      width: 300px;  
      height: 200px;  
      border: 2px solid black;
```

```

        position: relative;
    }
    .box1 {
        width: 100px;
        height: 100px;
        background: lightblue;
        position: absolute;
        top: 20px;
        left: 30px;
        z-index: 1;
    }
    .box2 {
        width: 100px;
        height: 100px;
        background: lightgreen;
        position: absolute;
        top: 40px;
        left: 50px;
        z-index: 2;
    }
</style>
</head>
<body>
    <div class="container">
        <div class="box1">Box 1</div>
        <div class="box2">Box 2</div>
    </div>
</body>
</html>

```

## 7. Lists and Tables in CSS

### Styling Lists

Lists in HTML are of two main types:

- **Ordered List** (<ol>) → numbered list.
- **Unordered List** (<ul>) → bullet list.

#### 1. List Style Type

Defines the type of marker.

```

ul {
    list-style-type: square; /* disc, circle, square, none */
}
ol {
    list-style-type: upper-roman; /* decimal, lower-alpha, upper-roman */
}

```

#### 2. List Style Image

Use a custom image as a bullet.

```

ul {
    list-style-image: url("icon.png");
}

```

```
}
```

### 3. List Style Position

- `inside` → bullets appear inside content box.
- `outside` (default) → bullets outside content.

```
ul {  
  list-style-position: inside;  
}
```

### 4. Shorthand

```
ul {  
  list-style: square inside url("icon.png");  
}
```

## Styling Tables

HTML tables use `<table>`, `<tr>`, `<td>`, `<th>` tags.

### 1. Borders

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

### 2. Border Collapse

```
table {  
  border-collapse: collapse; /* collapse or separate */  
}
```

### 3. Cell Padding & Spacing

```
td {  
  padding: 10px;  
}  
table {  
  border-spacing: 15px; /* works only if border-collapse: separate */  
}
```

### 4. Text Alignment

```
th {  
  text-align: left; /* left, right, center */  
}  
td {  
  vertical-align: top; /* top, middle, bottom */  
}
```

### 5. Table Width and Height

```
table {  
  width: 100%;  
}  
td {  
  height: 50px;  
}
```

## 6. Table Background

```
th {
    background-color: lightgray;
}
td {
    background-color: #f9f9f9;
}
```

### Example

```
<!DOCTYPE html>
<html>
<head>
    <style>
        /* List Example */
        ul {
            list-style-type: square;
            padding-left: 20px;
        }
        ol {
            list-style-type: upper-roman;
        }

        /* Table Example */
        table {
            width: 80%;
            margin: 20px auto;
            border-collapse: collapse;
        }
        th, td {
            border: 1px solid black;
            padding: 10px;
            text-align: center;
        }
        th {
            background: lightblue;
        }
        tr:nth-child(even) {
            background: #f2f2f2;
        }
    </style>
</head>
<body>
    <h2>Styled List</h2>
    <ul>
        <li>Apple</li>
        <li>Banana</li>
        <li>Mango</li>
    </ul>

    <h2>Styled Table</h2>
    <table>
        <tr>
            <th>Name</th>
            <th>Age</th>
            <th>City</th>
        </tr>
        <tr>
            <td>John</td>
```

```
        <td>25</td>
        <td>New York</td>
    </tr>
    <tr>
        <td>Alice</td>
        <td>30</td>
        <td>London</td>
    </tr>
</table>
</body>
</html>
```