## CSS Padding

The CSS padding properties are used to generate space around an element's content, inside of any defined borders.

With CSS, you have full control over the padding. There are properties for setting the padding for each side of an element (top, right, bottom, and left), and a shorthand property for setting all the padding properties in one declaration.

---

Padding - Individual Sides

CSS has properties for specifying the padding for each side of an element:

- padding-top - sets the top padding of an element
- padding-right - sets the right padding of an element
- padding-bottom - sets the bottom padding of an element
- padding-left - sets the left padding of an element

All the padding properties can have the following values:

- *length* - specifies a padding in px, pt, cm, etc.
- *%* - specifies a padding in % of the width of the containing element
- inherit - specifies that the padding should be inherited from the parent element

**Example**

Set different padding for all four sides of a <div> element:

```
div {
  padding-top: 50px;
  padding-right: 30px;
  padding-bottom: 50px;
  padding-left: 80px;
}
```

---

### Padding - Shorthand Property

To shorten the code, it is possible to specify all the padding properties in one declaration.

The padding property is a shorthand property for the following individual padding properties:

- padding-top
- padding-right
- padding-bottom
- padding-left

Here is how it works:

If the padding property has four values:

- padding: 25px 50px 75px 100px;
  - top padding is 25px
  - right padding is 50px
  - bottom padding is 75px
  - left padding is 100px

### Example

Use the padding shorthand property with four values:

```
div {
  padding: 25px 50px 75px 100px;
}
```

If the padding property has three values:

- padding: 25px 50px 75px;
  - top padding is 25px
  - right and left paddings are 50px
  - bottom padding is 75px

**Example**

Use the padding shorthand property with three values:

```
div {
  padding: 25px 50px 75px;
}
```

If the padding property has two values:

- padding: 25px 50px;
    - top and bottom paddings are 25px
    - right and left paddings are 50px

Example

Use the padding shorthand property with two values:

```
div {
  padding: 25px 50px;
}
```

If the padding property has one value:

- padding: 25px;
    - all four paddings are 25px

Example

Use the padding shorthand property with one value:

```
div {
  padding: 25px;
}
```

---

## Padding and Element Width

The CSS width property specifies the width of the element's content area. The content area is the portion inside the padding, border, and margin of an element (the box model).

So, if an element has a specified width, the padding added to that element will be added to the total width of the element. This is often an undesirable result.

Example

Here, the <div> element is given a width of 300px. However, the actual width of the <div> element will be 350px (300px + 25px of left padding + 25px of right padding):

```
div {
  width: 300px;
  padding: 25px;
}
```

---

Padding and box-sizing

The box-sizing property defines how the width and height of an element are calculated: should they include padding and borders, or not.

The box-sizing property can have the following values:

- content-box - This is default. The width and height properties includes only the content (border and padding are not included)
- border-box - The width and height properties includes content, padding and border

So, to keep the width at 300px, no matter the amount of padding, you can use the box-sizing: border-box;. This causes the element to maintain its actual width; if you increase the padding, the available content space will decrease.

**Example**

```
div {
  width: 300px;
  padding: 25px;
  box-sizing: border-box;
}
```

## CSS Height, Width and Max-width

The CSS height and width properties are used to set the height and width of an element.

The CSS max-width property is used to set the maximum width of an element.

This element has a height of 70 pixels and a width of 100%.

---

CSS Set height and width

The height and width properties are used to set the height and width of an element.

The height and width do not include padding, borders, or margins. It sets the height and width of the area inside the padding, border, and margin of the element.

---

CSS height and width Values

The height and width properties can have the following values:

- auto - This is default. The browser calculates the height and width
- length - Defines the height or width in px, cm, em, etc.
- % - Defines the height or width in percent of the containing block
- initial - Sets the height or width to its default value
- inherit - The height or width will be inherited from its parent value

---

CSS height and width Examples

This element has a height of 200 pixels and a width of 50%

**Example**

Set the height and width of a <div> element:

```
div {
  height: 200px;
  width: 50%;
  background-color: powderblue;
}
```

This element has a height of 100 pixels and a width of 500 pixels.

Example

Set the height and width of another <div> element:

```
div {
  height: 100px;
  width: 500px;
  background-color: powderblue;
}
```

---

### CSS Using max-width

The max-width property sets the maximum allowed width of an element. This prevents the width of an element to be larger than the max-width property value.

The max-width property can have the following values:

- length - Defines the maximum width in px, cm, etc.
- *%* - Defines the maximum width in percent of the containing block
- none - This is default. Means that there is no maximum width

One problem with the width property can occur when the browser window is smaller than the width of the element. The browser then adds a horizontal scrollbar to the page. So, using max-width will improve the browser's handling on small windows.

### CSS max-width Examples

Drag the browser window to smaller than 600px wide, to see the difference between the two divs below!

Using width:

This element has a width of 600 pixels.

Using max-width:

This element has a max-width of 600 pixels.

**Example**

One <div> element with a max-width of 500 pixels, and one <div> element with a width of 500 pixels:

```
.div1 {
  max-width: 500px;
  background-color: powderblue;
}

.div2 {
  width: 500px;
  background-color: powderblue;
}
```

**Example**

A <div> element with both a width and a max-width property:

```
.div1 {
  width: 100%;
  max-width: 900px;
  background-color: powderblue;
}
```