The Padding Area

The CSS padding properties are used to generate space around content. What does that mean? We take a playful look at the padding area - with illustrations.

Assume you had a box of clothes.

You fold your clothes and have them kept neatly in this box.

I found you a good picture of the box. See below 😀.

The reader's box

What if for some reason you needed more space within the box?

You do NOT want the clothes to touch the edges of the box. You want the box to expand in size and create an empty space that separates the clothes from

the edges of the box.

How would you do this?

This is kinda difficult as boxes do NOT expand magically.

Now, back to CSS.

This box may represent any `html` element. Within this box we can choose to nest other elements, include texts or images. This represents the clothes inside the box.

Unlike the boxes in the real world, we can choose to create empty spaces within the CSS box!

This can be for many reasons such as taking advantage of white space and creating a breathing space within your design. To do this, **padding** to the rescue!

To include a padding within your element, aka *inner space*, use the **padding** property.

For example:

```
div {
    padding: 10px
}
```

This will add a `10px` padding on **all** sides, within the boundaries of the element.

To target specific sides witin the box, do this:

```
div {
    padding-left: 10px;
    padding-right: 10px;
    padding-bottom: 10px;
    padding-top: 10px;
}
```

In a much later lesson, we will take a second look at the `padding` shorthand. For now, this will suffice.

Exercise

Below is a `div` that contains the text, “I am an happy element”. Give this `div` a padding of `10px` on the left, `15px` on the right, `21px` at the top, and `18px` at the bottom. Use the long form, setting each direction explicitly.

Output

HTML

CSS (SCSS)

```
<html>
 <head>
   <title>Padding</title>
 </head>
 <body>
   <div>I am an happy element</div>
 </body>
</html>
```