Box Model
RECAP

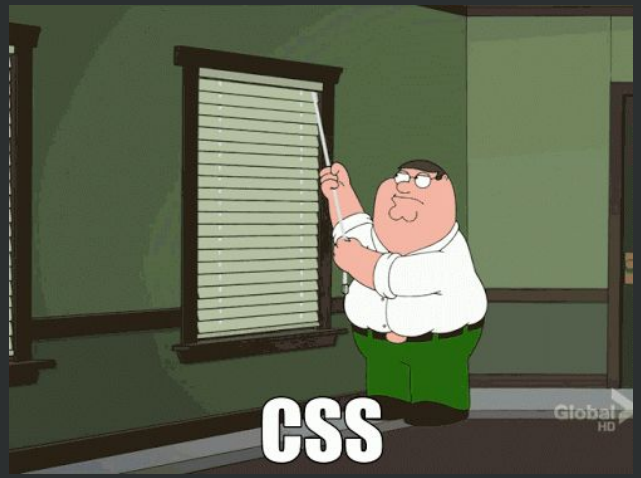
Summary

In this episode...

- The CSS Box Model
- Margin and Padding
- Normalizing styles
- Displaying element
- Box-shadow rule
- CSS Box Sizing

The CSS Box Model

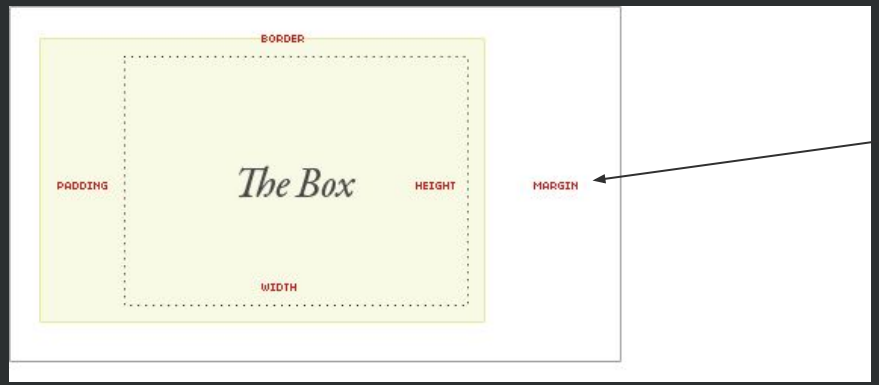
As you already know, **every element** in web design is a rectangular box.



IRON HACK

The CSS Box Model

This is what every box looks like,



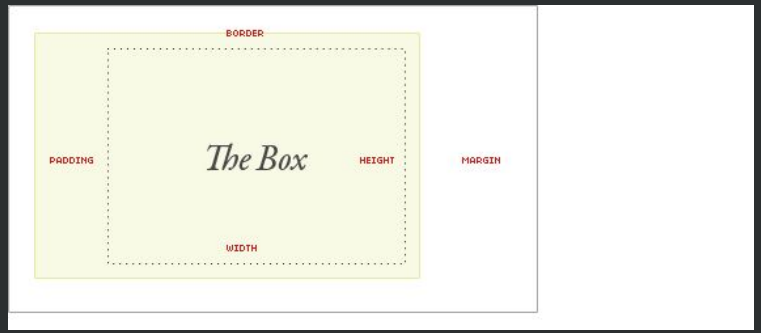
The Margin doesn't affect the size of the box itself, but it affects other content interacting with the box

We use `padding`, `border`, `height`, `width` *and* `margin` *attributes to specify the size and position of the box itself and its content*

IRON HACK

The CSS Box Model

The size of the box itself is calculated like this:



Width	width + padding-left + padding-right + border-left + border-right
Height	height + padding-top + padding-bottom + border-top + border-bottom

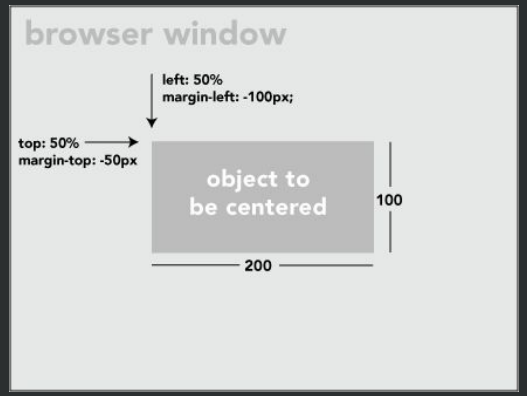
IRON HACK

The CSS Box Model

Margin & Padding

Margin

The margin property allows us to set the amount of space that surrounds an element. Margins for an element fall outside of any border and are completely transparent in color. Margins can be used to help position elements in a particular place.



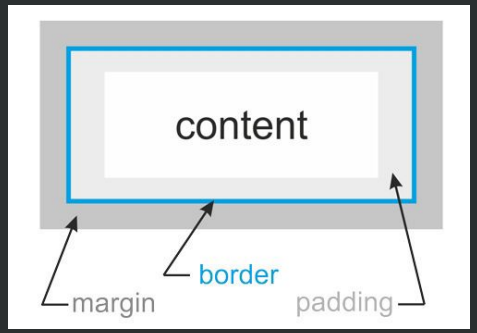
IRON HACK

The CSS Box Model

Margin & Padding

Padding

The padding property is very similar to the margin property; however, it falls inside of an element's border, should an element have a border. The padding property is used to provide spacing directly within an element.



IRON HACK

The CSS Box Model

WARNING! What happens if these values are undeclared?

The CSS Box Model

If `padding` or `borders` are undeclared, they are either zero (if you are using a CSS reset file like normalize.css) or the browser default value (probably **not** zero-especially on form elements that are commonly not reset).

The CSS Box Model

If the `width` of a box is undeclared (and the box is a block level element), things get a little weirder...

Let's start with that, and then move on to some other good-to-know stuff about the box model.

The CSS Box Model

Let's code

- Create a folder for this exercise. Let's call it '`boxmodel`' and create a new `boxmodel.html` file and a `boxmodel.css` file
- In your html file, add this code to initialize your file, link the css and create a new div

```
<!DOCTYPE html>
<html>
<head>
      <meta charset="UTF-8">
      <link rel="stylesheet" type="text/css" href="boxmodel.css">
</head>
<body>
      <div class="box"></div>

</body>
</html>
```

The CSS Box Model

Let's code

- Set some attributes in your css for your class `'box'`

```
.box {
    height: 150px;
    background: #654678;
}
```

We need to set the `height` *and* `background` *because without those attributes we couldn't see our div (but we are not setting the* `width`*)*

To width or not to width

To make the width remain 100% in width and the padding and border pushing inwards instead of outward, you should declare a width, and the box needs a static or relative positioning. But if you *explicitly set* the width of the box to be 100%, the padding will push the box outward as normal.

*The default width of a box isn't really 100% but a less tangible "whatever is left". This is particularly valuable to know, since there are lots of circumstances where it is immensely useful to either **set** or **not set** a width.*

Click here and check it out at

The CSS Box Model

Let's code

- Create a **new div** inside the box with class content. Set a proper height (smaller than the box) and a different background so you can see the changes.

- Change the `width` of the box and the content, the `padding` and the `margin` attributes to see the differences in the layout.

- Try to set `content`'s `width` to `100%` and see how it affects `padding` and `margin`.

The CSS Box Model

Displaying elements

Every element has a default display property value; however, as with all other property values, that value may be overwritten.

There are quite a few values for the display property, but the most common are `block`, `inline`, `inline-block`, and none.

We can change an element's display property value by selecting that element within CSS and declaring a new `display` property value.

The CSS Box Model

Displaying elements

Block

A value of block will make that element a block-level element.

The CSS Box Model

Displaying elements

Inline

A value of inline will make that element an inline-level element.

The CSS Box Model

Displaying elements

Inline-block

Using this value will allow an element to behave as a block-level element, accepting all box model properties. However, the element will be displayed in line with other elements, and it will not begin on a new line by default.

The CSS Box Model

Let's play a little bit with display attributes:

Click here and
check it out at

The CSS Box Model

The box-shadow

The box-shadow property describes one or more shadow effects as a comma-separated list.

It enables you to cast a drop shadow from the frame of almost any element.

If a border-radius is specified on the element with a box shadow, the box shadow takes on the same rounded corners.

The CSS Box Model

The box-shadow - Let's code

- In the same exercise, add this line to your css. The properties of box shadow in this line are specified in this order: offset-x, offset-y, blur-radius, spread-radius, color.

```
.box                                                    {
    box-shadow: 2px 2px 2px 1px rgba(0, 0, 0, 0.2);
}
```

CSS3 Box Sizing

One of the hardest parts about layout with CSS is the relationship of width and padding.

Instead of adding the padding or border to the inside of the element, the majority of modern browsers add the padding/border to the outside, increasing the width/height of your element by the padding/border value.

CSS3 Box Sizing

The box-sizing property was introduced in CSS3. It has three possible values (content-box, padding-box and border-box) and the most popular value is **border-box**.

It is used to alter the default CSS box model used to calculate widths and heights of elements.

CSS3 Box Sizing

CSS3 Box Sizing

With box-sizing: border-box we can change the box model to what was once the "quirky" way, where an element's specified `width` and `height` aren't affected by `padding` or `borders`.

This has proven so useful in responsive design that has found its way into reset styles.

CSS3 Box Sizing

Let's code

- *Let's create a navigation menu for our exercise*

HTML:

```
<ul>
    <li><a href="#">Menu 1</a></li>
    <li><a href="#">Menu 2</a></li>
    <li><a href="#">Menu 3</a></li>
    <li><a href="#">Menu 4</a></li>
    <li><a href="#">Menu 5</a></li>
</ul>
```

Let's code

- *Let's add some style in our css file*

First, we take care of our nav bar, centering at the top of the page, floating every element of the list and removing the bullets

```
nav {
    width: 500px;
    margin: 50px auto 0;
    height: 50px;
}
nav ul {
    padding: 0;
    margin: 0;
}
nav li {
    float: left;
    list-style-type: none;
}
```

Let's code

- *Let's add some style in our css file*

We will give some style to the links of our menu. Setting width and height, removing the blue links and underlined text.
Also, we will set the height of the 'buttons' and centering the text.

```
nav a {
    display: inline-block;
    width: 100px;
    height: 100%;
    background-color: #ccc;
    color: #fff;
    text-decoration: none;
    line-height: 300%;
    text-align: center;
}
```

Let's code

- *Let's add some style in our css file (li elements with different background colors)*

We are changing the background for each link menu to see the difference between them.
We will group them in pairs and odds.

```
nav li:nth-child(1n) a {
    background-color: #3C536F;
    border-left: 0;
}
nav li:nth-child(2n) a {
    background-color: #6EA634;
}
```

Let's code - Test

- *The problem will come when you try to add some borders (or padding) to your links. Add these rules to your file and see the changes:*

```
nav a {
    border-left: 1px solid #aaa;
    border-right: 1px solid #f3f3f3;
}
```

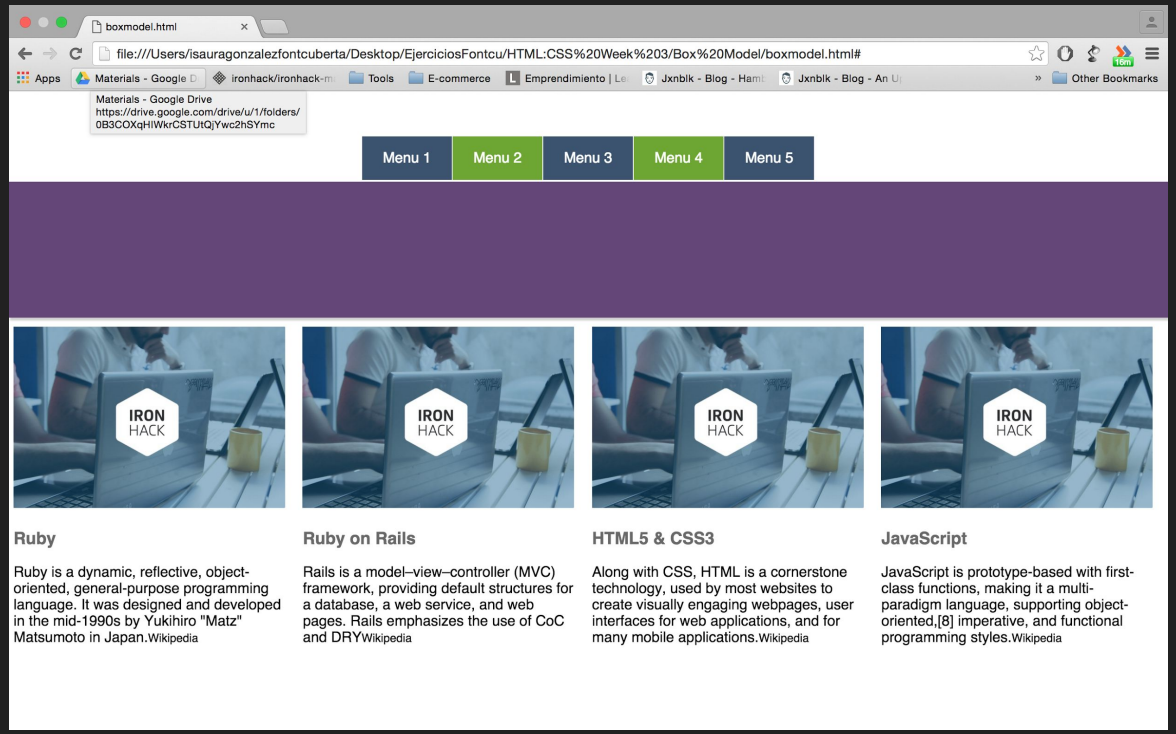
Let's code - The Solution

```
nav a {
    -moz-box-sizing: border-box;
    -webkit-box-sizing: border-box;
    box-sizing: border-box;
}
```

IRON HACK

Exercise

With the documents you created, try to reach this result:



Download Resources

IRON HACK