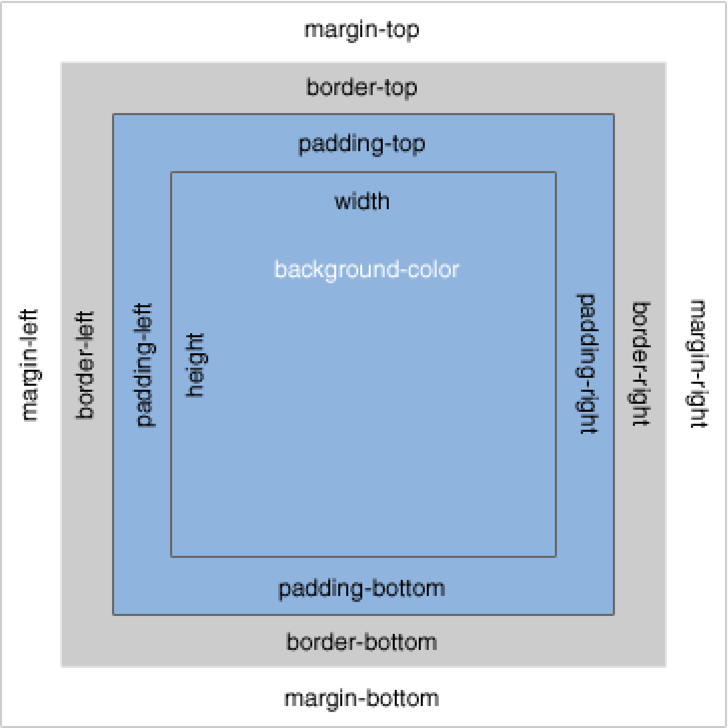
SchoolNova

IT101

CSS – The Box Model

The Box Model

- Every HTML element is a rectangular box.
- Box properties are Height, Width, Padding, Border and Margin
- HTML tags that are used for custom box definition:
    - **<DIV>** - The HTML <div> element is a block level element that can be used as a container for grouping other HTML elements. The <div> element has no special meaning. Except that, because it is a block level element, the browser will display a line break before and after it. When used together with CSS, the <div> element can be used to set style attributes to large blocks of content. Another common use of the <div> element, is for document layout. It replaces the "old way" of defining layout using tables. Using <table> elements for layout is not the correct use of <table>. The purpose of the <table> element is to display tabular data.
    - **<SPAN>** - The HTML <span> element is an inline element that can be used as a container for text. The <span> element has no special meaning. When used together with CSS, the <span> element can be used to set style attributes to parts of the text.



Border, Margin and Padding

- The CSS border properties allow you to specify the style and color of an element's border.
- The **border-style** property specifies what kind of border to display. Existing styles: dotted, dashed, solid, double, groove, ridge, inset, outset and none.
- The **border-width** property is used to set the width of the border.
- The **border-color** property is used to set the color of the border.
- To shorten the code, it is also possible to specify all the individual border properties in one property. This is called a shorthand property. The border property is a shorthand for the following individual border properties:
  - border-width
  - border-style (required)
  - border-color
- Example: border: 5px solid red;
- **Margin** - The margin clears an area around an element (outside the border). The margin does not have a background color, and is completely transparent.
- The top, right, bottom, and left margin can be changed independently using separate properties. To shorten the code, it is possible to specify all the margin properties in one property. This is called a shorthand property.
- Example: margin:25px 50px 75px 100px;
- **Padding** - The padding clears an area around the content (inside the border) of an element. The padding is affected by the background color of the element.
- The top, right, bottom, and left padding can be changed independently using separate properties. A shorthand padding property can also be used, to change all paddings at once.
- Example: padding: 25px 50px 75px 100px;

ID, Class, Grouping and Nesting

- The **id** selector is used to specify a style for a single, unique element. Example:

```
#block101 {
    text-align:center;
    color: #FF0000;
}
```

- The **class** selector is used to specify a style for a group of elements. Unlike the id selector, the class selector is most often used on several elements. Example:

```
.centered_content {text-align:center;}
```

- **Grouping** - In style sheets there are often elements with the same style. To minimize the code, you can group selectors. Separate each selector with a comma. Example:

```
h1, h2, p {
    color: #00FF00;
}
```

- In addition to setting a style for a HTML element, CSS allows you to specify your own selectors called "id" and "class".

- **Nesting** - It is possible to apply a style for a selector within a selector. In the example below, one style is specified for all p elements, one style is specified for all elements with class="marked", and a third style is specified only for p elements within elements with class="marked":

```
P {
    color:blue;
    text-align:center;
}
.marked {
    background-color: #FF0000;
}
.marked p {
    color: #FFFFFF;
}
```

Reference: http://www.w3schools.com/css

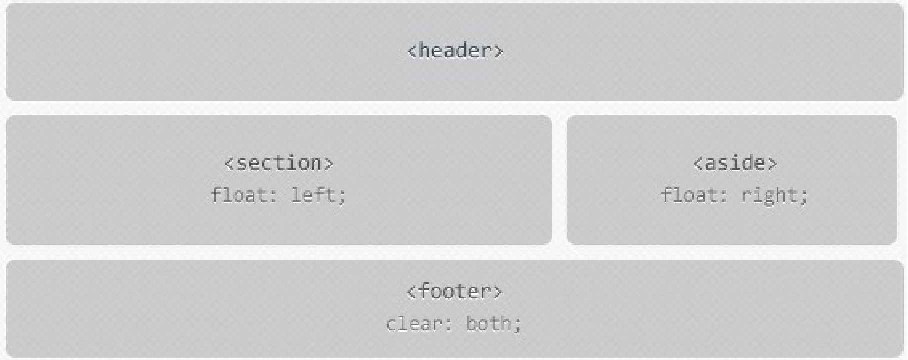
Floating Elements

- One way to position elements along side one another is to use the **float** property. The float property allows you to take elements and float them right or left, positioning them directly next to or opposite each other.

- Whenever an element is floated, it breaks the normal flow of a page and other elements will wrap around the floated one as necessary. The **clear** property acts on the element it is applied to and returns the document back to its normal flow, clearing every floated element up to that point.

- Example:

```
section {
  float: left;
  margin: 10px;
  width: 600px;
}
aside {
  float: right;
  margin: 10px;
  width: 320px;
}
footer {
      clear: both;
}
```



Homework

- Design a layout using the box model and floating elements.
- Take a look at sample page layouts and designs at http://csszengarden.com, you may want to create something similar, or if you have not finished the Facebook exercise from the previous homework, continue working on it.
- Create a page (or several pages) using your layout. Use classes, ID's, <span> and <div> tags. Make sure your page has image(s) and text. Do not use tables for layout.
- Test your page from different viewports: desktop and mobile, and different browsers. Correct as necessary for optimal display.