CSS Layout

INFOCUS

CSS isn’t just used to control the appearance of HTML elements, it’s also use to control their position. By using CSS to set position properties on container elements you can create sophisticated page layouts including centered content, multiple columns and grids.

In this session you will:

- ✓ gain an understanding of block and inline elements
- ✓ gain an understanding of the box model
- ✓ learn how to add a content container
- ✓ gain an understanding of the box model
- ✓ learn how to align text and images using the float property
- ✓ learn how to create a fixed width column layout
- ✓ learn how to create a comments field.

Block And Inline Elements

Before you begin creating CSS layouts, it's best to spend some time understanding how HTML is laid out by your browser, starting with block and inline elements and the box model. All HTML elements are either block elements or inline elements by default, which affects how they fit into the flow of the HTML document. The box model is important to understand to properly size elements.

Types Of Elements

Block elements are displayed on their own line and take up the full width of the page. Examples of block elements are headings (<h1> to <h6>), paragraphs (<p>), lists (<ul> and <ol>) and divs (<div>).

```
<p>Paragraphs take up the full width of the page.</p>
```

Inline elements are displayed on the same line as other elements and take up only as much space as they need. Examples of inline elements are links (<a>), emphasis (<em>), images (<img>) and spans (<span>).

```
<p>Links only take up as much <a href="Destination.html">space as they need</a>.</p>
```

Divs And Spans

While elements such as headings and links impart some meaning to their contents, <div> and <span> are mostly used to group elements for CSS styling. As spans are inline, they are used mostly to group text and images for formatting, such as making text a different colour or underlining it. Divs, on the other hand, are used to group related elements into containers which can be positioned in a great number of ways using CSS.

The Display Property

All HTML elements are displayed by default as either block or inline, but you can override this using the CSS ***display*** property.

This is handy, for example, when creating a horizontal list. List item (<li>) elements have display set to block by default, which means that each item will be displayed on its own line. By setting <li> elements to *display: inline*, all of the items will be displayed on the same line.

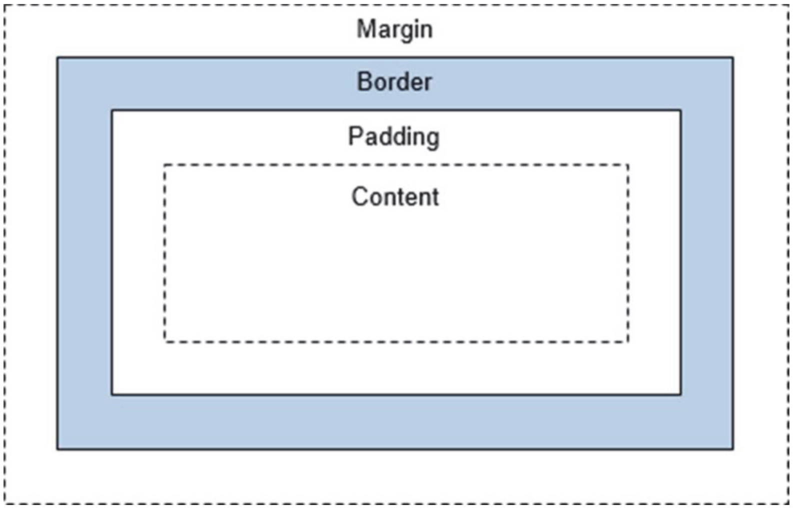
You can also set the display property to *none* to remove the element from the HTML layout entirely.

THE BOX MODEL

Every HTML element can be considered as a box, with margins, a border and content. The box model is used to describe the way the element's margins, border and content affect the size of the element. Historically this has sometimes been difficult to calculate and so the CSS box-sizing property was introduced.

The Box Model

The ***box model*** is used to describe design and layout in CSS. The name ***box model*** is derived from the fact that every element on your web page (headings, list items, paragraphs etc.) exists in a rectangular box, which can be visualised with the following diagram:



The ***content*** is where the text and images for the element appear. The ***padding*** is a transparent area of space around the content. The ***border*** is a border that goes around the padding and content. The ***margin*** is an area of space around the border which separates the element from other elements.

Element Width And Height

By default, when you set the height and width of an element using CSS, you are setting the height and width of the content only. Any padding, borders and margins applied will add to the total height and width of the element. So if you set the width of an element to 50% of the page and it has a margin of 10px and a border of 1px, the total element width will actually be 50% plus 22px (10px margin on the left, 1px border on the left, 50% for content, 1px border on the right and 10px margin on the right). As you can see, this can be quite confusing when you expect your 50% wide element to take up half the page.

The Box-Sizing Property

Setting the CSS ***box-sizing*** property on an element to the value of ***border-box*** means that setting its height and width will set the total height and width. This is a very handy trick to know when laying out elements with CSS and means that your element with a width of 50% will be exactly 50%.

Adding a Content Container

Without any layout options set, your paragraphs will stretch across the full width of the screen. This becomes a problem on wide screens, where the user is forced to move their eyes all the way from the left of the screen to the right. To reduce reader fatigue and make this a nicer experience, we can set the maximum width of the content and center it in the browser window.

Try This Yourself:

Open File *Before starting this exercise you MUST open the file CSS Layout_1.html...*

1. Open the file in your web browser and make it as wide as possible

   The text stretches all the way across the screen…

2. In your text editor, add a ***<div>*** element with ***id="container"*** around all of the elements in the document body

3. Save the changes to the file

4. Open the ***CSSLayout_1.css*** file in your text editor

5. Add the ***div#container*** style as shown

   The max-width property specifies the maximum width the container can be. Setting margin-left and margin-right to auto centers the container

6. Open the file in your web browser to display the page

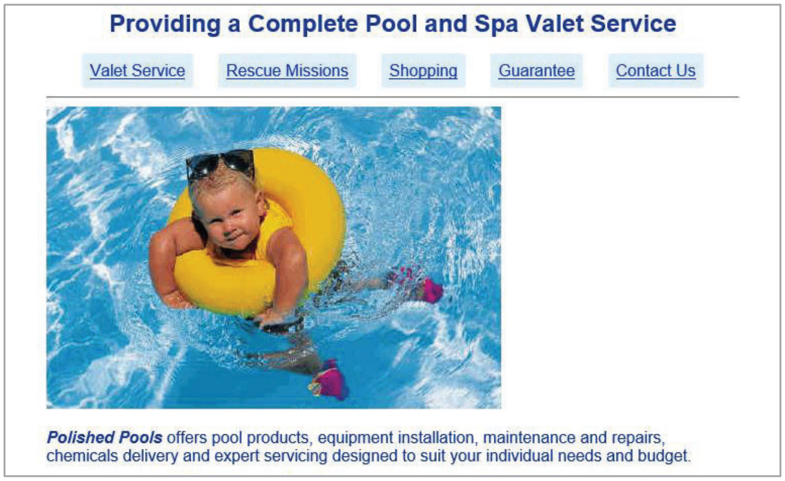
   The text should now be 1000px wide and centered

```
<!-- Created for Polished Pools by PP Web Design -->
<!DOCTYPE html>
<html>
<head>
  <title>Polished Pools Valet Service</title>
  <link rel="stylesheet" type="text/css"
href="CSSLayout_1.css">
</head>
<body>
  <div id="container">
    ...
  </div>
</body>
</html>
```

2

```
div#container {
  max-width: 1000px;
  margin-left: auto;
  margin-right: auto;
}
```

4



Polished Pools offers pool products, equipment installation, maintenance and repairs, chemicals delivery and expert servicing designed to suit your individual needs and budget.

6

For Your Reference…

To ***add a content container***:

1. Add a ***<div>*** around the body content
2. Set the ***max-width*** CSS property for the div to the maximum width it should be
3. Set the ***margin-left*** and ***margin-right*** CSS properties for the div to ***auto***

Handy to Know…

- For most comfortable reading, it's recommended that your lines should be at most between 50 to 75 characters in length. This is very subjective, though, so it's best to experiment with your layouts to find the best line length.

UNDERSTANDING THE FLOAT PROPERTY

By default, when block elements are added to a page they are automatically stacked on top of each other. By applying the CSS ***float*** property to these elements, you can arrange them side-by-side and create a page with much more visual interest and practicality. The float property is used to create multi-column layouts and wrap text around images.

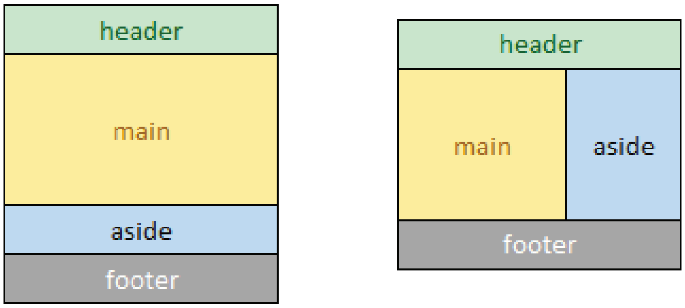
Using The Float Property

Most complex CSS-based layouts are float-based, which means that the elements on the page are floated to achieve the necessary element positioning. An unstyled document uses normal document flow for positioning – that is, its block elements appear stacked on top of one another, in the order that they are encountered in the code.

For instance, you may create a web page with four main elements: ***header***, ***main***, ***aside*** and ***footer***. These elements appear ***stacked*** on top of each other in this order. Wrapped within these elements, it is possible to have other elements and these will also adopt a stacked order, according to their position within the code. The height of the element will be automatically calculated from the height of its content.

To enable you to create a customised page layout, you are able to ***float*** elements. By adjusting the ***float*** property, you change the document flow. Elements can be floated to the left or the right and following elements move up to occupy vacated space. You can also float items within elements - if you float an image within an element to the right for instance, text within the same element will wrap around the left and bottom edges of that image.

In the example below, the ***width*** of the ***main*** element has been reduced and the element has been floated to the left. The ***aside*** element has also been reduced in width and then floated to the right. In this way, we can arrange elements alongside each other:



Clearing Floated Elements

In some instances, you might need to force floated elements to no longer affect the positioning of the elements following them, such as when you need to start a new line or when floated elements are overlapping each other. In this case, you can use the ***clear*** property to prevent elements from occupying the space to the left or the right of an element.

The ***clear*** property has three values: ***left***, ***right*** and ***both***. By clearing to the ***left*** or ***right***, you prevent anything floating to the left or right (respectively) of the cleared element. Clearing ***both*** prevents elements floating on either the left or the right of the cleared element.

Aligning Text And Images

By default, image elements take up their own line so that subsequent text is displayed below them. If you need images to be displayed next to text, you can use the ***float*** property in CSS in conjunction with the image. Images can have a float value of ***left***, where the image is on the left and text flows around it to the right, or ***right***, where the image is on the right of the text.

Try This Yourself:

Open File

Before starting this exercise you MUST open the file CSS Layout_2.html…

1. In your text editor, scroll down to the position as shown and add the code for the image as shown
2. Save the changes to the file
3. Add the image style as shown to the corresponding CSS file

   If you are using the CSS Layout_2.html file, use the CSS Layout_2.css file. If you are continuing on with the CSS Layout_1.html file, use the CSS Layout_1.css file…
4. Save the changes to the CSS file
5. Switch to your web browser and refresh the page to display the image

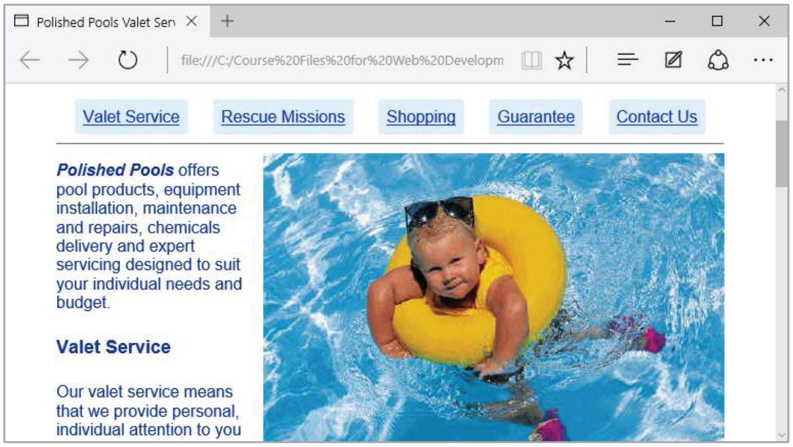
   The image is right-aligned, and there are 20 pixels between the text and the image

```
<a href="#guarantee">Guarantee</a>
    <a href="mailto:ppools@example.com">Contact Us</a>
  <p>
<hr />
  <img class="main-image" src="ChildInPool.jpg" alt="Child
enjoying pool" />
  <p><b><i>Polished Pools</i></b> offers pool products,
equipment installation, maintenance and repairs, chemicals
delivery and expert servicing designed to suit your
individual needs and budget.</p>
```

1

```
img.main-image {
  float: right;
  margin-left: 20px;
}
```

3


Polished Pools Valet Serv
file:///C:/Course%20Files%20for%20Web%20Developm
Valet Service
Rescue Missions
Shopping
Guarantee
Contact Us

Polished Pools offers pool products, equipment installation, maintenance and repairs, chemicals delivery and expert servicing designed to suit your individual needs and budget.

Valet Service

Our valet service means that we provide personal, individual attention to you

5

For Your Reference…

To ***align text*** and ***images***:

- Optionally give the image an ID or class name to distinguish it from other images in the page
- Add the selector to your style sheet
- Add the ***float*** property with a value of ***left*** or ***right***

Handy to Know…

- To stop elements such as paragraphs from flowing around a floating element, you can use the ***clear*** property in CSS.

FIXED WIDTH COLUMN LAYOUT

Block elements such as divs usually take up a whole line, but we can allow them to be positioned next to each other using the CSS ***float*** attribute. This enables you to create multi-column layouts. Often you will want one of these columns to be a fixed width (specified in pixels, for example) and the other column to fill the remaining space. You can do this using float and margin styles.

Try This Yourself:

Same File

Continue using the previous file with this exercise, or open the file CSS Layout_3.html in the text editor and browser...

Let's move the opening hours table to the left of the main content as it is too important to be at the bottom of the page…

1. In your text editor, add a ***<div>*** element with ***id="opening-hours"*** to the position as shown and cut and paste the opening hours table into it
2. Create another ***<div>*** element underneath with ***id="content"*** to contain the remaining body of the document
3. Save the changes to the file
4. Add the code for the opening-hours class and content class as shown to the the ***CSSLayout_3.css*** file (if using ***CSS Layout_3.html*** but if you are using an earlier exercise file use the corresponding CSS file) and save the changes
5. Switch to your web browser, then click on ***Refresh*** to display the new layout

```
  <p style="text-align: center">
      <a href="#valet" class="menu">Valet Service</a>
      <a href="#rescue" class="menu">Rescue Missions</a>
      <a href="#shopping" class="menu">Shopping</a>
      <a href="#guarantee" class="menu">Guarantee</a>
      <a href="mailto:ppools@example.com"
class="menu">Contact Us</a>
  <p>
  <hr />
  <div id="opening-hours">
    <h4>Our Opening Hours</h4>
    <table id="openinghours">
      <tr>
        <th>Day</th>
        <th>Open</th>
      </tr>
      ...
    </table>
  </div>
```

Type this

Cut and paste this from the bottom of the page

Type this

1

```
      </table>
    </div>
    <div id="content">
      <img class="main-image" src="ChildInPool.jpg"
alt="Child enjoying pool" />
     ...
    </div>
  </div>
</body>
</html>
```

2

```
div#opening-hours {
  float: left;
  width: 200px;
}

div#content {
  margin-left: 220px;
}
```

4

For Your Reference…

To ***use float*** to ***position elements***:

1. Add the CSS property ***"float: left"*** to the div on the left
2. Set a CSS ***width*** for the div on the left
3. Set the CSS ***margin-left*** property on the div on the right to the value of the width for the div on the left

Handy to Know…

- You can use fixed width column layouts with the fixed width column on the left or on the right.

PERCENTAGE WIDTH COLUMN LAYOUT

Block elements such as divs usually take up a whole line, but we can allow them to be positioned next to each other using the CSS ***float*** attribute. This enables you to create multi-column layouts. Often you will want these columns to take up a certain percentage of the screen and adjust their width with different screen sizes. You can do this using divs floated next to each other.

Try This Yourself:

Open File *Before starting this exercise you MUST open the file CSS Layout_4.html…*

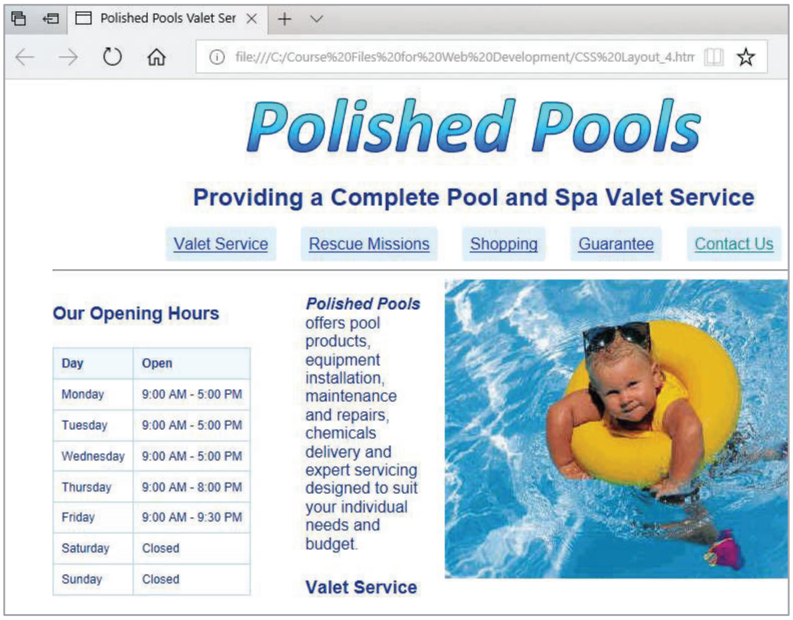
1. Change the code in the ***CSSLayout_4.css*** file as shown (if using ***CSS Layout_4.html***, but if you are using an earlier exercise file use the corresponding CSS file) and save the changes
2. Switch to your web browser then click on ***Refresh*** to display the new layout

The opening hours column should now take up 30% and the content column 70%. Try resizing the browser to see how this stays the same at various screen sizes

```
div#opening-hours {
  float: left;
  width: 30%;
}

div#content {
  float: left;
  width: 70%;
}
```

1



Polished Pools Valet Ser

file:///C:/Course%20Files%20for%20Web%20Development/CSS%20Layout_4.htm

Polished Pools

Providing a Complete Pool and Spa Valet Service

Valet Service | Rescue Missions | Shopping | Guarantee | Contact Us

Our Opening Hours

Day	Open
Monday	9:00 AM - 5:00 PM
Tuesday	9:00 AM - 5:00 PM
Wednesday	9:00 AM - 5:00 PM
Thursday	9:00 AM - 8:00 PM
Friday	9:00 AM - 9:30 PM
Saturday	Closed
Sunday	Closed

Polished Pools offers pool products, equipment installation, maintenance and repairs, chemicals delivery and expert servicing designed to suit your individual needs and budget.

Valet Service

3

For Your Reference…

To ***use float*** to ***position elements***:

1. Add the CSS property ***"float: left"*** to the div on the left
2. Set a CSS ***width*** for the div on the left
3. Add the CSS property ***"float: left"*** to the div on the right and set its CSS width

Handy to Know…

- You can set the width of two divs to 50% to have each take up a half of the page, or the width of three divs to 33% to have each take up a third. If you start to put borders, margins or padding on the divs, you will have to set the CSS ***box-sizing*** property of the divs to ***border-box***.