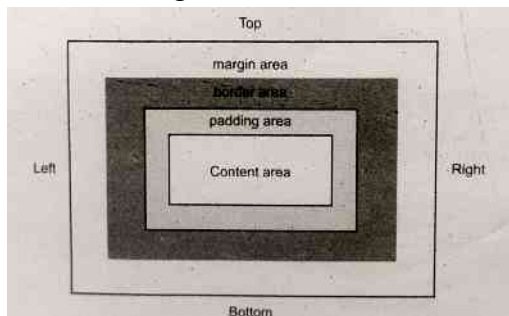# NIELIT GORAKHPUR

Cascading Style Sheets

The Box Model

CSS treats an HTML document as a hierarchical tree of elements, where each element can have zero or more child elements arranged in an ordered Way. The topmost element of this tree is called as the root element or the parent element. These elements display their content at a specific position, which is defined by using CSS properties. CSS converts the data of HTML elements in the form rectangular boxes, by using a layout model, called the box model, to set the design and layout of HTML documents. This means that the box model determines how HTML elements are displayed as boxes. The box model allows placing a border around the elements and also provides space between elements.

Figure shows the various areas and edges of a box:

The following are the areas of a box, as shown in Figure:

- **Content area** - Displays the content of a document, such as text and images. This is bounded by a rectangle, which is called as the content edge. Note that the content area always appears; inside the padding area.
- **Padding area** - Specifies the area around the content area. This is bounded by the padding edge. Outside the padding is the border area and the outside boundary of that area is the border edge. Finally, outside the border is the margin area whose outer edge is called the margin edge,
- **Border area** - Specifies the area around the padding area. This is bounded by the border edge.
- **Margin area** - Specifies the area around the border area. This is bounded by the margin edge.
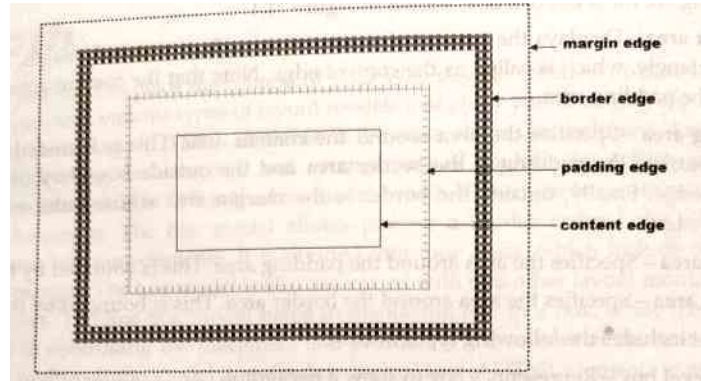
A box model includes the following types of boxes:

- **block-level box** - Represents a box to show a paragraph.
- **line box** - Represents a box to show a line of text.
- **inline-level box** - Represents a box to show the words of a line.

Note: *A block-level box can contain either other block-level boxes, such as a section containing paragraphs or a table containing tows, or fine boxes, such as a paragraph containing lilies of text. A line box contains inline-level boxes, such as a fine with words in different styles. An inline-level box can contain either text with other inline-level boxes or a block-level box, such as a small table that is in inline format.*

Exploring Box Dimensions

All HTML elements in a box model are represented as rectangular boxes. The dimensions of the box model are calculated by using the height and width of the content area that gets applied to the element. Each box is associated with a content area and many optional areas, such as padding, border, and margin. The size of each area is specified by using the box model dimensions shown in Figure:

The perimeter of the content, padding, border, and margin is called an edge. Each box in the box model has the following four edges:

- **The content edge** - Surrounds a rectangle specified by the width and height of the box. This is also called the inner edge. The four content edges represent the content area of the box.
- **The padding edge** - Surrounds the padding box. If the padding has 0 width, the padding edge becomes a content edge. The four padding edges represent the padding area of the box.
- **The border edge** - Surrounds the border of the box. If the border has 0 width, the border edge becomes a padding edge. The four border edges represent the border area of the box.
- **The margin edge** - Surrounds the margin of the box. If the border has 0 width, the margin edge becomes a border edge. The margin edge is also called as outer edge. The four border edges represent the margin area of the box.

Exploring Padding Properties

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

Padding (space) in the box model specifies the distance between the border of an element and the content within it. The padding is affected by the background color of an element. The value of padding cannot be negative. The shorthand property in padding is used to change all the padding properties, such as **padding-top**, **padding-bottom**, **padding-right**, and **padding-left** at once. The syntax used to set padding properties is as follows:

.class1 {padding: value or [top-value right-value bottom-value left-value] ;}

Padding can be set according to the following values:

- **Length** - Specifies fixed padding in the pt and em units
- **Percentage(%)** - Specifies padding in percentage with respect to the width of a parent block

- **Auto** - Specifies default padding from the top, bottom, left, or right direction

Note: Negative values are not allowed.

The following code snippet shows the specifications of padding properties in the box model:

```
div {
  padding: 70px;
  border: 1px solid #4CAF50;
}
```

Individual padding properties of edges can also define as following:

```
#container {
        padding-top: 7px;
        padding-left: 25%;
        padding-right: auto;
        padding-bottom: 4Spx;
        border: 2px solid 000000;
}
```

```
<!DOCTYPE html>
<html><head><style>
.div1 {  padding: 70px;
border: 1px solid #4CAF50;
}
.div2 {
  border: 1px solid black;
  background-color: lightblue;
  padding-top: 50px;
  padding-right: 30px;
  padding-bottom: 50px;
  padding-left: 80px;
}
.div3 {
  border: 1px solid black;
  padding: 25px 50px 75px 100px;
  background-color: lightblue;
}
</style></head>
<body>
<div class="div1">This element has a padding of
70px.</div>
<div class="div2">This div element has a top padding
of 50px, a right padding of 30px, a bottom padding
of 50px, and a left padding of 80px.</div>
<div class="div3">This div element has a top padding
of 25px, a right padding of 50px, a bottom padding
of 75px, and a left padding of 100px.</div>
</body></html>
```

This element has a padding of 70px.

This div element has a top padding of 50px, a right padding of 30px, a bottom padding of 50px, and a left padding of 80px.

This div element has a top padding of 25px, a right padding of 50px, a bottom padding of 75px, and a left padding of 100px.

Exercise:

1: Write short note on padding property.