



SNS COLLEGE OF TECHNOLOGY

Coimbatore-35
An Autonomous Institution

Accredited by NBA – AICTE and Accredited by NAAC – UGC with 'A+' Grade
Approved by AICTE, New Delhi & Affiliated to Anna University, Chennai



DEPARTMENT OF INFORMATION TECHNOLOGY

WEB TECHNOLOGY

III YEAR - V SEM

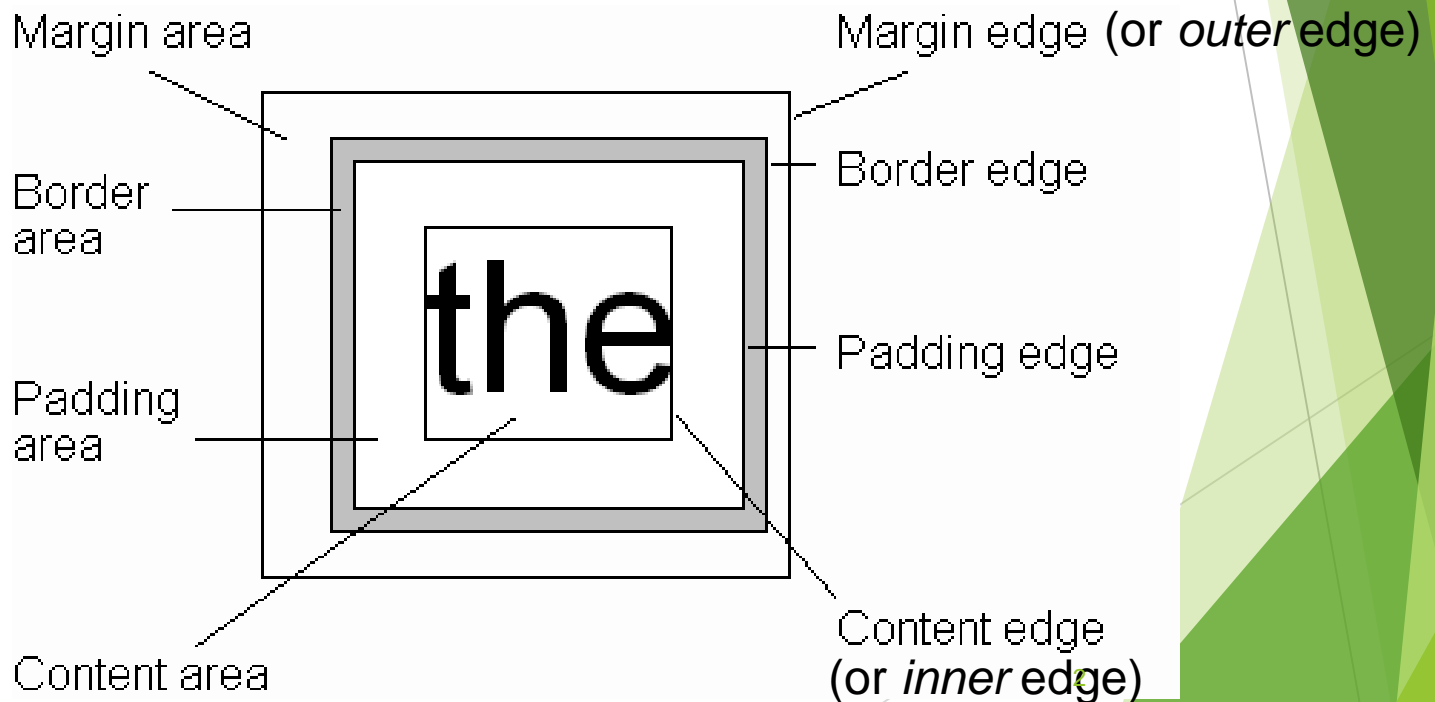
UNIT 2 – CSS AND CLIENT-SIDE SCRIPTING

TOPIC 4 - Box Model Normal Flow Box Layout Beyond the Normal Flow-CSS3.0



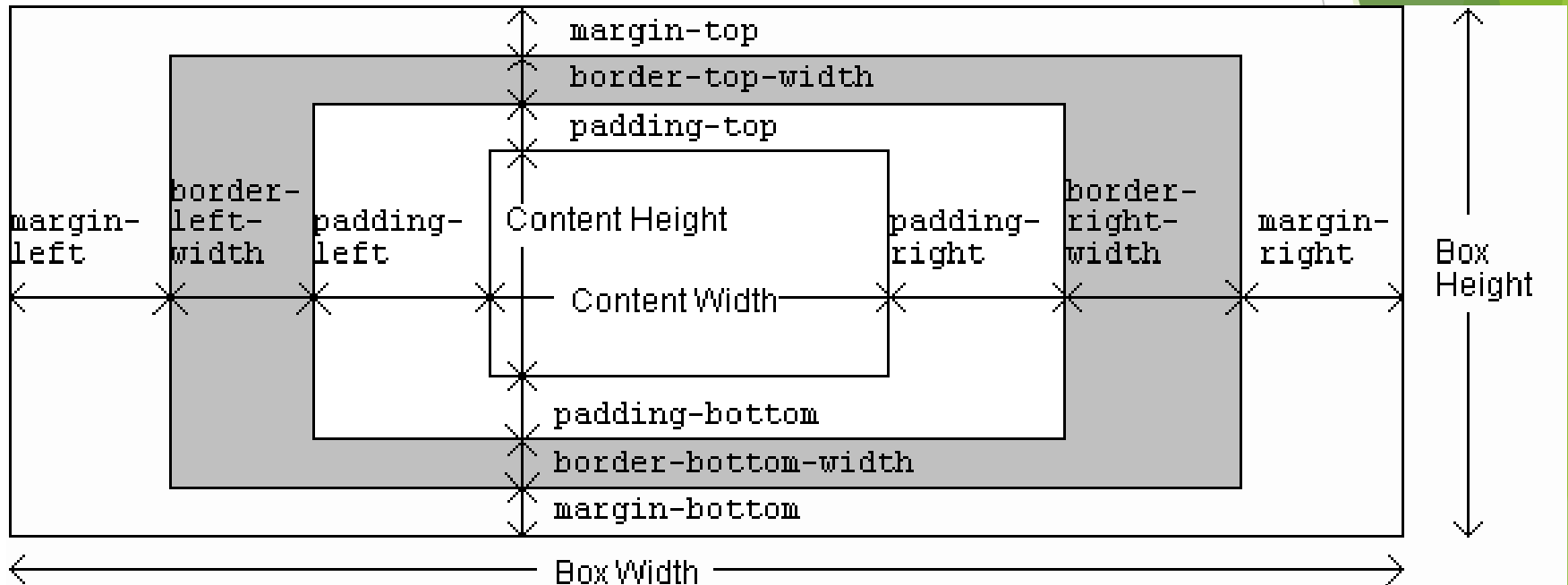
CSS Box Model

- ▶ Every rendered element occupies a box:





CSS Box Model

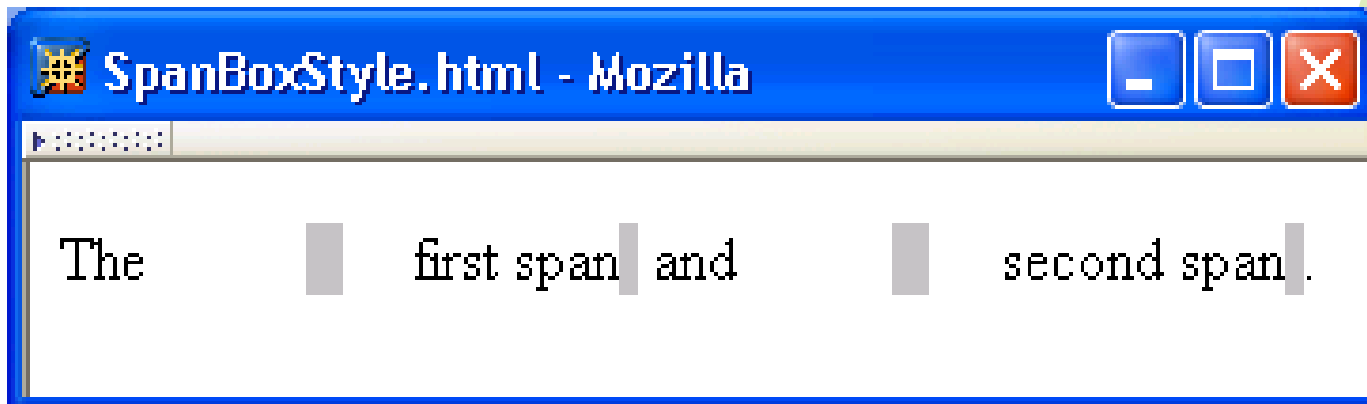




CSS Box Model



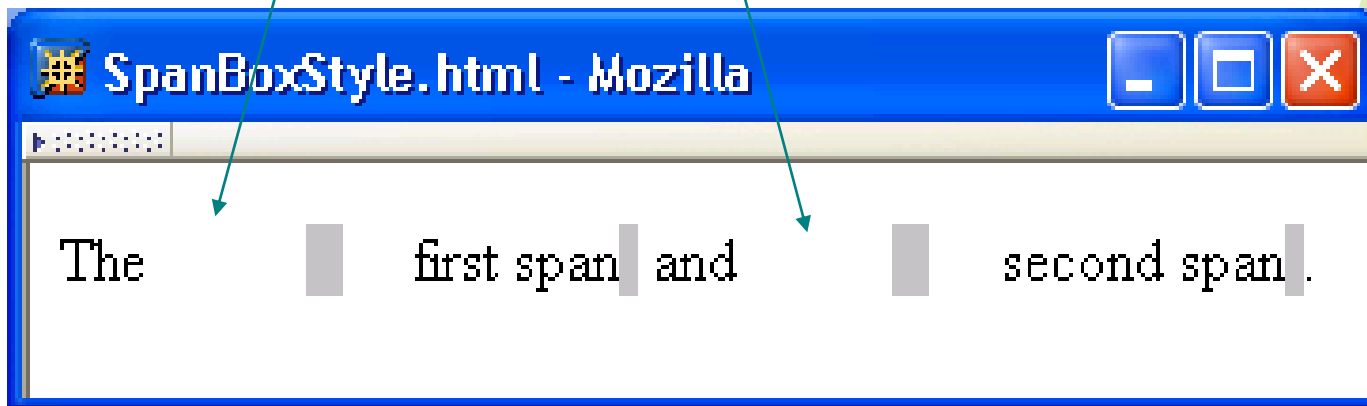
```
span { margin-left: 1cm;  
        border-left-width: 10px;  
        border-left-color: silver;  
        border-left-style: solid;  
        padding-left: 0.5cm;  
        border-right-width: 5px;  
        border-right-color: silver;  
        border-right-style: solid }
```





CSS Box Model

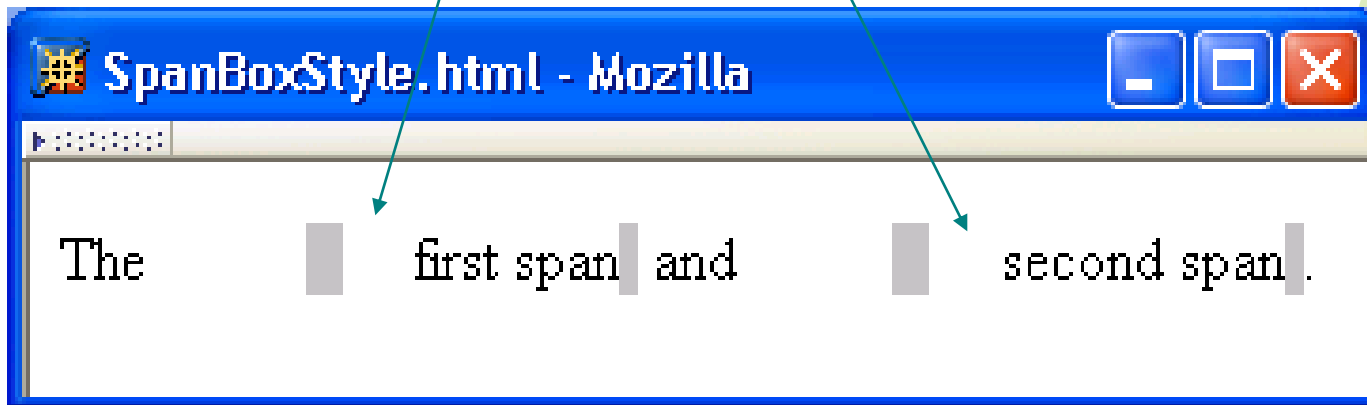
```
span { margin-left: 1cm;  
        border-left-width: 10px;  
        border-left-color: silver;  
        border-left-style: solid;  
        padding-left: 0.5cm;  
        border-right-width: 5px;  
        border-right-color: silver;  
        border-right-style: solid }
```





CSS Box Model

```
span { margin-left: 1cm;  
        border-left-width: 10px;  
        border-left-color: silver;  
        border-left-style: solid;  
        padding-left: 0.5cm;  
        border-right-width: 5px;  
        border-right-color: silver;  
        border-right-style: solid }
```





CSS Box Model

TABLE 3.9: Basic CSS style properties associated with the box model.

Property	Values
<code>padding-{top,right,bottom,left}</code>	CSS length (Sec. 3.6.2).
<code>padding</code>	One to four length values (see text).

TABLE 3.10: Meaning of values for certain shorthand properties that take one to four values.

Number of values	Meaning
One	Assign this value to all four associated properties (top , right , bottom , and left).
Two	Assign first value to associated top and bottom properties, second value to associated right and left properties.
Three	Assign first value to associated top property, second value to right and left , and third value to bottom .
Four	Assign first value to associated top property, second to right , third to bottom , and fourth to left .



CSS Box Model



TABLE 3.9: Basic CSS style properties associated with the box model.

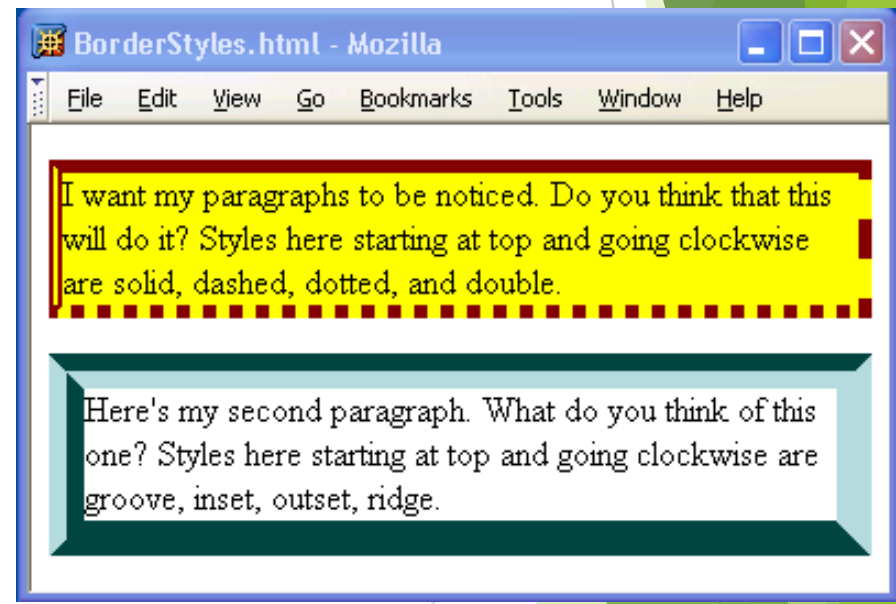
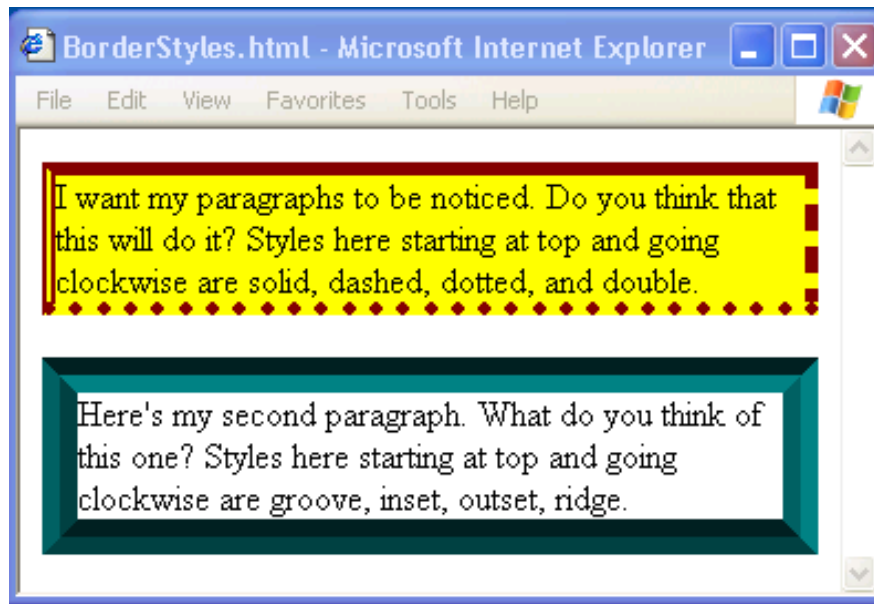
<code>border-{top,right,bottom,left}-width</code>	<code>thin</code> , <code>medium</code> (initial value), <code>thick</code> , or a length.
<code>border-width</code>	One to four <code>border-*-width</code> values.
<code>border-{top,right,bottom,left}-color</code>	Color value. Initial value is value of element's <code>color</code> property.
<code>border-color</code>	<code>transparent</code> or one to four <code>border-*-color</code> values.



CSS Box Model

TABLE 3.9: Basic CSS style properties associated with the box model.

<code>border-{top,right,bottom,left}-style</code>	none (initial value), hidden, dotted, dashed, solid, double, groove, ridge, inset, outset.
<code>border-style</code>	One to four border-*-style values.





CSS Box Model



TABLE 3.9: Basic CSS style properties associated with the box model.

<code>border-{top,right,bottom,left}</code>	One to three values (in any order) for <code>border-*-width</code> , <code>border-*-color</code> , and <code>border-*-style</code> . Initial values are used for any unspecified values.
<code>border</code>	One to three values; equivalent to specifying given values for each of <code>border-top</code> , <code>border-right</code> , <code>border-bottom</code> , and <code>border-left</code> .
<code>margin-{top,right,bottom,left}</code>	<code>auto</code> (see text) or length.
<code>margin</code>	One to four <code>margin-*</code> values.



CSS Box Model



- ▶ If multiple declarations apply to a property, the last declaration overrides earlier specifications

```
{ border: 15px solid;  
border-left: 30px inset red;  
color: blue }
```

Left border is 30px wide, inset style, and red



Backgrounds



- ▶ **background-color**
 - ▶ Specifies background color for content, padding, and border areas
 - ▶ Margin area is always transparent
 - ▶ Not inherited; initial value transparent
- ▶ **background-image**
 - ▶ Specifies (using `url()` function) image that will be **tilled** over an element



Backgrounds

```
<body style="background-image:url('CucumberFlowerPot.png')">
```





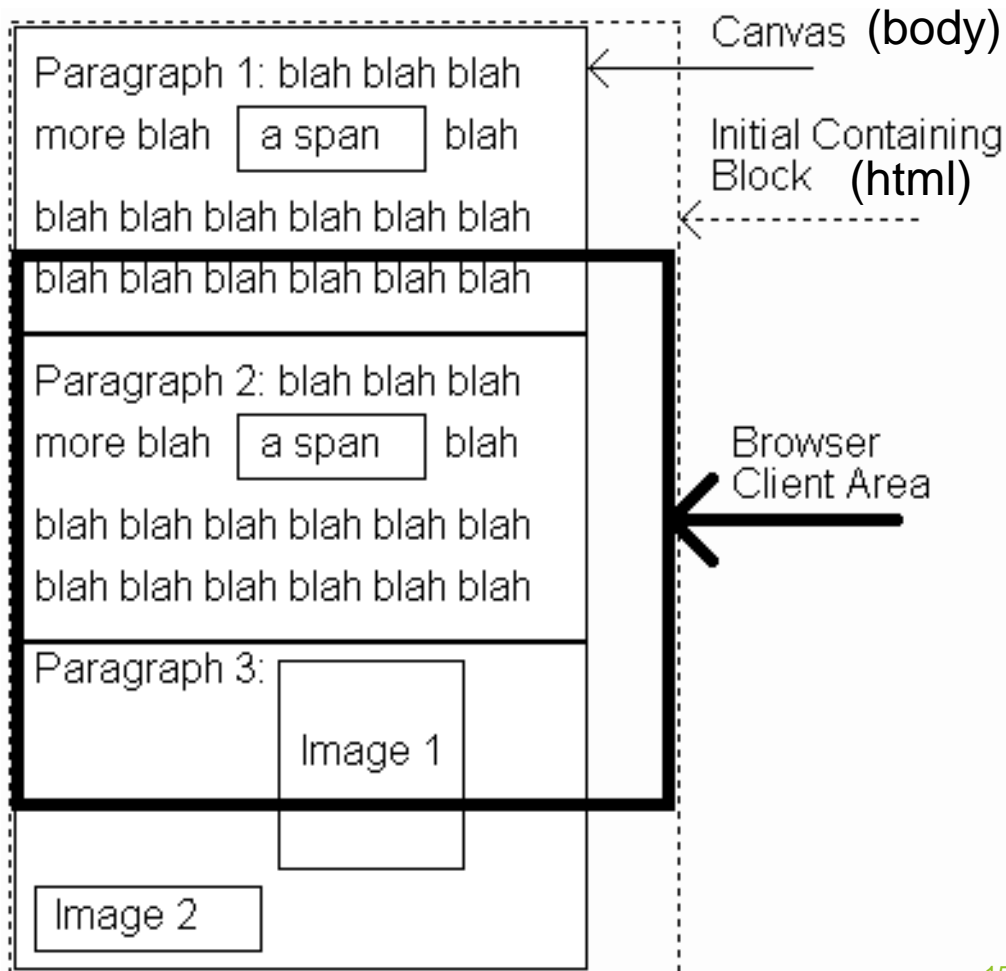
Normal Flow Layout



- ▶ In **normal flow processing**, each displayed element has a corresponding box
 - ▶ html element box is called **initial containing block** and corresponds to entire document
 - ▶ Boxes of child elements are contained in boxes of parent
 - ▶ Sibling **block elements** are laid out one on top of the other
 - ▶ Sibling **inline elements** are one after the other



Normal Flow Layout





Normal Flow Layout



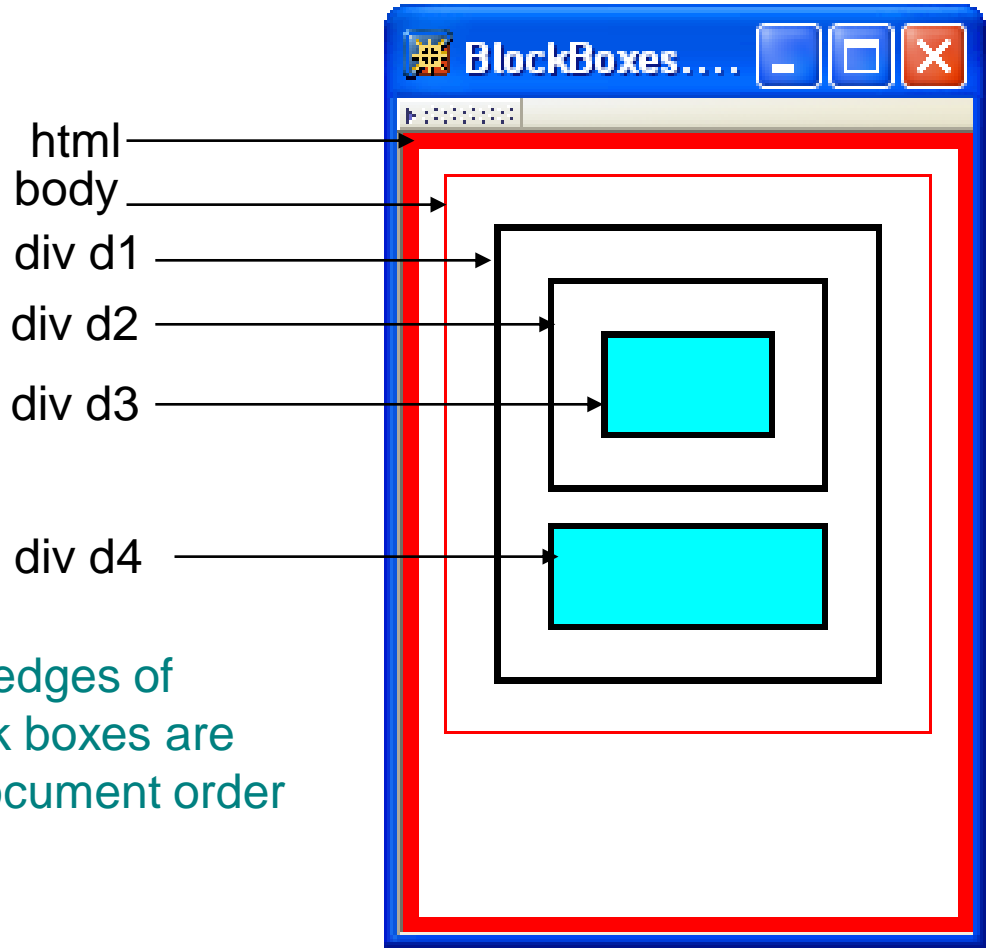
```
html, body { border:solid red thin }
html { border-width:thick }
body { padding:15px }
div { margin:0px; padding:15px; border:solid black 2px }
.shade { background-color:aqua }
.topMargin { margin-top:10px }
```

Block
elements
only

```
<body>
  <div id="d1">
    <div id="d2">
      <div id="d3" class="shade"></div>
    </div>
    <div id="d4" class="shade topMargin"></div>
  </div>
</body>
```




Normal Flow Layout



Top edges of block boxes are in document order



Normal Flow Layout

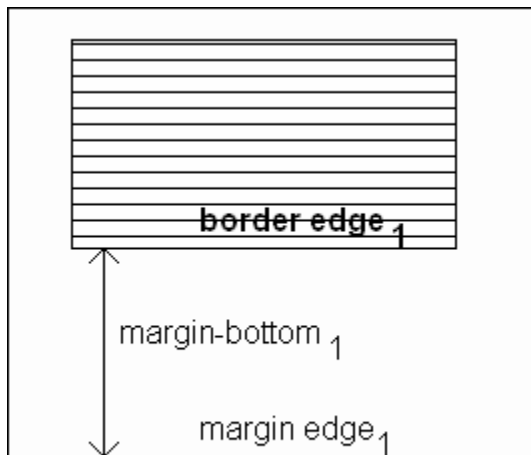


- ▶ What is a “block element”?
 - ▶ Element with value `block` specified for its `display` property
 - ▶ User agent style sheet (not CSS) specifies default values; typical block elements include `html`, `body`, `p`, `pre`, `div`, `form`, `ol`, `ul`, `dl`, `hr`, `h1` through `h6`
 - ▶ Most other elements except `li` and table-related have `inline` specified for `display`

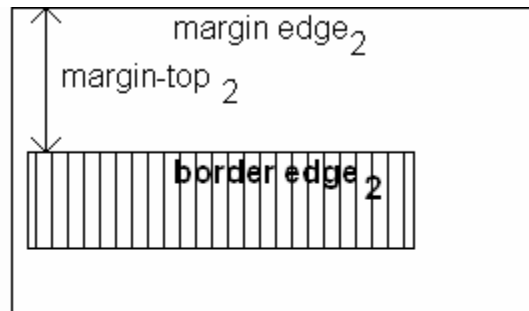


Normal Flow Layout

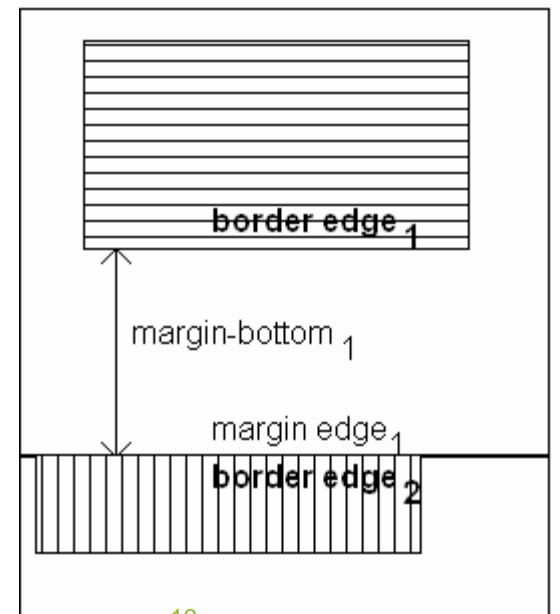
- ▶ When blocks stack, adjacent margins are **collapsed** to the size of the larger margin



(a)



(b)

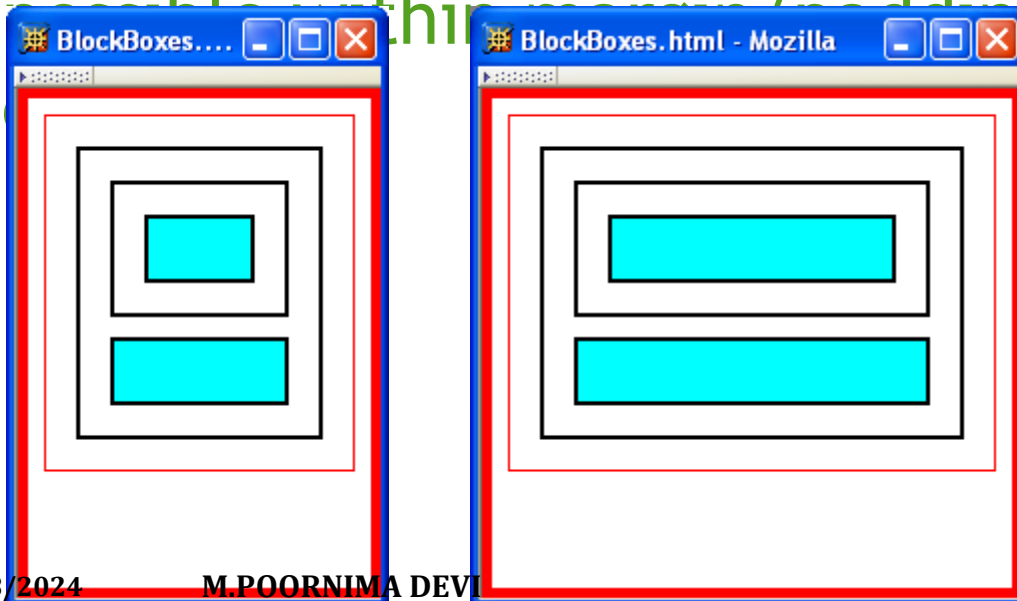


(c)



Normal Flow Layout

- ▶ Initial value of width property is auto, which for block boxes means to make the content area as wide as

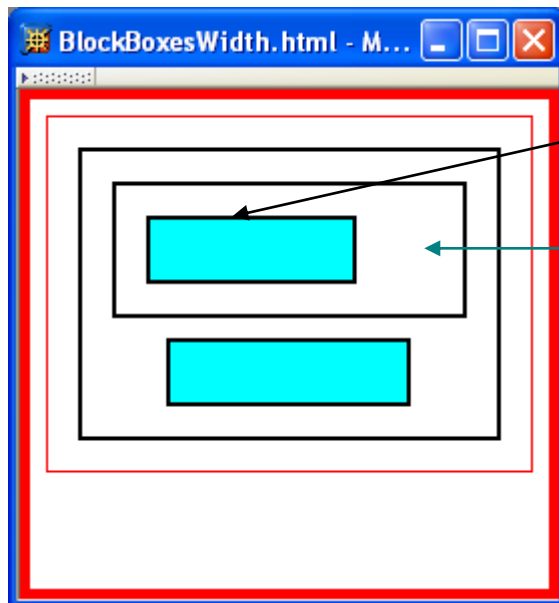


Width of block boxes increases as browser client area is widened



Normal Flow Layout

- ▶ Can also specify CSS length or percentage (of parent's content width) for width property



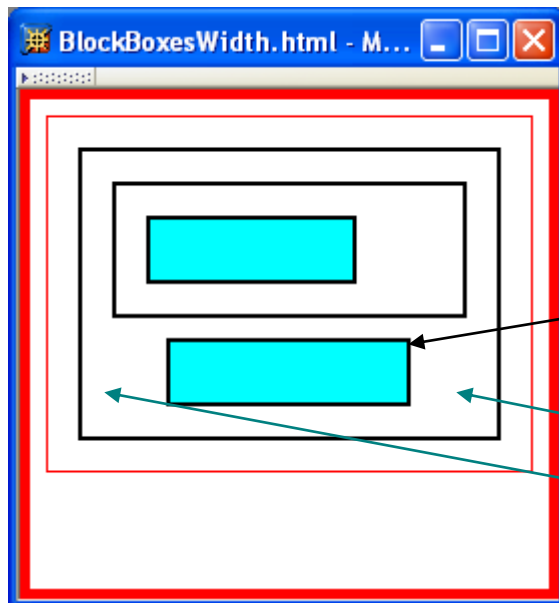
```
#d3 { width:50% }
```

By default, width of right margin is adjusted to accommodate a change to width



Normal Flow Layout

- ▶ Can also specify CSS length or percentage (of parent's content width) for width property



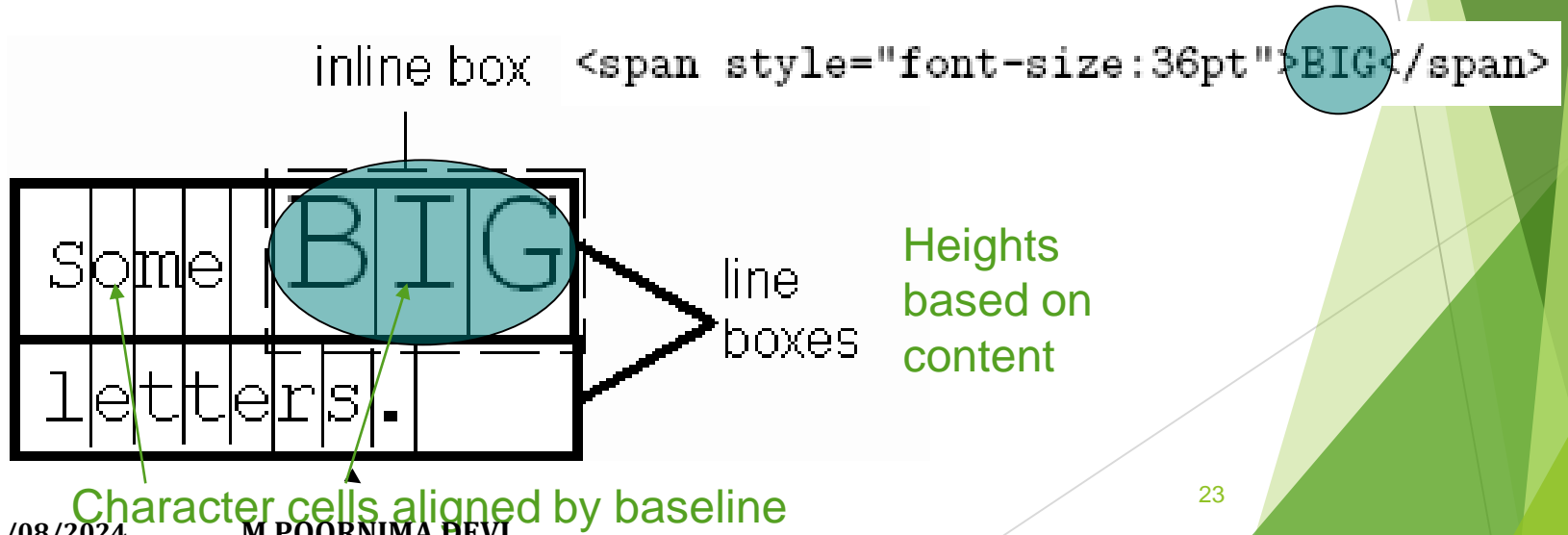
```
#d4 { width:50%; margin-left:auto; margin-right:auto }
```

Centering can be achieved by setting both margins to auto



Normal Flow Layout

- ▶ Boxes corresponding to **character cells** and **inline elements** are laid out side by side in **line boxes** that are stacked one on top of the other

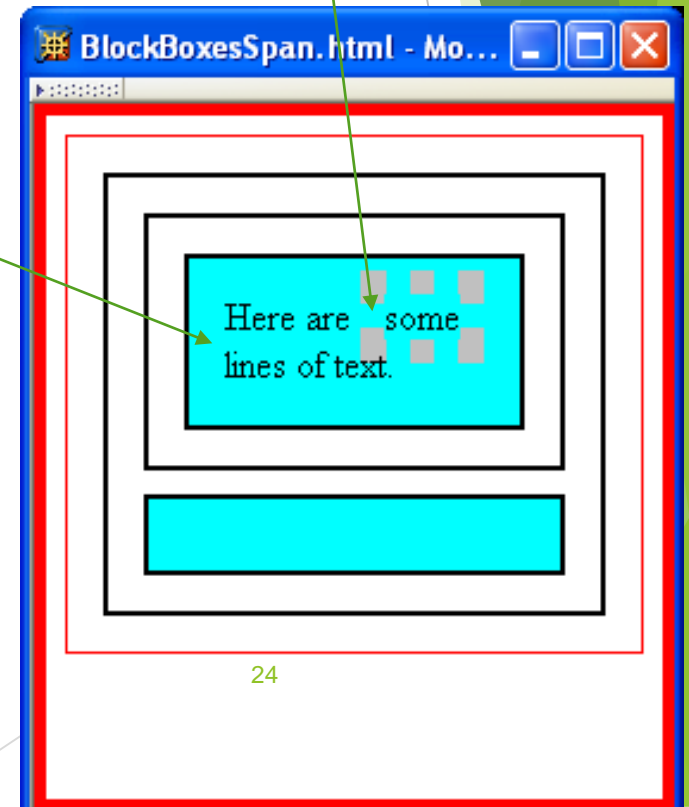




Normal Flow Layout

Padding/borders/margins affect **width** but not **height** of inline boxes

```
<div id="d3" class="shade">  
  Here are  
  <span style="border:dotted silver 10px">some</span>  
  lines of text.  
</div>
```





Normal Flow Layout

- Specify value for `vertical-align` to position an inline element within line box:

initial
value of
`vertical-align`

The screenshot shows a Mozilla browser window with the title 'VerticalAlign.html'. The page content displays six examples of vertical alignment for an inline element (represented by a small image of two kiwis) relative to a line box (indicated by a dashed line). The examples are: `text-top`, `middle`, `text-bottom`, `baseline`, `-.4em`, and `20%`. A green arrow points from the text 'initial value of vertical-align' to the `baseline` example, indicating that `baseline` is the default value.



Beyond Normal Flow

- ▶ CSS allows for boxes to be positioned outside the normal flow:
 - ▶ Relative positioning



span's shifted backwards relative to normal flow

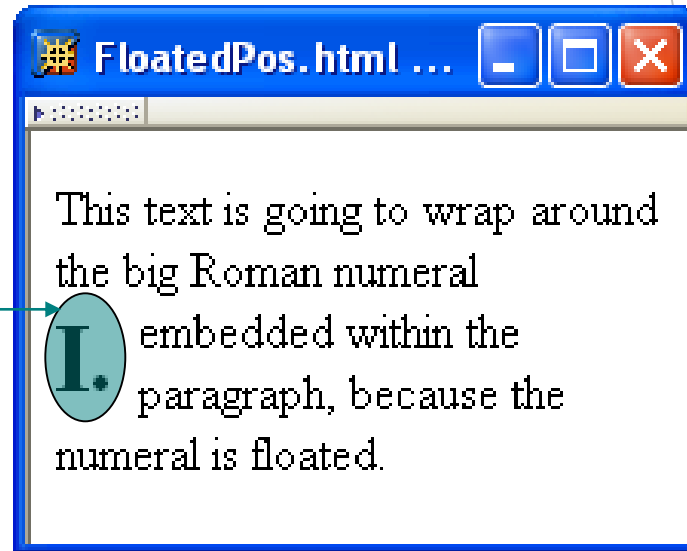


Beyond Normal Flow



- ▶ CSS allows for boxes to be positioned outside the normal flow:
 - ▶ **Float** positioning

span taken out of normal flow and “floated” to the left of its line box



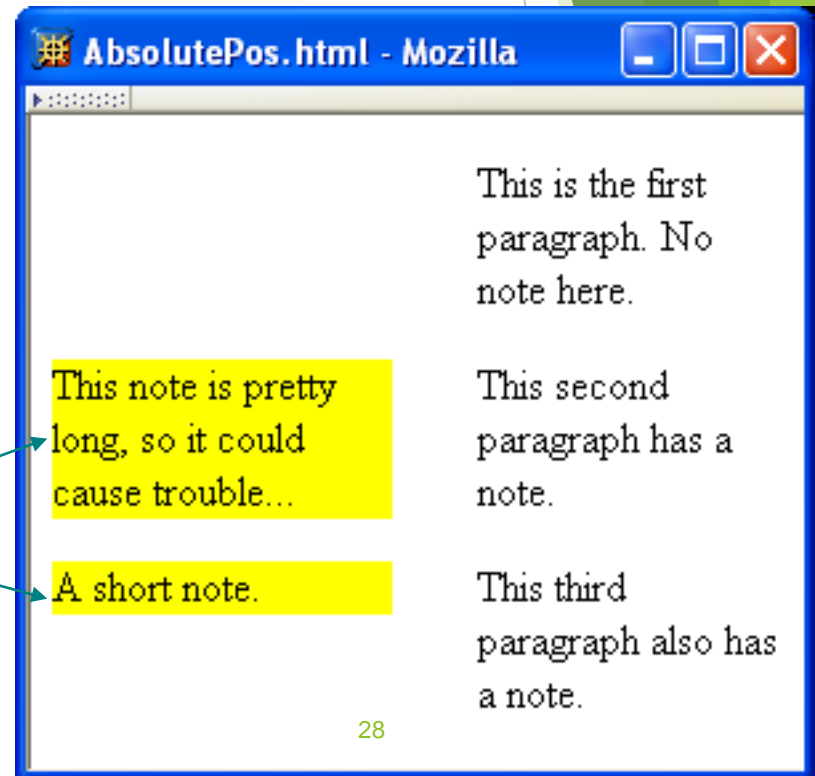


Beyond Normal Flow



- ▶ CSS allows for boxes to be positioned outside the normal flow:
 - ▶ Absolute positioning

span's removed from normal flow and positioned relative to another box





Beyond Normal Flow

- ▶ Properties used to specify positioning:
 - ▶ **position**: static (initial value), relative, or absolute
 - ▶ Element is **positioned** if this property not static
 - ▶ Properties left, right, top, bottom apply to positioned elements
 - ▶ Primary values are auto (initial value) or CSS length
 - ▶ **float**: none, left, or right
 - ▶ Applies to elements with static and relative positioning only



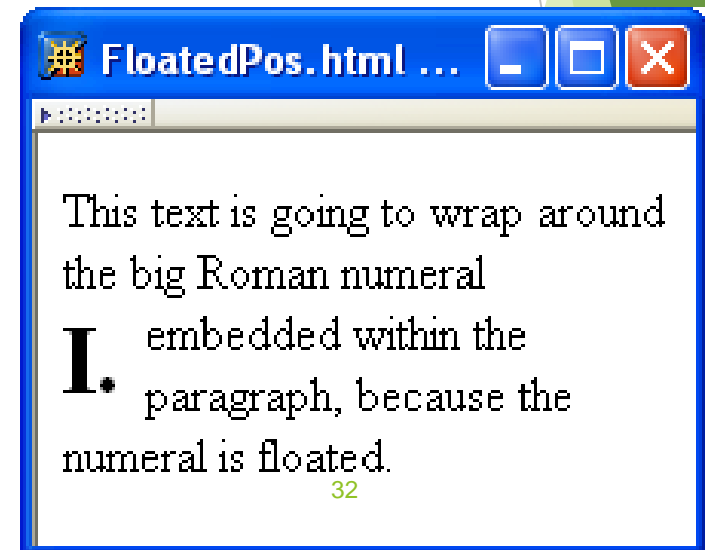
Beyond Normal Flow



- ▶ Float positioning
 - ▶ Specify value for float property

```
.bigNum { float:left; font-size:xx-large; font-weight:bold }
```

This text is going to wrap
around the
I.
big Roman numeral





Beyond Normal Flow



- ▶ Float positioning
 - ▶ Specify value for float property

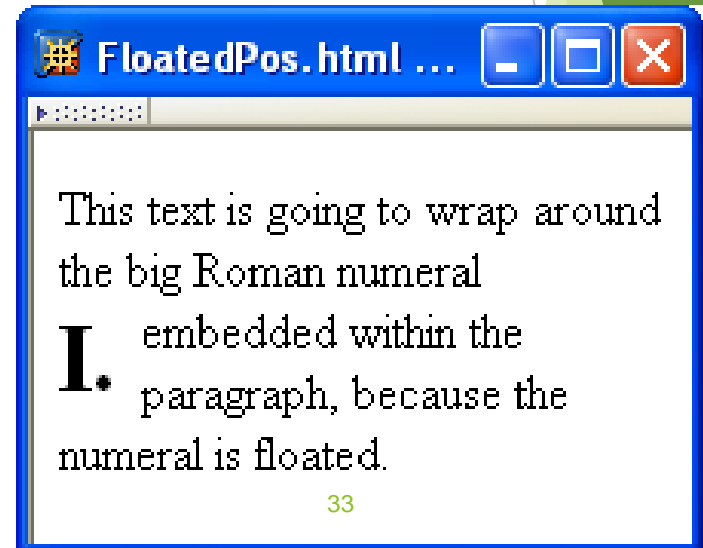
```
.bigNum { float:left; font-size:xx-large; font-weight:bold }
```

This text is going to wrap
around the

```
<span class="bigNum">I.&nbsp;  </span>
```

big Roman numeral

Floated element becomes a CSS block
element (e.g., can set height and width)

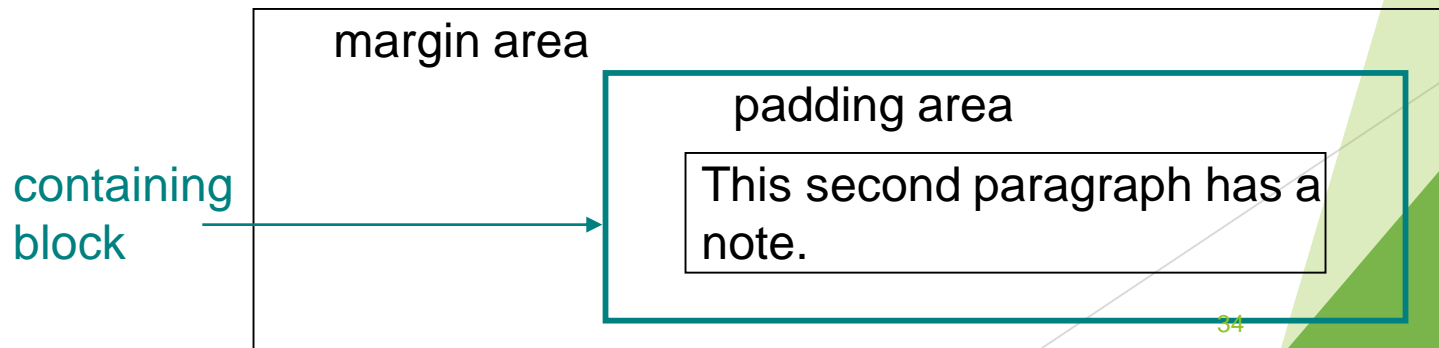




Beyond Normal Flow

- ▶ Absolute positioning
 - ▶ Specify location for corner of box relative to **positioned containing block**

p elements are positioned (but don't move!)
`p { position:relative; margin-left:10em }`





Beyond Normal Flow



- ▶ Absolute positioning
 - ▶ Specify location for edges of box relative to positioned containing block

```
.marginNote { position:absolute;  
               top:0; left:-10em; width:8em;  
               background-color:yellow }
```



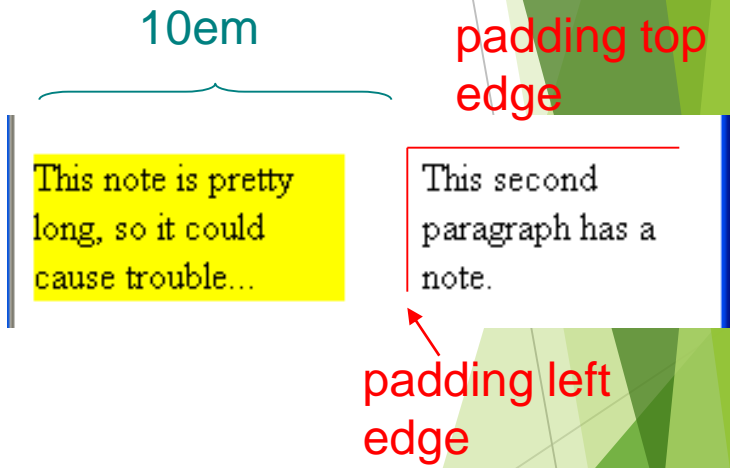
Beyond Normal Flow



► Absolute positioning

```
p { position:relative; margin-left:10em }  
.marginNote { position:absolute;  
  top:0; left:-10em; width:8em;  
  background-color:yellow }
```

```
<p>  
  This second paragraph has a note.  
  <span class="marginNote">This note is pretty long, so  
  it could cause trouble...</span>  
</p>
```





Beyond Normal Flow



► Absolute positioning

```
p { position:relative; margin-left:10em }  
.marginNote { position:absolute;  
  top:0; left:-10em; width:8em;  
  background-color:yellow }
```

This note is pretty long, so it could cause trouble...

This second paragraph has a note.

8em

```
<p>  
  This second paragraph has a note.  
  <span class="marginNote">This note is pretty long, so  
  it could cause trouble...</span>  
</p>
```

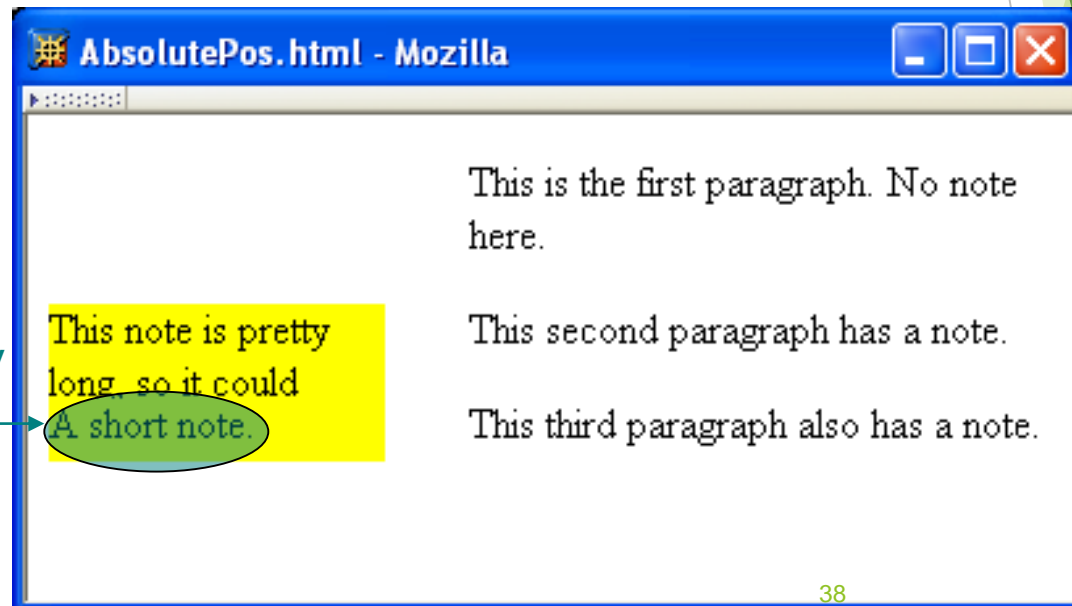


Beyond Normal Flow



- ▶ Absolutely positioned box does *not* affect positioning of other boxes!

Second absolutely positioned box obscures first





CSS Position-Related Properties

- ▶ z-index: drawing order for overlaid boxes (largest number drawn last)

```
#text { position:absolute; top:10px; left:10px;
font-family:"Courier",monospace; letter-spacing:0.1ex;
background-color:yellow;
z-index:1 }
#overlay { position:absolute; top:10px; left:10px;
width:1.1ex; height:4.5em;
border:solid red 1px;
z-index:2 }
```





CSS Position-Related Properties



- ▶ **display**: value none means that element and its descendants are not rendered and *do not* affect normal flow
- ▶ **visibility**: value hidden (initial value is visible) means that element and its descendants are not rendered but still *do* affect normal flow