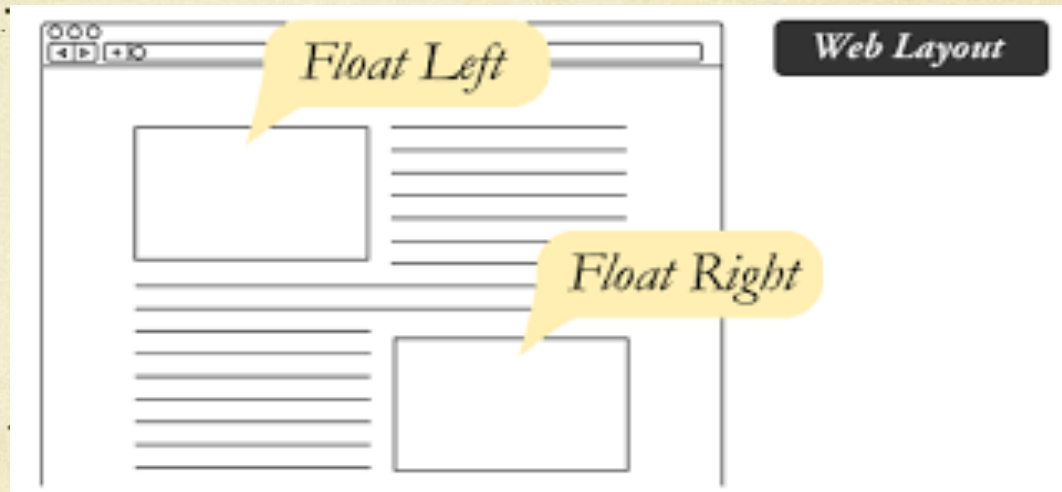


Web Development Fundamentals



ReCap – CSS/Box Model and Creating Page Layout in HTML

Floating elements

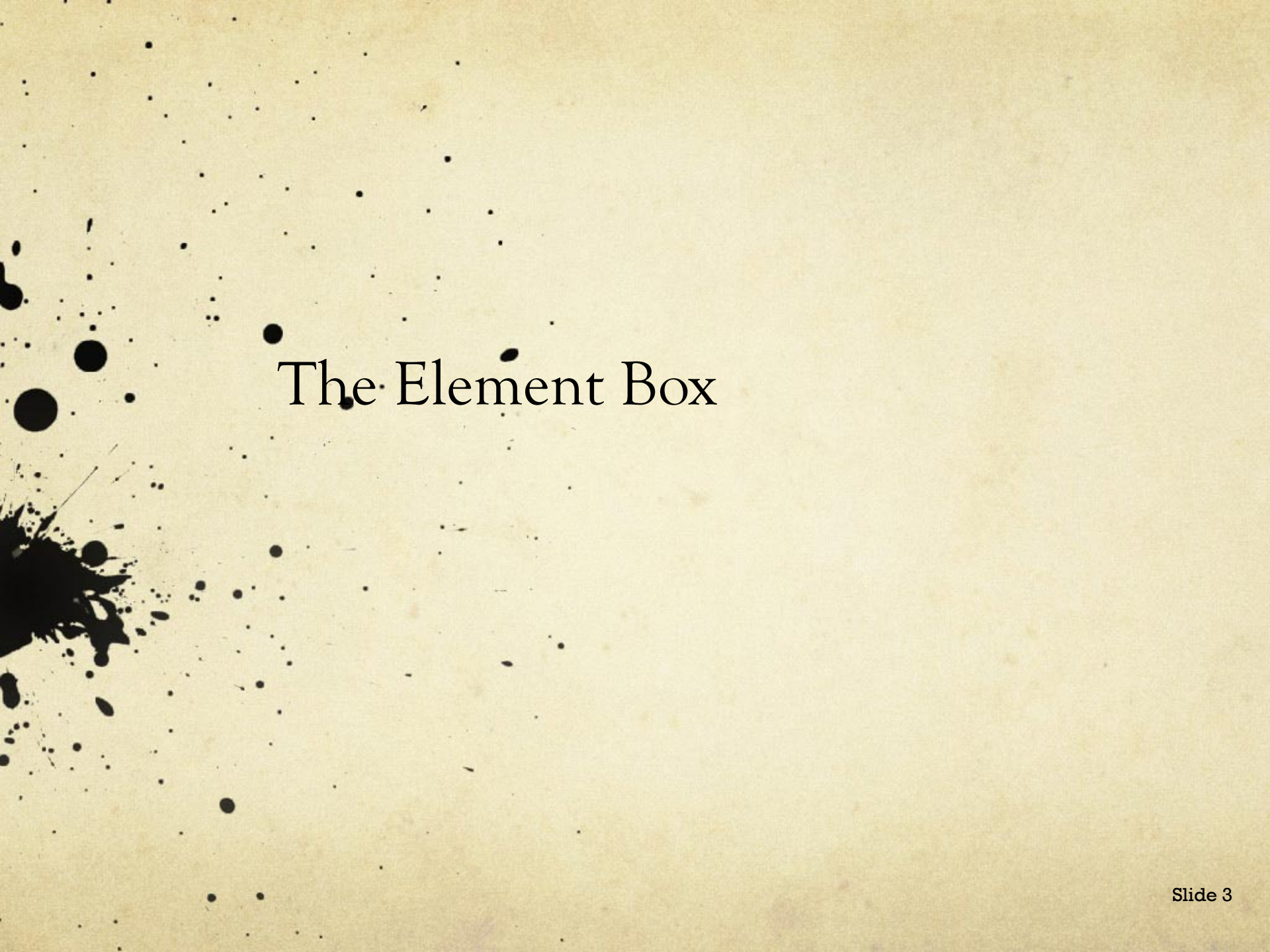
Multi-column layouts

Lecturer: Marie Brennan

Autumn 2017

Introduction

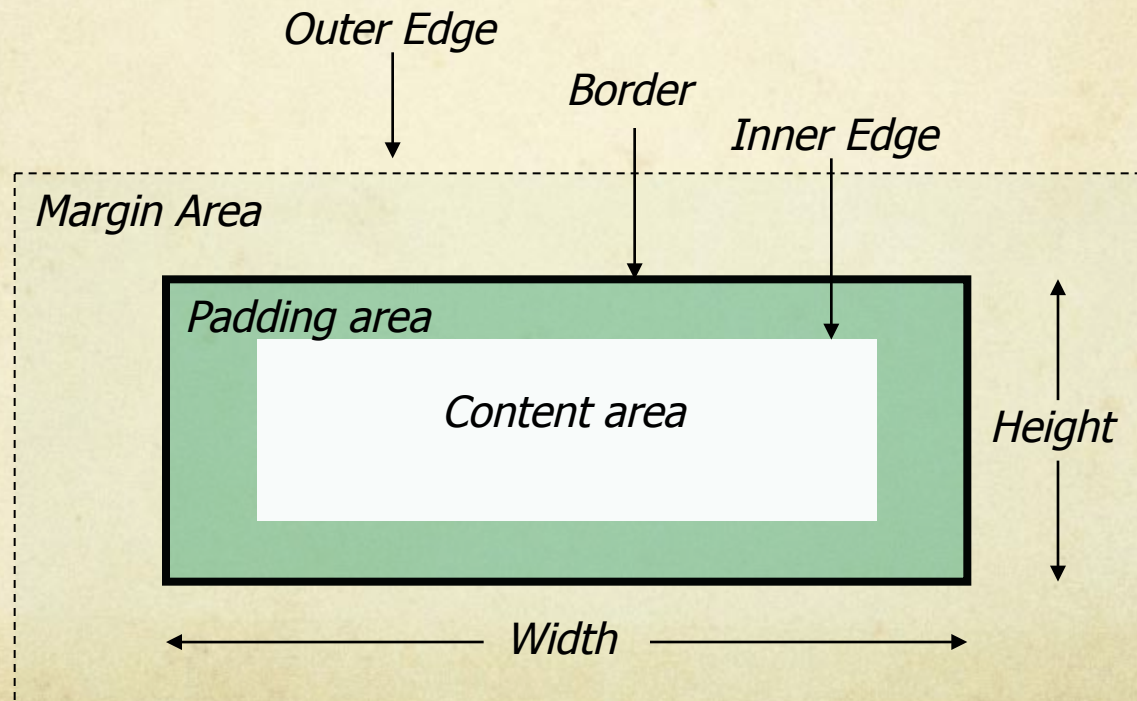
- Every element in a HTML document generates a BOX in which the content is housed.
- In this lecture we are going to look at :
 - Borders
 - Padding

The background of the slide is a light beige or cream color, resembling aged paper. It is decorated with numerous black ink splatters and dots of varying sizes. A large, dense cluster of black ink is located on the left side, with several smaller, isolated dots scattered across the entire surface.

The Element Box


The Element Box

- As we have seen, every HTML element, both block-level or inline generates a rectangular *element box*.
- The components of this box are shown below.



Example -CSS/html and browser

```
p{  
  
background-color: lightgrey;  
  
width: 300px;  
  
border: 25px solid green;  
  
padding: 25px;  
  
margin: 25px;  
  
}
```

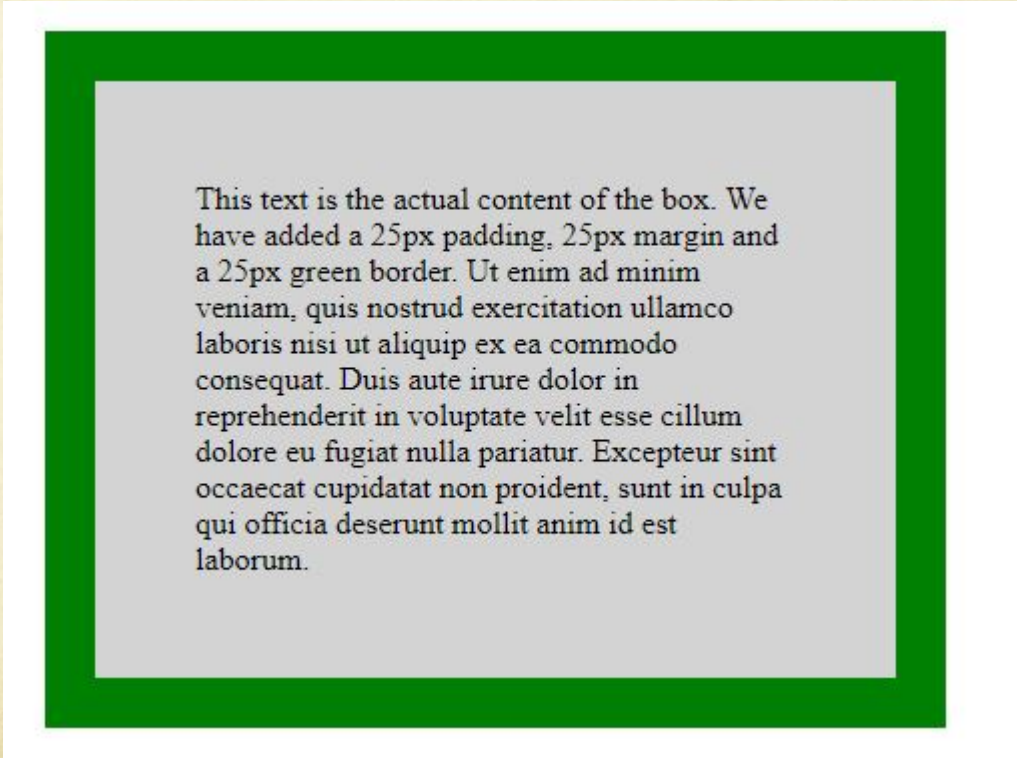


This text is the actual content of the box. We have added a 25px padding, 25px margin and a 25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

<p>This text is the actual content of the box. We have added a 25px padding, 25px margin and a 25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.</p>

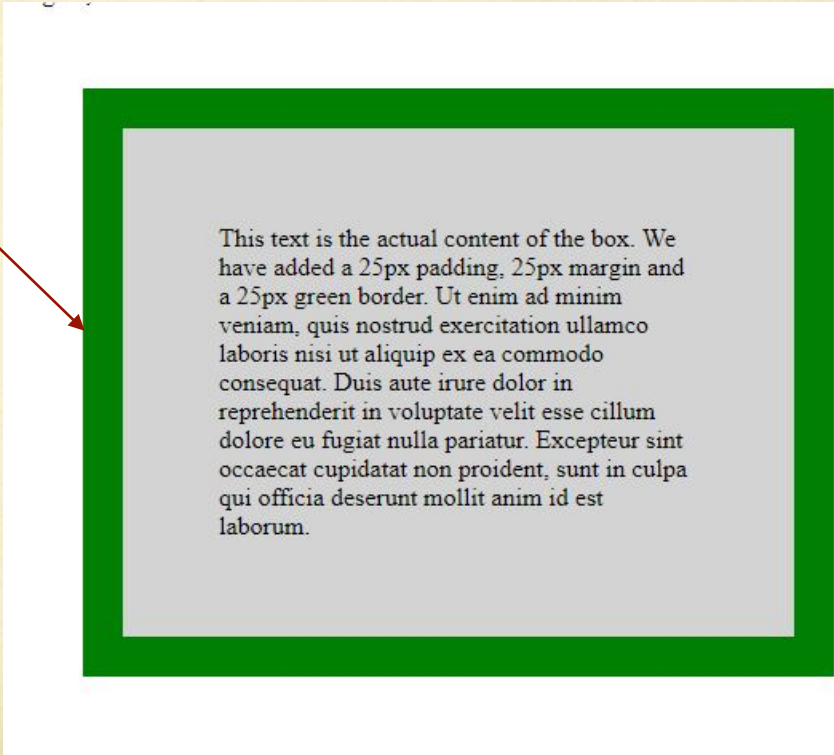
Lets change some of the css and see what happens

```
div {  
  
    background-color: lightgrey;  
  
    width: 300px;  
  
    border: 25px solid green;  
  
    padding: 50px;  
  
    margin: 25px;  
  
}
```



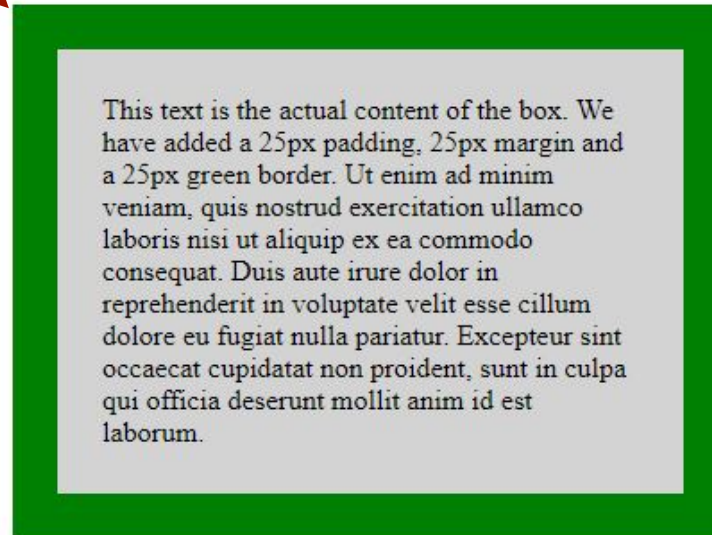
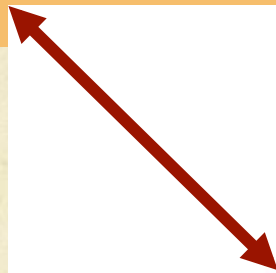
This text is the actual content of the box. We have added a 25px padding, 25px margin and a 25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.


```
p{
  background-color:
lightgrey;
  width: 300px;
  border: 25px solid
green;
  padding: 60px;
  margin: 50px;
}
```



This text is the actual content of the box. We have added a 25px padding, 25px margin and a 25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

```
div {  
  background-color: lightgrey;  
  width: 300px;  
  border: 25px solid green;  
  padding: 25px;  
  margin: 150px;  
}
```




```
p{  
  background-color: lightgrey;  
  width: 600px;  
  border: 25px solid green;  
  padding: 25px;  
  margin: 150px;  
}
```

This text is the actual content of the box. We have added a 25px padding, 25px margin and a 25px green border. Ut enim ad minim veniam, quis nostrud exercitation ullamco laboris nisi ut aliquip ex ea commodo consequat. Duis aute irure dolor in reprehenderit in voluptate velit esse cillum dolore eu fugiat nulla pariatur. Excepteur sint occaecat cupidatat non proident, sunt in culpa qui officia deserunt mollit anim id est laborum.

Padding

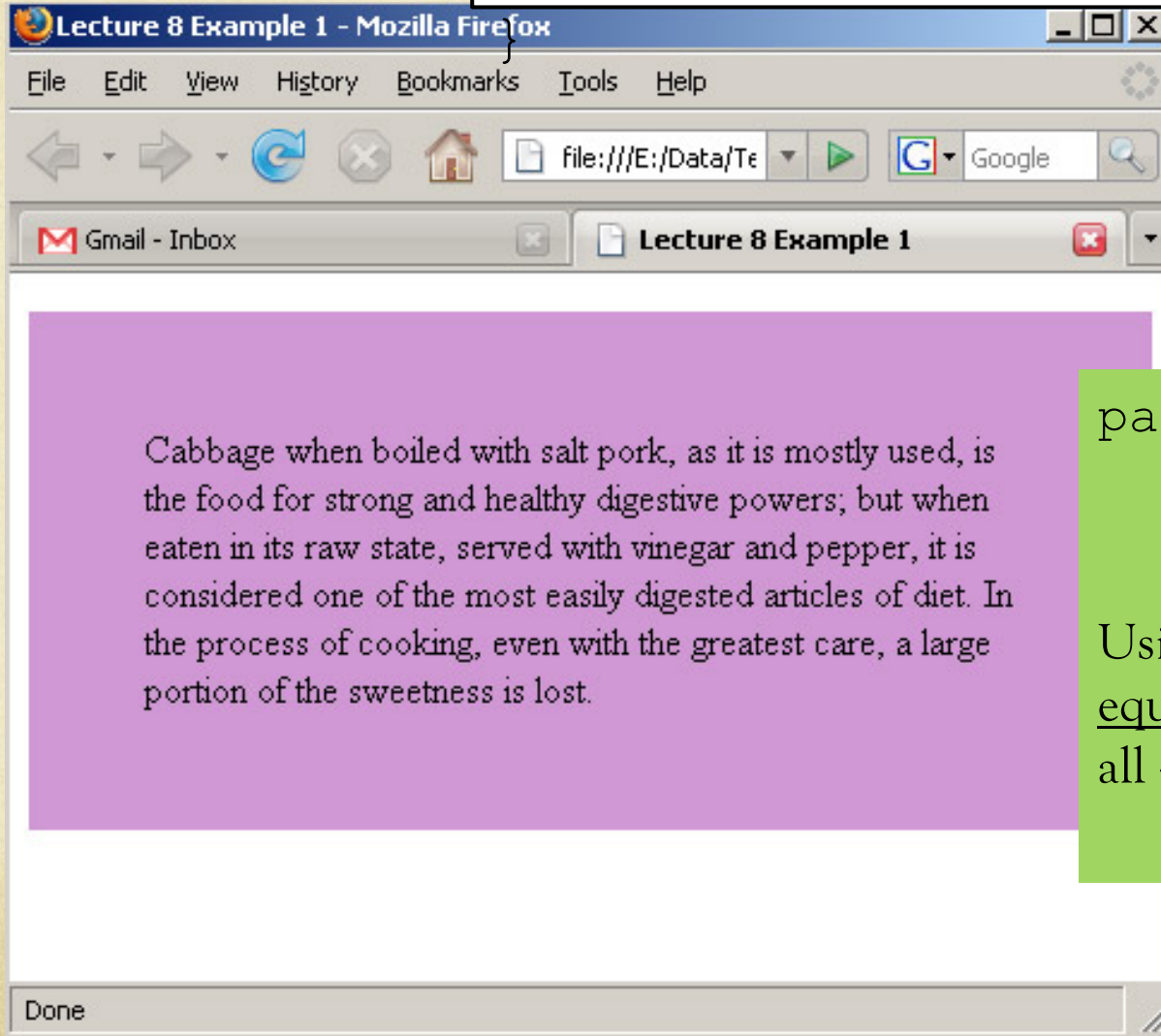
Padding

- *Padding* is space between content area and border
- Rule of thumb – seriously consider some padding if element has background colour or a visible border

example

```
p {  
    padding: 4em;  
    background-color: #D098D4;  
}
```

```
p {  
    padding: 4em;  
    background-color: #D098D4;  
}
```



```
padding: 4em;
```

Using **padding** this way adds equal amount of padding on all 4 sides

Slide

Cabbages and Cauliflowers:

How to grow them

A Practical Treatise, Giving Full Details on Every Point, Including
Keeping and Marketing the Crop
by JAMES J. H. GREGORY

Individual padding for
all 4 sides can be defined:

```
p {  
    padding-top: 4em;  
    padding-bottom: 1em;  
    padding-left: 1em;  
    padding-right: 1em;  
    background-color: #D098D4;  
}
```

padding-top
padding-right
padding-bottom
padding-left



Borders

Borders: style

- A border is a line drawn around content area and its (optional padding).
- 8 different border styles and border can be any width or colour we want

border-style

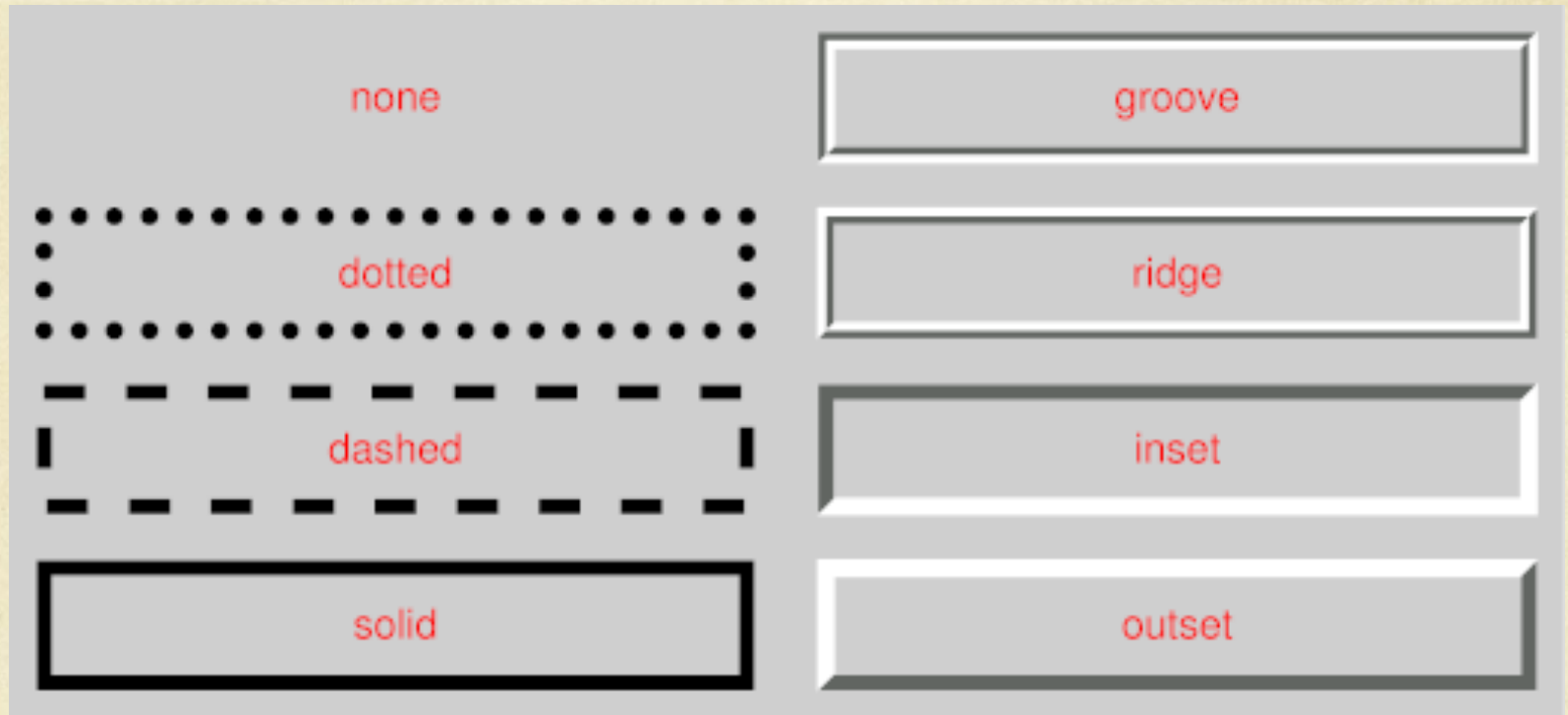
values: none | dotted | dashed | solid
| double | groove | ridge | inset |
outset | inherit

default: none

applies to: all elements

inherits: no

Borders: style



- Image above shows result of applying the different options.

Cabbage when boiled with salt pork, as it is mostly used, is the food for strong and healthy digestive powers; but when eaten in its raw state, served with vinegar and pepper, it is considered one of the most easily digested articles of diet. In the process of cooking, even with the greatest care, a large portion of the sweetness is lost.

```
/* another example of adding a border */  
p {  
    border-style: dashed;  
    background-color: #D098D4;  
    padding: 2em;  
}
```

Borders: style

- You can also specify different border for the top, left, bottom and right of elements.
- This is not too common but if you want to, you can do it with these properties:
 - border-top-style
 - border-right-style
 - border-bottom-style
 - border-left-style

border-top-style

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
p {  
border-top-style: dashed;  
background-color: #FFF9DC;  
padding: 2em;  
}
```

border-left-style

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and made a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
p {  
  border-left-style: dashed;  
  background-color: #FFF9DC;  
  padding: 2em;  
}
```


Borders: width

- Border width can be controlled with the **border-width** property.
- You can give this a length unit or use predefined keywords – `thin`, `medium`, or `thick`. Here are some examples:
- And here is an example of creating a class for formatting...

```
.help {  
    border-width: 2em;  
    border-style: solid;  
    width: 300px;  
    height: 100px;  
}  
  
.banner {  
    border-width: thin;  
    border-style: dashed; }
```

```
.banner {  
border-width: thin;  
border-style: dashed; }
```

CSS

```
<p class = "banner">
```

HTML

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
</p>
```

browser

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

CSS

```
.help {  
border-width: 0.5em;  
border-style: solid;  
width: 300px;  
height: 100px; }
```

HTML

```
<p class = "help">
```

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged. It was popularised in the 1960s with the release of Letraset sheets containing Lorem Ipsum passages, and more recently with desktop publishing software like Aldus PageMaker including versions of Lorem Ipsum.

```
</p>
```

browser



orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

Borders: color

- You can set also specify border colour
 - This then overrides the inheritance of the foreground (text) color of the element to have a border)
- You do this with, you guessed it, the **border-color** property.
- You can give this an RGB value or one of the predefined colour names we talked about previously
- Also **border-top-color** etc. properties for specifying different colours for top, right, left and bottom borders.


```
.banner {  
border-width: thin;  
border-style: dashed;  
border-top-color: red;  
}
```

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.

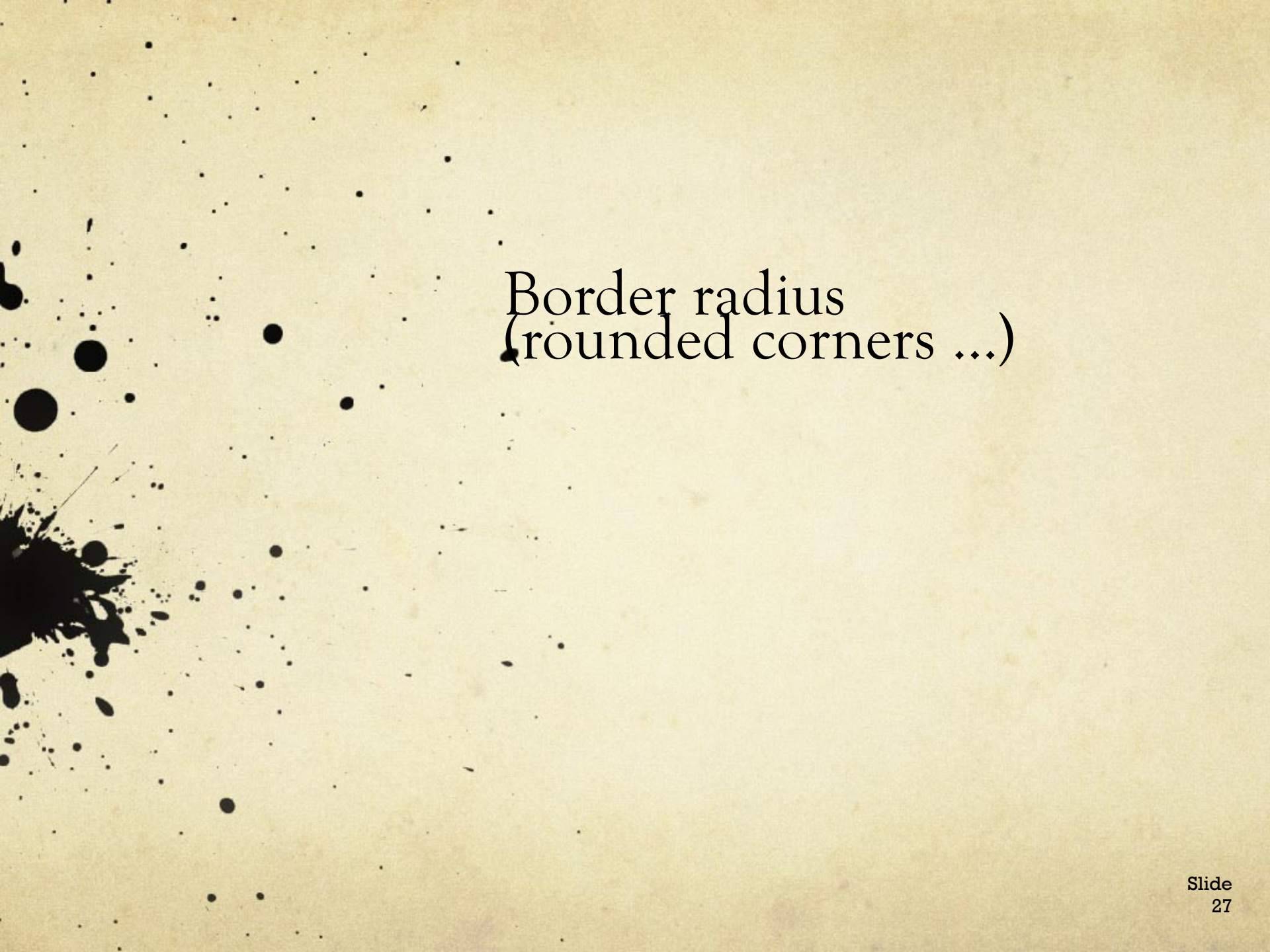
Borders: shorthand rule

- Can set width/style/color in a single declaration:
 - Order does not matter, although most web developers use the order: width style color
 - E.g.

```
footer { border: 2px solid red; }
```

- Can also use shorthand rule for individual parts (top, bottom, left, right) of the border, e.g.:

```
a:hover {  
    border-top: 1px dashed red;  
    border-bottom: 1px dashed red;  
}
```

Border radius
(rounded corners ...)



`border-radius: 1em;`



`border-radius: 50px;`

- A curved radius can be defined
 - The same for all 4 corners
 - Differently for each corner individually
- Length values of EMs, pixels etc. can be used

```
border-radius: 1em;      /* subtle rounding */  
border-radius: 50px;     /* very rounded */
```



```
.help {  
border-width: 0.5em;  
border-style: solid;  
width: 300px;  
border-radius: 1em;  
height: 200px;  
}
```

orem Ipsum is simply dummy text of the printing and typesetting industry. Lorem Ipsum has been the industry's standard dummy text ever since the 1500s, when an unknown printer took a galley of type and scrambled it to make a type specimen book. It has survived not only five centuries, but also the leap into electronic typesetting, remaining essentially unchanged.



`border-top-right-radius: 50 px;`

- Changing just one or two corners can be effective
 - E.g. for 'tab' effects etc.

`border-top-right-radius: 50px;`

B



- Can give 2 values
 - First for horizontal radius
 - Second for vertical radius
- `border-top-right-radius: 100px 50px;`
`border-top-left-radius: 100px 50px;`
 - Here we see less vertical curve, but larger horizontal curve

Width and height

- Width and Height of an Element
- In order to set the width and height of an element correctly in all browsers, you need to know how the box model works.
- **Important:** When you set the width and height properties of an element with CSS, you just set the width and height of the **content area**.
- To calculate the full size of an element, you must also add padding, borders and margins.

Assume we want to style a <p> element to have a total width of 350px:

Example

```
<h2>Calculate the total width:</h2>
```

```
  
<p>The picture above is 350px wide. The total width of this  
element is also 350px.</p>
```

Calculate the total width:



The picture above is 350px wide. The total width of this element is also 350px.

```
p {  
  width: 320px;  
  padding: 10px;  
  border: 5px solid  
gray;  
  margin: 0;  
}
```



Some maths 😊

- 320px (width)
 - + 20px (left + right padding)
 - + 10px (left + right border)
 - + 0px (left + right margin)
 - = 350px

```
p {  
    width: 320px;  
    padding: 10px;  
    border: 5px solid gray;  
    margin: 0;  
}
```

The total width of an element should be calculated like this:

Total element width = width + left padding + right padding + left border + right border + left margin + right margin

The total height of an element should be calculated like this:

Total element height = height + top padding + bottom padding + top border + bottom border + top margin + bottom margin

Note for old IE: Internet Explorer 8 and earlier versions, include padding and border in the width property. This problem is fixed by adding the <!DOCTYPE html> to the HTML page.

Add the code to set the width to 200px.

```
div {
```

```
background-color: lightblue;
```

```
width: 200px;
```

```
}
```

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Lorem ipsum dolor sit amet, consectetur adipiscing elit, sed do eiusmod tempor incididunt ut labore et dolore magna aliqua.

Set the padding of the div to "25px"

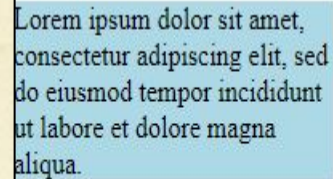
```
div {
```

```
    background-color: lightblue;
```

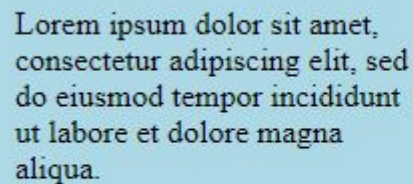
```
    width: 200px;
```

```
    padding:25px;
```

```
}
```



Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed
do eiusmod tempor incididunt
ut labore et dolore magna
aliqua.



Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed
do eiusmod tempor incididunt
ut labore et dolore magna
aliqua.

Set the border of the div to "25px solid navy"

```
div {  
  
  background-color: lightblue;  
  
  width: 200px;  
  
  padding: 25px;  
  
  border: 25px solid navy;  
  
}
```

Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed
do eiusmod tempor incididunt
ut labore et dolore magna
aliqua.

Lorem ipsum dolor sit amet,
consectetur adipiscing elit, sed
do eiusmod tempor incididunt
ut labore et dolore magna
aliqua.

Summary and Conclusions

The Box Model

We set a `width` property value of 400 pixels, but the actual width of our element is 492pixels.

- By default the box model is additive
- To determine the actual size of a box we need to take into account `padding`, `borders`, and `margins` for all four sides of the box.
- Our width not only includes the `width` property value, but also the size of the left and right padding, left and right borders, and left and right margins.
- So far a lot of these properties might not make a whole lot of sense, and that's all right.

To clarify things, let's take a close look at all of the properties—`width`, `height`, `padding`, `border`, and `margin`—that go into forming the box model.

Border, Padding, Margins

- Border-top, border-bottom...left and right

```
div {  
border-top-right-radius: 5px; }
```

...And same applies for **border-top-left-radius**.....etc

Border, Padding, Margins

Border Radius Demo

HTML

CSS

Result

```
border-top-right-radius: 50%;
```

5px

50%

15px 75px

Border, Padding, Margins

- Margin-top, margin-bottom....left and right
- Padding-top, bottom, left and right

New

Padding-box

- The **padding-box** value alters the box model by including any **padding** property values within the width and height of an element.
- When using the padding-box value, if an element has a width of 400 pixels and a padding of 20 pixels around every side, the actual width will remain 400 pixels.
- As any padding values increase, the content size within an element shrinks proportionately.
- If we add a border or margin, those values will be added to the width or height properties to calculate the full box size.

Padding-box

For example, if we add a border of 10 pixels and a padding of 20 pixels around every side of the element with a width of 400 pixels,
.....the actual full width will become 420 pixels.

```
div { box-sizing: padding-box; }
```



So to have this in your site use this code in the css

Box Sizing

```
div { -webkit-box-sizing: content-box;  
      -moz-box-sizing: content-box;  
      box-sizing: content-box; }
```

- If you set the width of an element to 400 pixels and then add 20 pixels of padding and a border of 10 pixels on every side, the actual full width of the element becomes 460 pixels.
- Remember, we need to add the width, padding, and border property values together to get the actual, full width of an element.
- The box model may, however, be changed to support different calculations. CSS3 introduced the box-sizing property, which allows us to change exactly how the box model works and how an element's size is calculated.
- The property accepts three primary values—**content-box**, **padding-box**, and **border-box**—each of which has a slightly different impact on how the box size is calculated.

Box Model

- Box model
 - Spacing and size of elements (div, P, H1, img, UL, OL)
- Padding
 - Spacing between content and border
 - Padding on 4 sides (top/bottom/left/right)
- Borders
 - Line drawn around (content + padding) rectangle
 - Can get fancy with new properties like border-radius
- Design rule of thumb
 - IF you use border OR background color
 - THEN define some padding

Separating Content- recap

<div> Element tag

- Useful for dividing parts of the page into sections.
- Creates a “box” with the following attributes:
 - margin
 - padding
 - border
 - height
 - width
 - (..and lots more)
- Primary element used for CSS Layouts
(more information in CSS Layouts tutorial)

HTML semantic page elements

The magnificent 7 elements

- Main 5 useful new elements:
 - header
 - nav
 - footer
 - aside
 - section
- Also (less useful), remaining 2 new elements:
 - article
 - address

<header>

<nav>

<section>

<article>

<aside>

<footer>

header	Defines a header for a document or a section
nav	Defines a container for navigation links
section	Defines a section in a document
article	Defines an independent self-contained article
aside	Defines content aside from the content (like a sidebar)
footer	Defines a footer for a document or a section
details	Defines additional details
summary	Defines a heading for the details element



**Headers
Footers**

Headers and footers

<header>

- Headers
 - For introductory material about
 - Page
 - Section/article in a page

<footer>

- Footers
 - For information typically at the end of
 - Page
 - Section/article in a page
 - E.g. author details / copyright etc.

NOTE:

NESTING not permitted

- Headers cannot contain other headers or footers
- Footers cannot contain other headers or footers

section

Long documents, or parts of documents,
can benefit by being broken down into sections

<section>

Content in here

</section>

<section>

<h2>Typography Books</h2>

...

</section>

<section>

<h2>Online Tutorials</h2>

<p>These are the best tutorials on the web.</p>

...

</section>



**Aside
(sidebars)**

Aside (sidebars)

- For content that is related but surrounding content on left/right side of the screen
- Might be for:
 - Pull quotes
 - List of links or references
- Content that is related but not essential to the main document
 - E.g. If aside NOT visible, page still makes sense ...

<aside>

- **NOTE – do NOT use for main navigation links of the page – use <nav> for that ...**



we will cover these in detail

<h1>Web Typography</h1>

<p>

Back in 1997, there were competing font formats and tools for making them...</p>

<p>

We now have a number of methods for using beautiful fonts on web pages...</p>

<aside>

<h2>Web Font Resources</h2>

Typekit

Google Fonts

</aside>

Nav (navigation)

Part of a page that provides primary navigational links

- E.g.

- Links to important parts of the SITE
- Or to different parts of a long section of article

Not to be used for tangential links (use aside for that)

<nav> </nav>

```
<nav>
```

```
  <ul>
```

```
    <li><a href="...">Home page</a>/li>
```

```
    <li><a href="...">About</a></li>
```

```
    <li><a href="...">Contact Us</a></li>
```

```
    <li><a href="...">Products</a></li>
```

```
    <li><a href="...">Our customers</a>/li>
```

```
  </ul>
```

```
</nav>
```

DIV (division)

Generic block – when no other is appropriate

<div>

<div>

- Tip: The <div> element is very often used together with CSS, to layout a web page.
- By default, browsers always place a line break before and after the <div> element.
- However, this can be changed with CSS.

<!-- use of DIV to group 2 blocks →

<div>

<section>

main content here

</section>

<aside>

bonus content here

</aside>

</div>

<!-- use of DIV to distinguish from section in footer

- So can target each block in footer separately in CSS-->

<footer>

<section>

some content here

</section>

<div>

other content here

</div>

</div>

The background is a light beige or cream color with a textured, slightly mottled appearance. On the left side, there are several black ink splatters of varying sizes, including a large, dark, irregular blotch near the bottom left corner and numerous smaller, scattered dots and droplets extending towards the center.

Examples of page
structure
templates ...

```
<header>
    (logo sand navigation here)
    <nav>
        (list of links here)
    </nav>
</header>

<section class="main">
    (main page content here)
</section
```

```
<header>
    (logo and navigation here)
</header>
<nav>
    (list of links here)
</nav>

<section>
    (main page content here)
</section>
<aside>
    (bonus content here)
</aside>

<footer>(copyright information here)
</footer>
```



```
<nav>
    (list of links here)
</nav>

<div>
    <section>
        (column1 content here)
    </section>
    <section>
        (column2 content here)
    </section>
</div>

<footer>
    (copyright information here)
</footer>
```

Other page content here ...

```
<!-- multi-column footer -->
```

```
<footer>
```

```
    <section>
```

```
        (left content here)
```

```
    </section>
```

```
    <section>
```

```
        (middle content here)
```

```
    </section>
```

```
    <section>
```

```
        (right content here)
```

```
    </section>
```

```
</footer>
```

Conclusion: Page content

- HTML 5 page content – the most useful 5 block elements:
 - header
 - footer
 - nav
 - section
 - aside
- And DIV when the others not enough ...we will cover more divs and spans later...

Layout of web page

- 2 ways
- First = divs, ids and classes
- Second = HTML 5 sections

Divs, ids in HTML

<div id = “header”>CONTENT of HEADER HERE</div>

<div id =“nav”> NAVIGATION BAR HERE </div>

<div id =“main”> MAIN PAGE CONTENT HERE </div>

<div id =“section”> SOME MORE CONTENT HERE</div>

<div id = “footer”> FOOTER STUFF HERE</div>

Formatted in CSS

#header {}

#nav {}

#section {}

#main {}

#footer {}

Html Sections in html

<header> </header>

<nav> </nav>

<section> </section>

<main> </main>

<aside> </aside>

<article> </article>

<footer> </footer>

HTML sections formatted in CSS as

header {

nav {

section {

aside {...}

Section {.....}

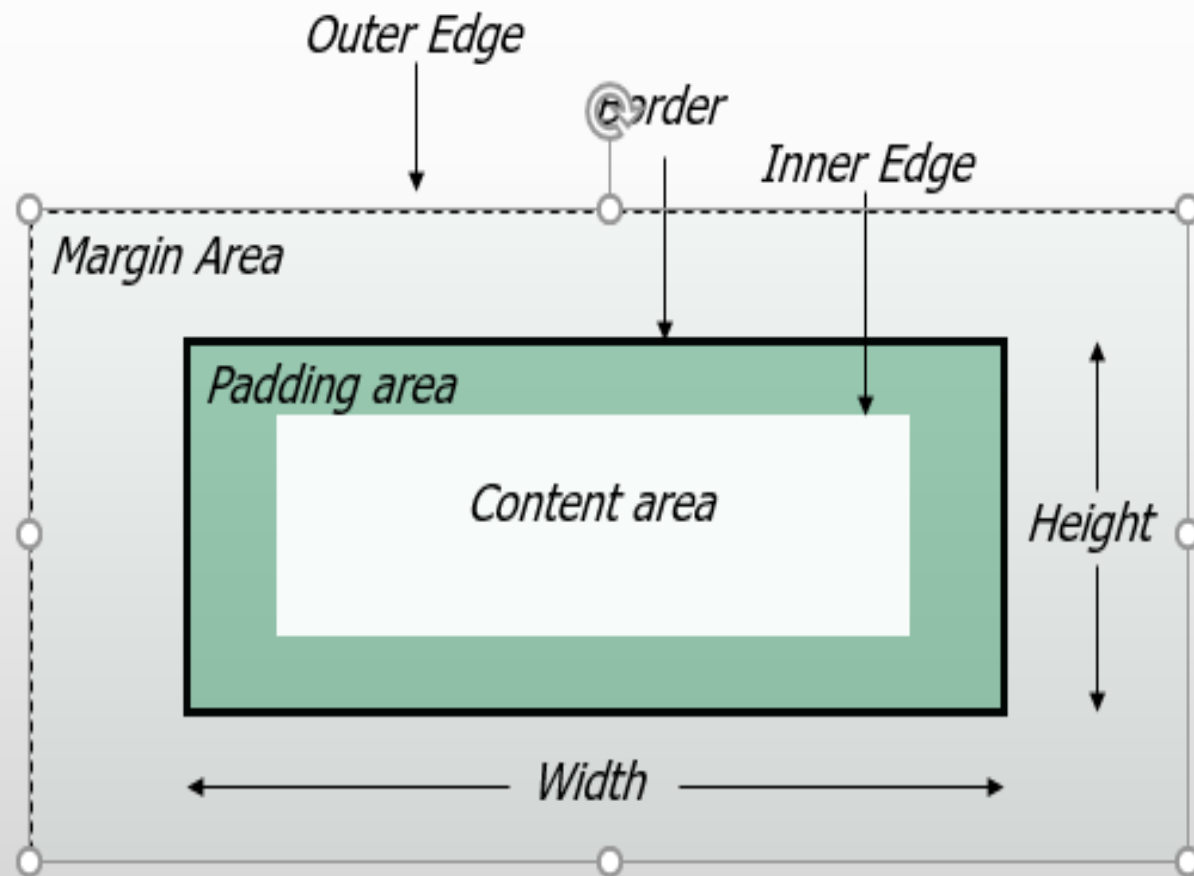
main {

footer {



Margins

The box model



Margins:

keep elements from bumping into one another

- Can use them to make space for another column or content
- Margins work similarly padding ..

margin

values: length | percentage | auto | inherit

default: auto

applies to: all elements

inherits: no

margin-top, margin-right, margin-bottom, margin-left

values: length | percentage | auto | inherit

default: auto

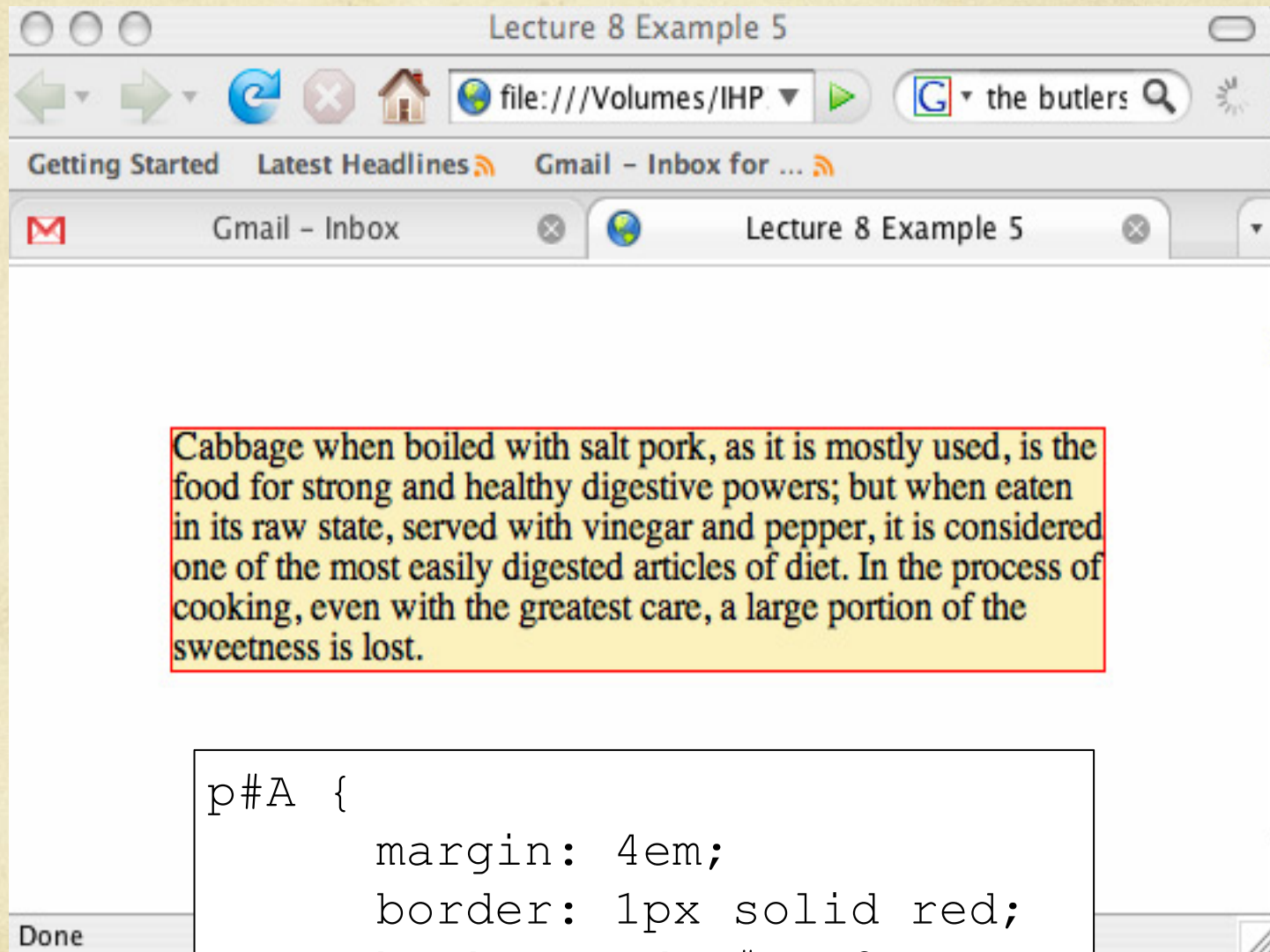
applies to: all elements

inherits: no

Margins

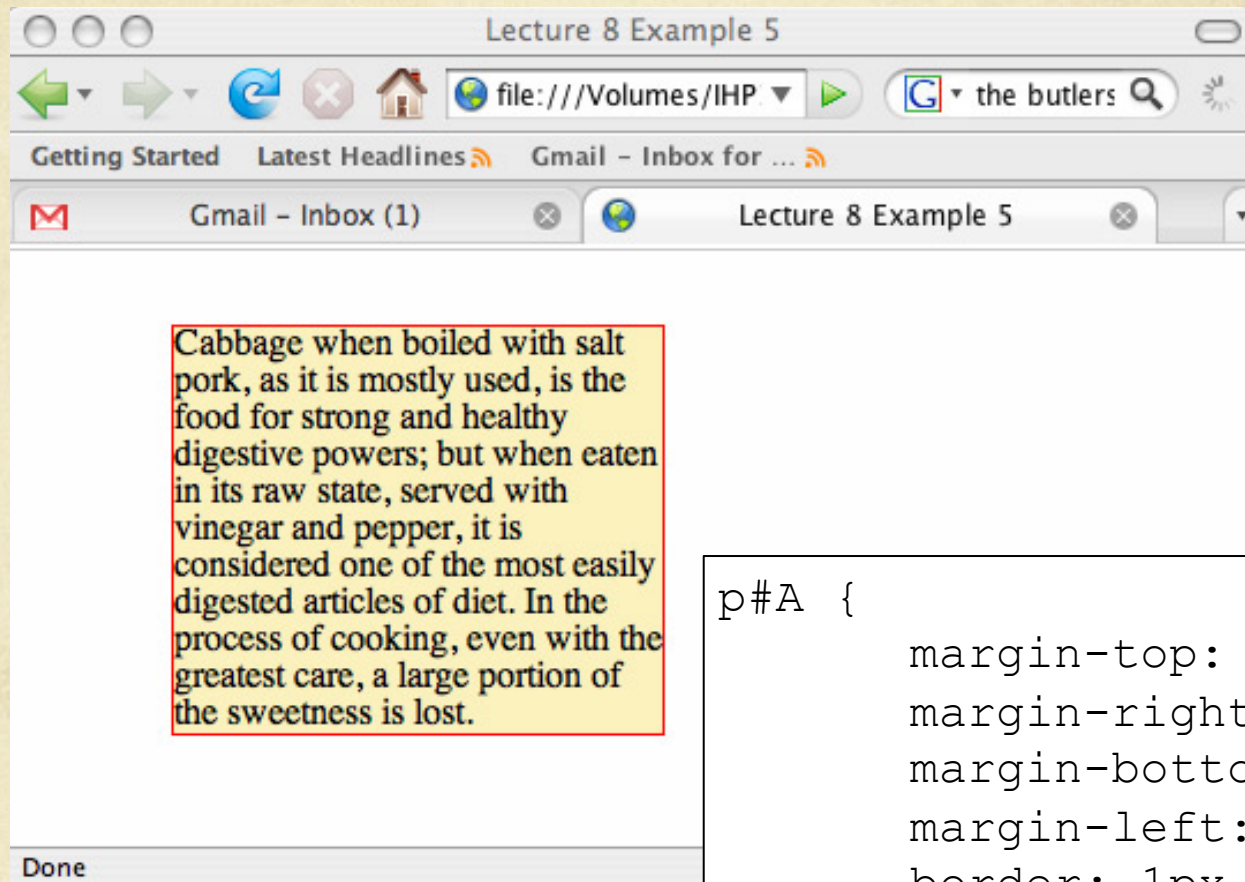
- To add margin just specify an amount
 - In general tend to use ems or pixels as units of measurement
- If we use **margin** it adds the same amount on all sides
- If want different amounts on different sides then use **margin-left, margin-top** etc.
- Following slides show some examples of adding margins to paragraph elements.

Margins



```
p#A {  
    margin: 4em;  
    border: 1px solid red;  
    background: #FCF2BE;  
}
```


Margins



```
p#A {  
    margin-top: 2em;  
    margin-right: 250px;  
    margin-bottom: 1em;  
    margin-left: 4em;  
    border: 1px solid red;  
    background: #FCF2BE;  
}
```



Margins

- Three things to be aware of with respect to margins:

1. Collapsing Margins

2. Margins on inline elements

3. Negative Margins



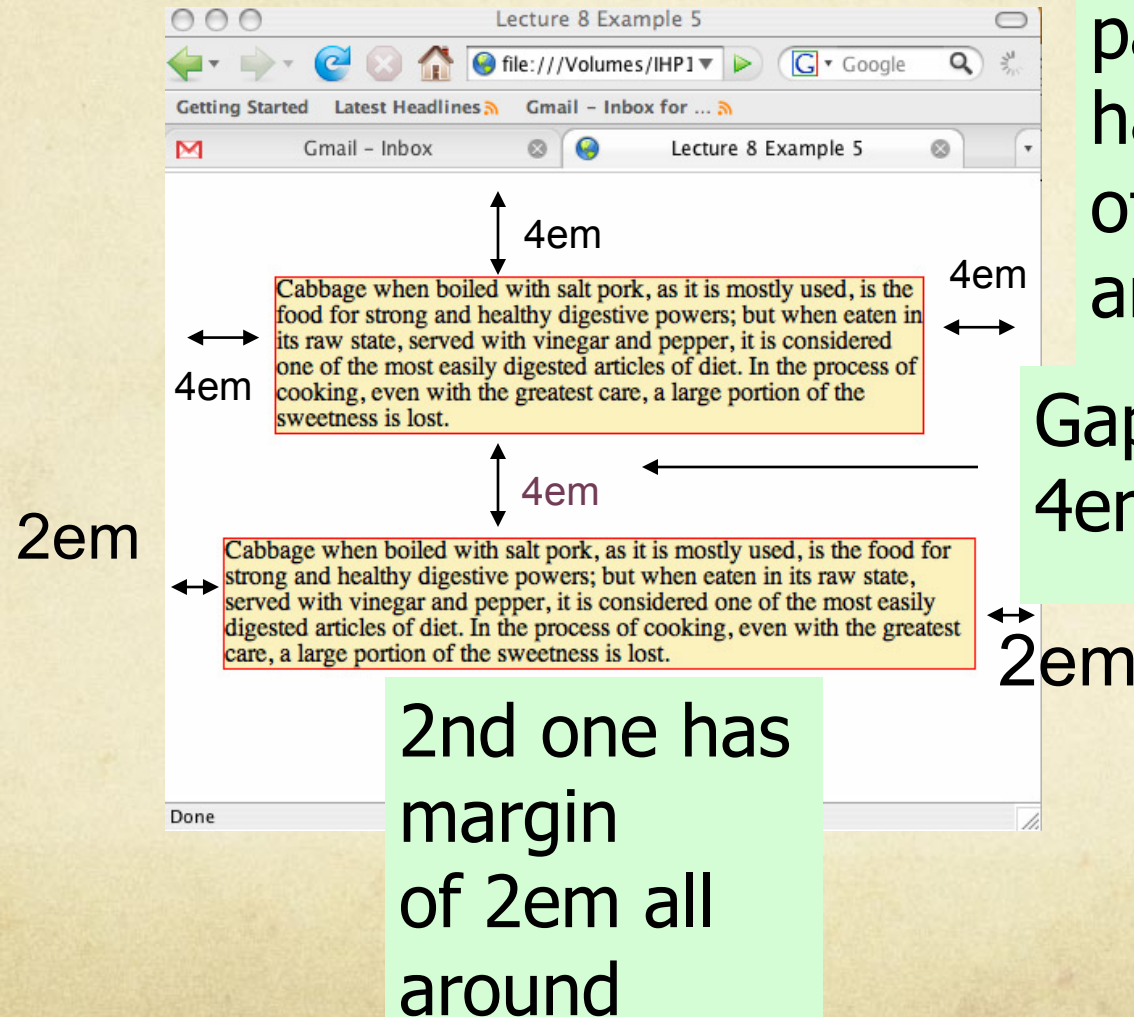
1. Collapsing Margins

Margins

1. Collapsing Margins

- The most important one is that *neighbouring ones collapse*.
- Suppose you have two adjacent elements that both have margins, for example two paragraphs.
- The gap between the elements will not be the sum of the two margins.
- The gap between the elements is equivalent to the larger margin i.e. they collapse into one another.
- Example on next slide.

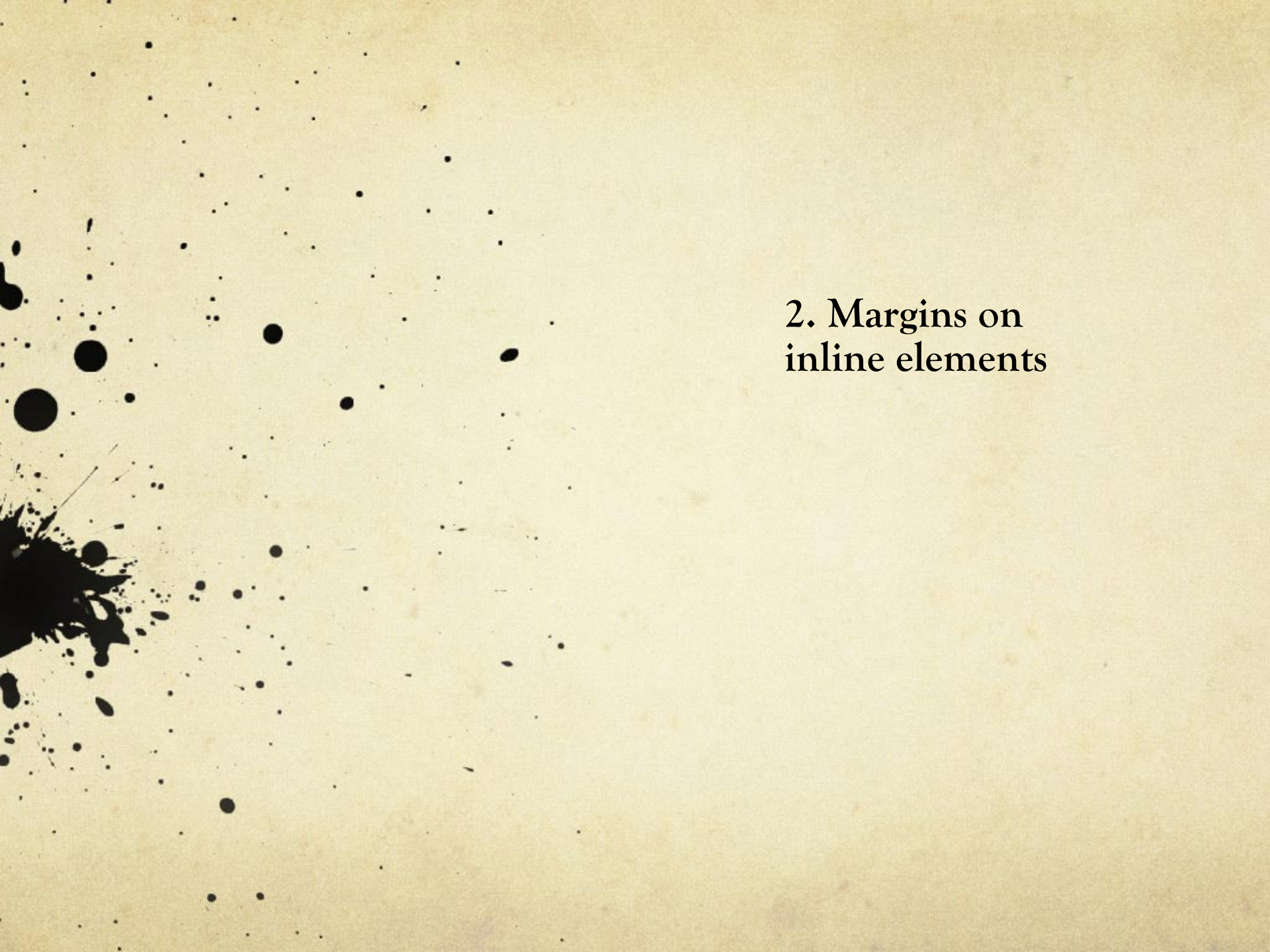
Margins



1st
paragraph
has margin
of 4em all
around

Gap is
4em!

2nd one has
margin
of 2em all
around



2. Margins on inline elements

Margins

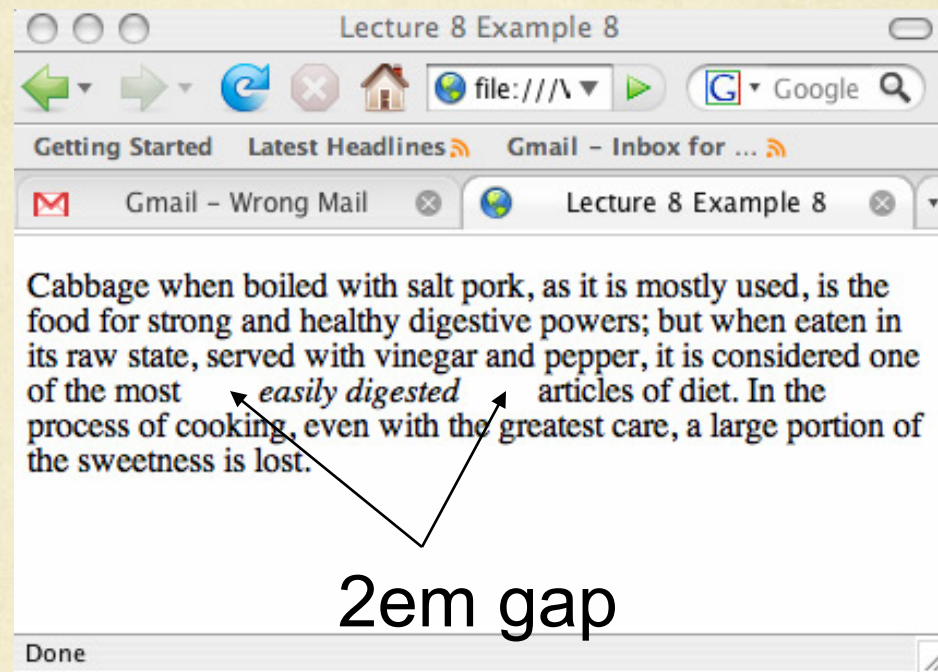
2. Margins on inline elements

- These behave a little differently to what you might expect.
- Basically, applying **top** and **bottom** margins will have no effect but applying **left** and **right** ones will.
- So suppose we make a style rule for an **em** element that includes an all round margin.

```
em { margin: 2em;}
```


- Overleaf shows what happens to an inline **em** element styled in this way.

Margins



Margins

- The exception to this behaviour is that image elements (**img**) will display margins top and bottom as well and left and right.



display: block;
display: inline;

You can CHANGE inline to be block elements and vice versa !!!!!!!!!!!

- Sometimes we want to ensure an element is displayed a particular way
 - E.g. we may wish to define width and height – we can only do this if an element is displayed as a BLOCK
 - So we may need to declare that such elements are displayed as block rather than their default INLINE
 - And vice versa
 - E.g. a list all on the same line rather than on separate lines

```
p { display: inline; }  
em { display: block; }
```

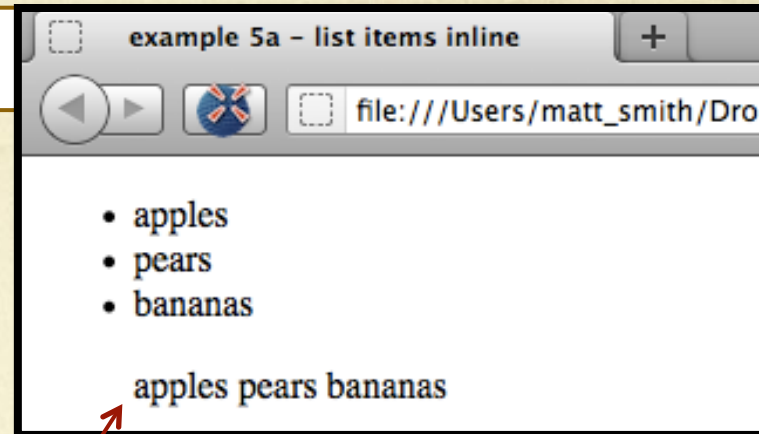
- The following 2 slides illustrate this concept
 - They are a little advanced
 - But we'll meet them soon to create nice navigation bars
 - So it's worth being aware that the BOX MODEL is important in terms of:
 - Can we set **HEIGHT** (BOX)
 - Can we have all on same line (INLINE)
 - Can we float (BOX)
- It will all make sense (hopefully!)

Display list items inline

```
.list2 li
{
    /*
    remove bullets from
    list items
    */
    list-style-type: none;

    /*
    make the list items
    display inline
    */
    display: inline;
}
```

CSS

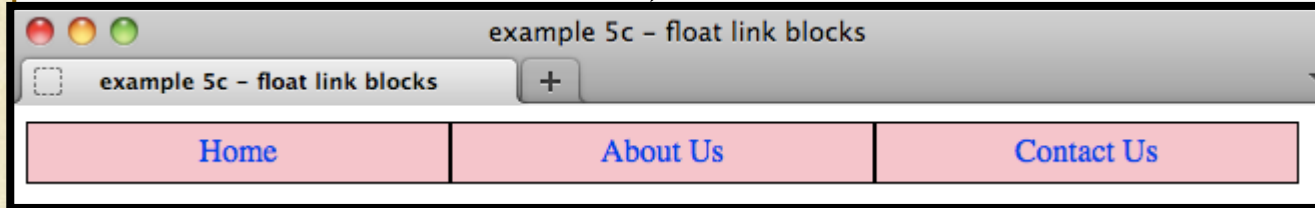


```
<ul>
  <li>apples</li>
  <li>pears</li>
  <li>bananas</li>
</ul>
<ul class="list2">
  <li>apples</li>
  <li>pears</li>
  <li>bananas</li>
</ul>
```

HTML

HTML

Make links blocks, so can float left



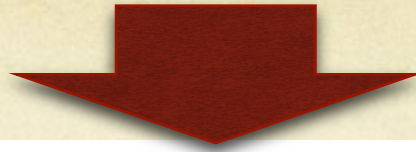
CSS

```
nav a
{
  display: block;
  width: 200px;
  border: 1px solid black;
  padding: 5px;
  background-color: pink;
  text-decoration: none;
  text-align: center;
  float: left;
}
```

```
nav li
{
  list-style-type: none;
  display: inline;
}
```

CSS

We will cover more on this next week....



I thought the layout of this site was simple and easy for new developers to create



[Work](#)

[Services](#)

[Resources](#)

[About us](#)

[Contact us](#)

01208 226522

Resources

[Troubleshooting](#)

[Bluffers guide](#)

[Build a better website](#)

[Hot topics](#)

Web page size and layout

Most web pages follow a common set of sizes and layouts, which you'll recognise even if you aren't aware of them. For example, did you know that nine out of the [top ten most popular websites](#) in the world use a fixed-width, centre-aligned design? Maybe you should too.

Note that this article is now a little old, for more information about modern design standards, check out our [Responsive Design](#) page.

Our article on web page size and layout breaks down web design into simple concepts, and helps you decide on a format for your website. To understand what the individual elements of a page are for, and why you need them, read our [anatomy of a web page](#).

But css3 is making life easier for us all the time with:

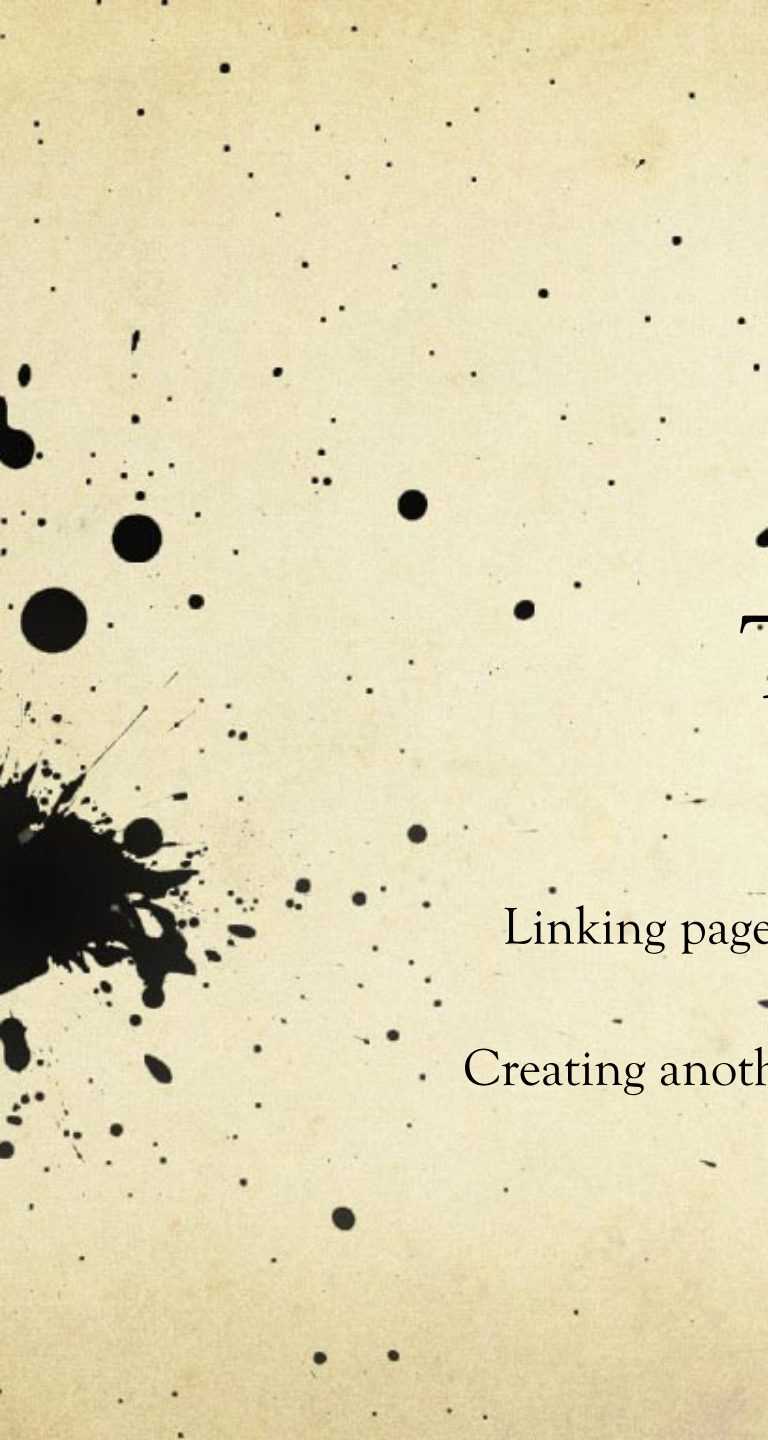
- Box-sizing
- Flex-boxes(later)

What next?

The navigation bar for your sites

Some practice with divs for layout

Some practice with the box model elements...padding,
borders ,margins and width/height



That's all folks

Next weeks lab:

Linking pages that do down a directory and back up a
directory.

Creating another couple of HTML pages and CSS for the
Blackgoose Website