CSS for Styling

The good, the bad and the… ugly!

```html
<p>
<font face="Arial">Welcome!</font>
News for <b>nerds!!</b> You will <i>never</i>,
<u>EVER</u> be
<font size="+4" color="red">BORED</font> here!
</p>
```

HTML

```
Welcome! News for nerds!! You will never, EVER be
BORED here!
```

output

- Tags such as b, i, u, and font are discouraged in strict XHTML
- Why is this bad?

Cascading Style Sheets (CSS)

- Describes the appearance, layout, and presentation of information on a web page
  - HTML describes **the content** of the page
- Describes *how* information is to be displayed, not *what* is being displayed
- Can be embedded in HTML document or placed into separate .css file

Basic CSS rule syntax

```
selector {
property: value;
property: value;
...
property: value;
}
            CSS
```

```
p {
font-family: sans-serif;
color: red;
}
            CSS
```

- A CSS file consists of one or more **rules**
- Each rule starts with a **selector**
- A selector specifies an HTML element(s) and then applies style **properties** to them
  - a selector of * selects all elements

Attaching a CSS file <link>

```
<head>
...
<link href="filename" type="text/css" rel="stylesheet" />
...
</head>
```

HTML

```
<link href="style.css" type="text/css" rel="stylesheet" />
<link href="http://www.google.com/uds/css/gsearch.css"
rel="stylesheet" type="text/css" />
```

HTML

- A page can link to multiple style sheet files
  - In case of a conflict (two sheets define a style for the same HTML element), the latter sheet's properties will be used

Embedding style sheets: <style>

```
<head>
<style type="text/css">
p { font-family: sans-serif; color: red; }
h2 { background-color: yellow; }
</style>
</head>
```

HTML

- CSS code can be embedded within the head of an HTML page
- B*ad style* and should be avoided when possible (why?)

Inline styles: the style attribute

```html
<p style="font-family: sans-serif; color: red;">
This is a paragraph</p>
```

HTML

```
This is a paragraph
```

output

- Higher precedence than embedded or linked styles
- Used for one-time overrides and styling a particular element
- B*ad style* and should be avoided when possible (why?)

CSS properties for colors

```
p {
color: red;
background-color: yellow;
}
```

CSS

```
This paragraph uses the style above
```

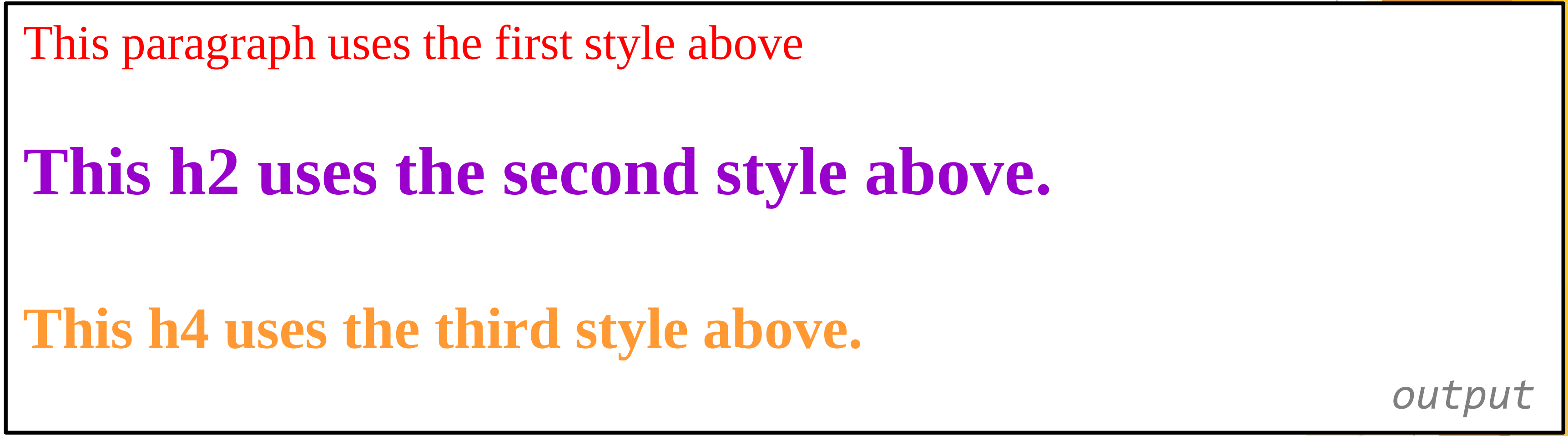
output

property	description
color	color of the element's text
background-color	color that will appear behind the element

Specifying colors

```
p { color: red; }
h2 { color: rgb(128, 0, 196); }
h4 { color: #FF8800; }
```

CSS



This paragraph uses the first style above

This h2 uses the second style above.

This h4 uses the third style above.

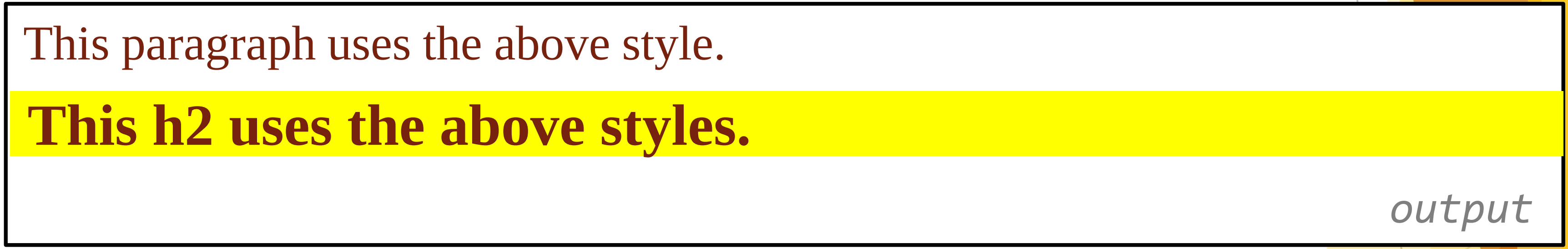
output

- color names: aqua, black, blue, fuchsia, gray, green, lime, maroon, navy, olive, purple, red, silver, teal, white (white), yellow
- RGB codes: red, green, and blue values from 0 (none) to 255 (full)
- hex codes: RGB values in base-16 from 00 (0, none) to FF (255, full)

Grouping styles

```
p, h1, h2 {
color: green;
}
h2 {
background-color: yellow;
}
```

CSS



This paragraph uses the above style.

This h2 uses the above styles.

output

- A style can select multiple elements separated by commas
- The individual elements can also have their own styles

CSS comments /*…*/

```
/* This is a comment.
It can span many lines in the CSS file. */
p {
color: red; background-color: aqua;
}                                                  CSS
```

- CSS (like HTML) is usually not commented as rigorously as programming languages such as Java
- The // single-line comment style is NOT supported in CSS
- The <!-- ... --> HTML comment style is also NOT supported in CSS

CSS properties for fonts

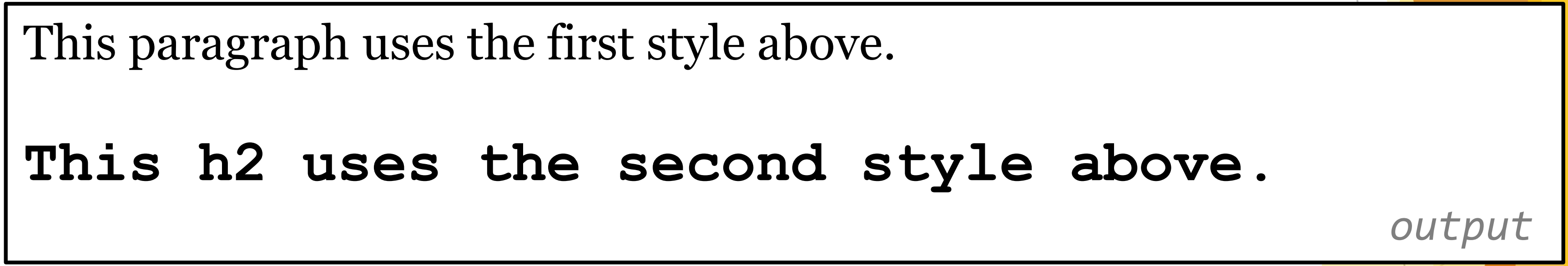
property	description
font-family	which font will be used
font-size	how large the letters will be drawn
font-style	used to enable/disable italic style
font-weight	used to enable/disable bold style

Complete list of font properties (http://www.w3schools.com/css/css_reference.as

font-family

```css
p {
font-family: Georgia;
}
h2 {
font-family: "Courier New";
}
```

CSS



This paragraph uses the first style above.

This h2 uses the second style above.

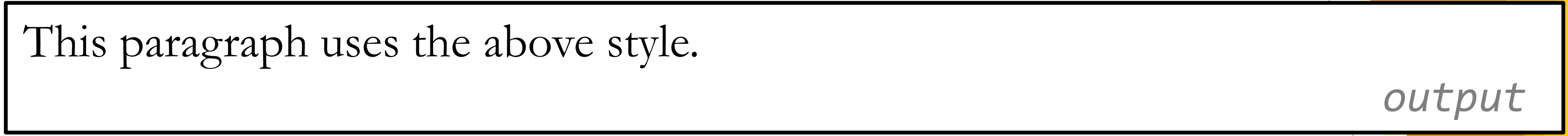
output

- Enclose multi-word font names in quotes

More about font-family

```
p {
font-family: Garamond, "Times New Roman", serif;
}
```

CSS



This paragraph uses the above style.

output

- We can specify multiple fonts from highest to lowest priority
- Generic font names:
  - serif, sans-serif, cursive, FANTASY, monospace
- If the first font is not found on the user's computer, the next is tried
- Placing a generic font name at the end of your font-family value, ensures that every computer will use a valid font

font-size

```
p {
        font-size: 24pt;
}                                                          CSS
```

```
This paragraph uses the style above.
                                                        output
```

- units: pixels (px) vs. point (pt) vs. m-size (em)

16px, 16pt, 1.16em

- vague font sizes: xx-small, x-small, small, medium, large, x-large, xx-large, smaller, larger
- percentage font sizes, e.g.: 90%, 120%

font-size

```
p {
        font-size: 24pt;
}
```

CSS

This paragraph uses the style above.

output

- pt specifies number of point, where a point is 1/72 of an inch onscreen
- px specifies a number of pixels on the screen
- em specifies number of m-widths, where 1 em is equal to the font's current size

font-weight, font-style

```
p {
font-weight: bold;
font-style: italic;
}
```

CSS

This paragraph uses the style above.

output

- Either of the above can be set to normal to turn them off (e.g. headings)

CSS properties for text

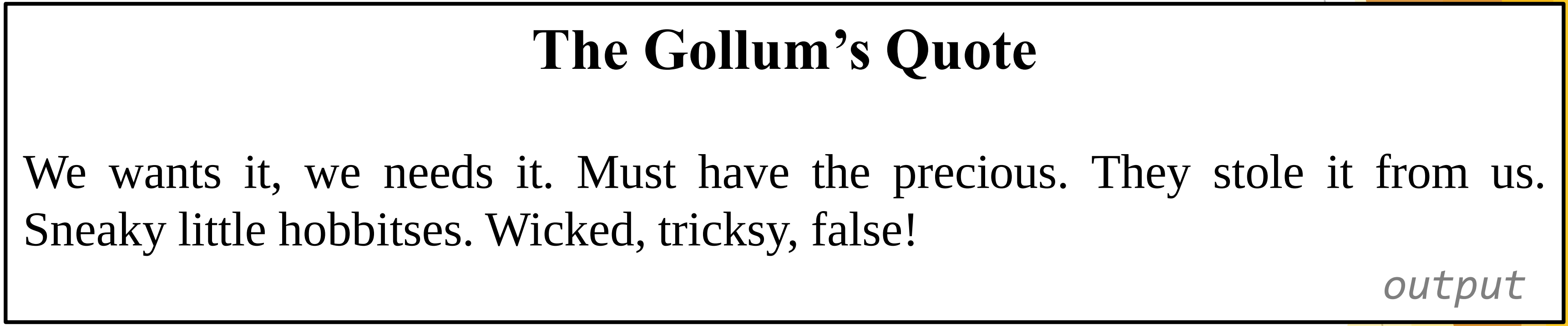
property	description
text-align	alignment of text within its element
text-decoration	decorations such as underlining
line-height, word-spacing, letter-spacing	gaps between the various portions of the text
text-indent	indents the first letter of each paragraph

Complete list of text properties (http://www.w3schools.com/css/css_reference.as

text-align

```
blockquote { text-align: justify; }
h2 { text-align: center; }
```

CSS



The Gollum's Quote

We wants it, we needs it. Must have the precious. They stole it from us. Sneaky little hobbitses. Wicked, tricksy, false!

output

- text-align can be `left`, `right`, `center`, or `justify`

text-decoration

```
p {
text-decoration: underline;
}
```

CSS

This paragraph uses the style above.

output

- can also be `overline`, ~~`line-through`~~, `blink`, or `none`
- effects can be combined:

```
text-decoration: overline underline;
```

The list-style-type property

```css
ol { list-style-type: lower-roman; }
```

CSS

- Possible values:

  i. `none` : No marker

  ii. `disc` (default), `circle`, `square`

  iii. `Decimal`: 1, 2, 3, etc.

  iv. `decimal-leading-zero`: 01, 02, 03, etc.

  v. `lower-roman`: i, ii, iii, iv, v, etc.

  vi. `upper-roman`: I, II, III, IV, V, etc.

  vii. `lower-alpha`: a, b, c, d, e, etc.

  viii. `upper-alpha`: A, B, C, D, E, etc.

  x. `lower-greek`: alpha, beta, gamma, etc.

  others: hebrew, armenian, georgian, cjk-ideographic, hiragana…

Body styles

```
body {
font-size: 16px;
}
```

CSS

- Applies a style to the entire body of your page
- Saves you from manually applying a style to each element

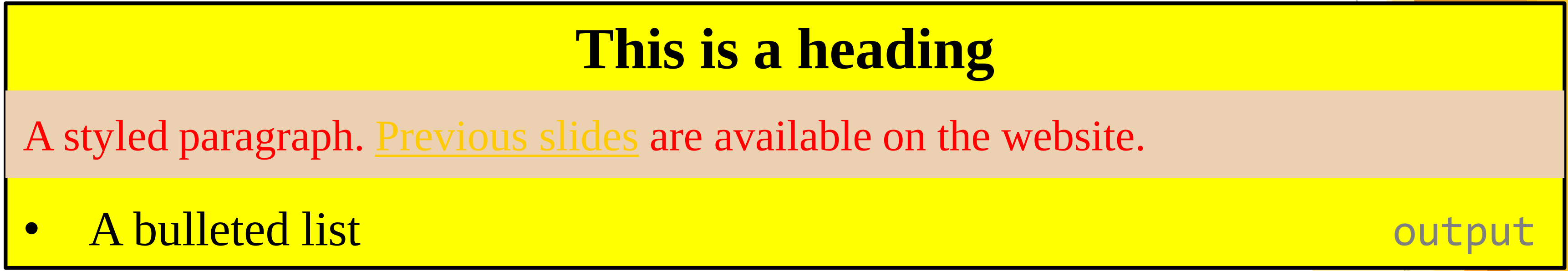
Cascading Style Sheets

- Properties of an element *cascade* together in this order:
  - browser's default styles
  - external style sheet files (in a <link> tag)
  - internal style sheets (inside a <style> tag in the page's header)
  - inline style (the style attribute of the HTML element)

Inheriting styles

```
body { font-family: sans-serif; background-color: yellow;
}
p { color: red; background-color: aqua; }
a { text-decoration: underline; }
h2 { font-weight: bold; text-align: center; }
```

CSS



This is a heading

A styled paragraph. Previous slides are available on the website.

- A bulleted list

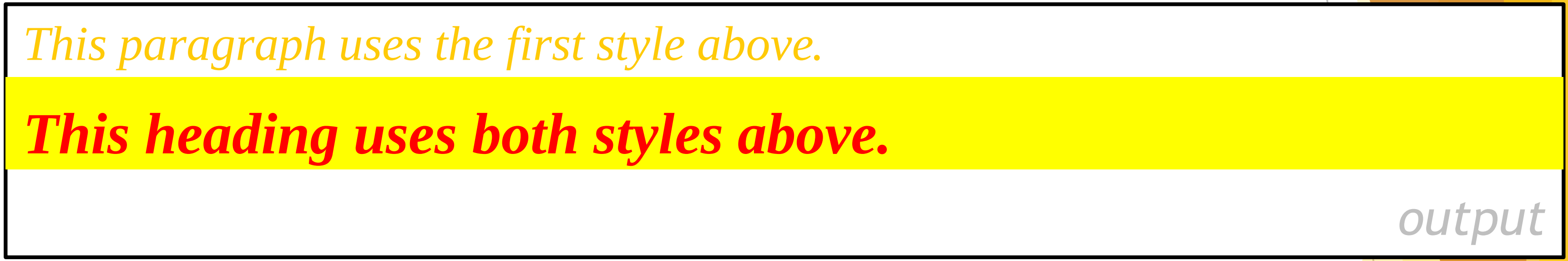
output

- when multiple styles apply to an element, they are inherited
- a more tightly matching rule can override a more general inherited rule

Styles that conflict

```css
p, h1, h2 { color: blue; font-style: italic; }
h2 { color: red; background-color: yellow; }
```

CSS



This paragraph uses the first style above.

This heading uses both styles above.

output

- when two styles set conflicting values for the same property, the latter style takes precedence

W3C CSS Validator

```
<p>
<a href="http://jigsaw.w3.org/css-
validator/check/referer">
<img src="http://jigsaw.w3.org/css-validator/images/vcss"
alt="Valid CSS!" /></a>
</p>
```

CSS

output

- jigsaw.w3.org/css-validator/
- checks your CSS to make sure it meets the official CSS specifications

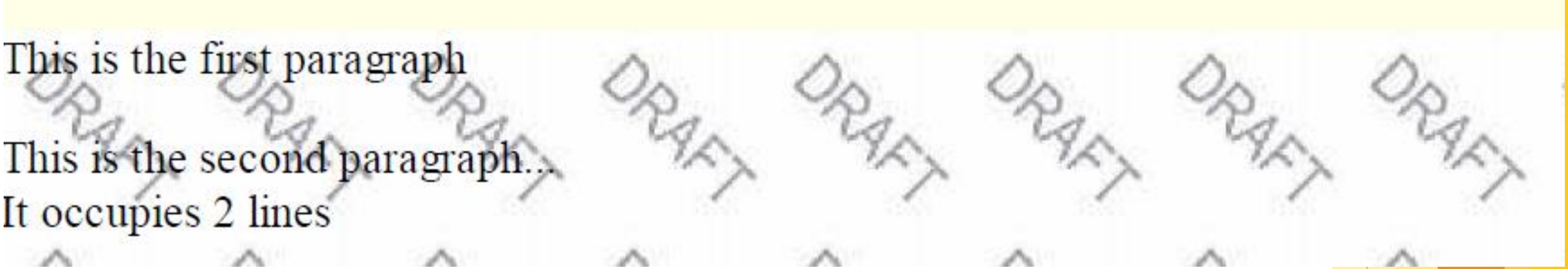
CSS properties for backgrounds

property	description
background-color	color to fill background
background-image	image to place in background
background-position	placement of bg image within element
background-repeat	whether/how bg image should be repeated
background-attachment	whether bg image scrolls with page
background	shorthand to set all background properties

background-image

```
body {
background-image: url("images/draft.jpg");
}
```

CSS



This is the first paragraph

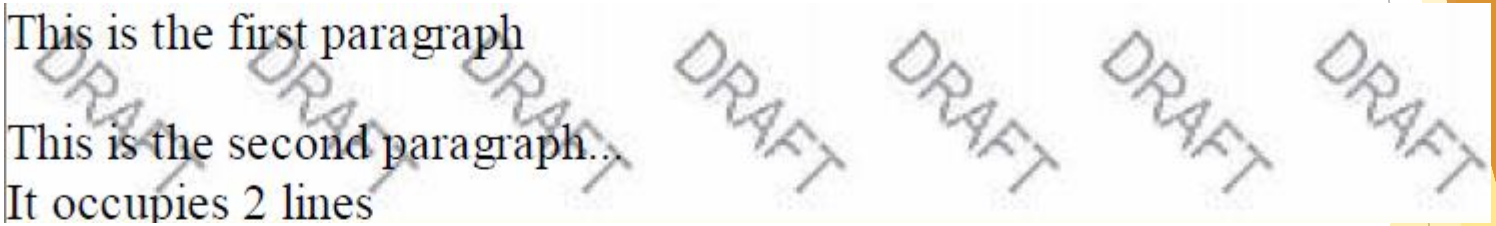
This is the second paragraph...
It occupies 2 lines

- background image/color fills the element's content area

background-repeat

```
body {
background-image: url("images/draft.jpg");
background-repeat: repeat-x;
}
```

CSS



This is the first paragraph

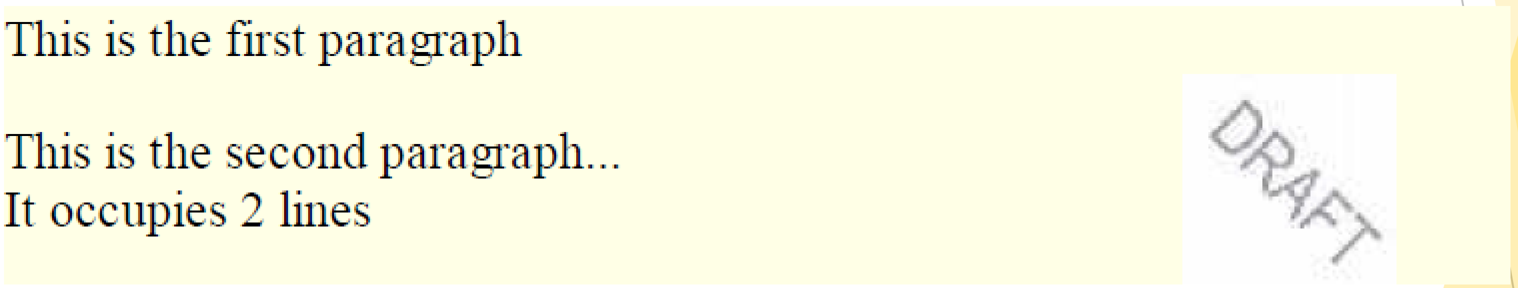
This is the second paragraph...
It occupies 2 lines

- can be repeat (default), repeat-x, repeat-y, or no-repeat

background-position

```
body {
background-image: url("images/draft.jpg");
background-repeat: no-repeat;
background-position: 370px 20px;
}
```

CSS



This is the first paragraph

This is the second paragraph...
It occupies 2 lines

- value consists of two tokens, each of which can be top, left, right, bottom, center, a percentage, or a length value in px, pt, etc.
- value can be negative to shift left/up by a given amount

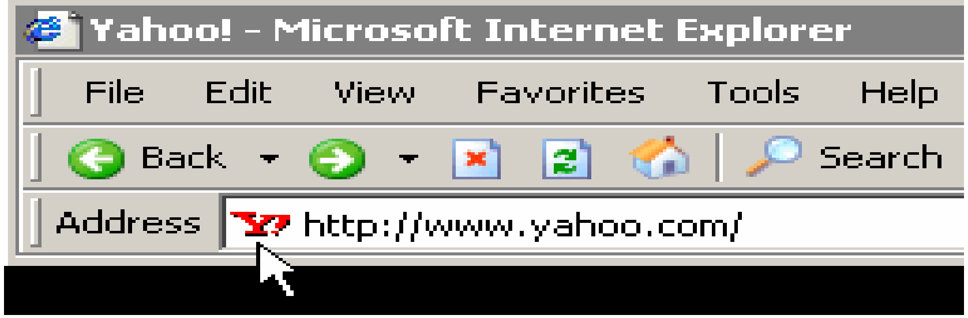
Aside: Favorites icon ("favicon")

```
<link href="filename" type="MIME type" rel="shortcut icon"
/>
```

HTML

```
<link href="yahoo.gif" type="image/gif" rel="shortcut
icon" />
```

HTML



http://www.yahoo.com/

- The link tag, placed in the HTML page's head section, can specify an icon
    - this icon will be placed in the browser title bar and bookmark/favorite

More CSS

```
<link href="filename" type="text/css"
rel="stylesheet" />
```

HTML id attribute

```
<p>Coding Horror! Coding Horror!</p>
<p id="mission">Our mission is to combine programming and
<q>human</q> factors with geekiness!</p>
```

HTML

```
Coding Horror! Coding Horror!

Our mission is to combine programming and “human” factors with geekiness!
```

output

- A unique ID for an element on a page
- Each ID must be unique; can only be used once in the page

Linking to sections of a web page

```
<p>Visit <a href=
"http://www.textpad.com/download/index.html#downloads">
textpad.com</a> to get the TextPad editor.</p>
<p><a href="#mission">View our Mission Statement</a></p>
```

HTML

```
Visit textpad.com to get the TextPad editor.

View our Mission Statement
```

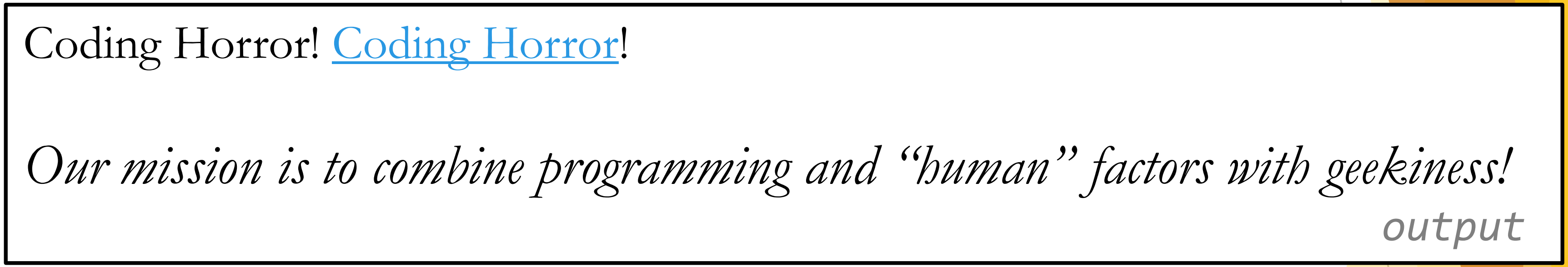
output

- Link target can include an ID at the end, preceded by a #
- Browser will load that page and scroll to element with given ID

CSS ID selectors

```
#mission {
font-style: italic;
font-family: "Garamond", "Century Gothic", serif;
}
```

CSS



Coding Horror! Coding Horror!

Our mission is to combine programming and "human" factors with geekiness!

output

- Applies style only to the paragraph that has the ID of mission

HTML class attribute

```
<p class="shout">Coding Horror! Coding Horror!</p>
<p class="special">See our special deal on Droids!</p>

<p class="special">Today only!</p>                  HTML
```

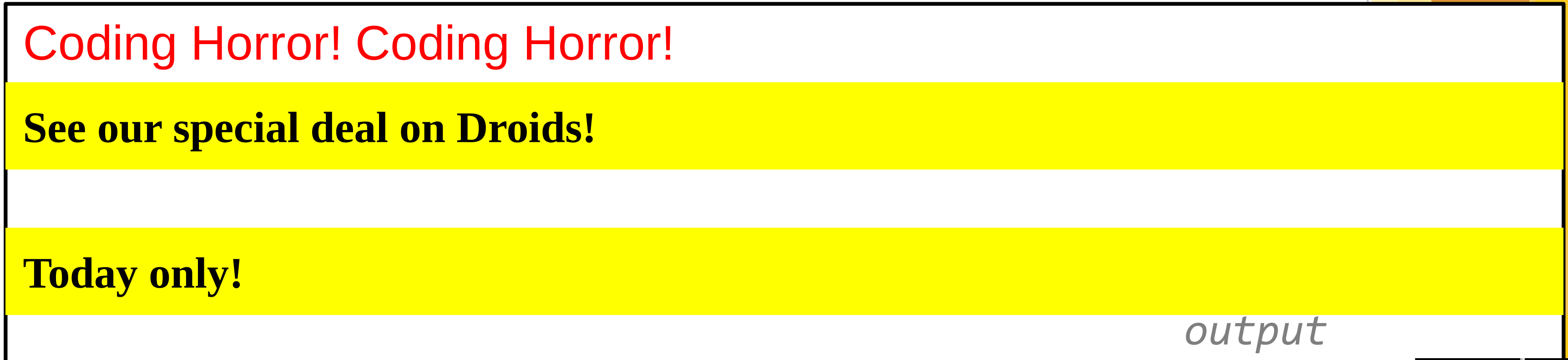
```
Coding Horror! Coding Horror!

See our special deal on Droids!

Today only!                                          output
```

- A way to group some elements and give a style to only that group
- Unlike an id, a class can be reused as much as you like on the page

CSS class selectors

```
.special {
background-color: yellow;
font-weight: bold;
}
p.shout {
color: red;
font-family: cursive;
}
```

CSS



Coding Horror! Coding Horror!

See our special deal on Droids!

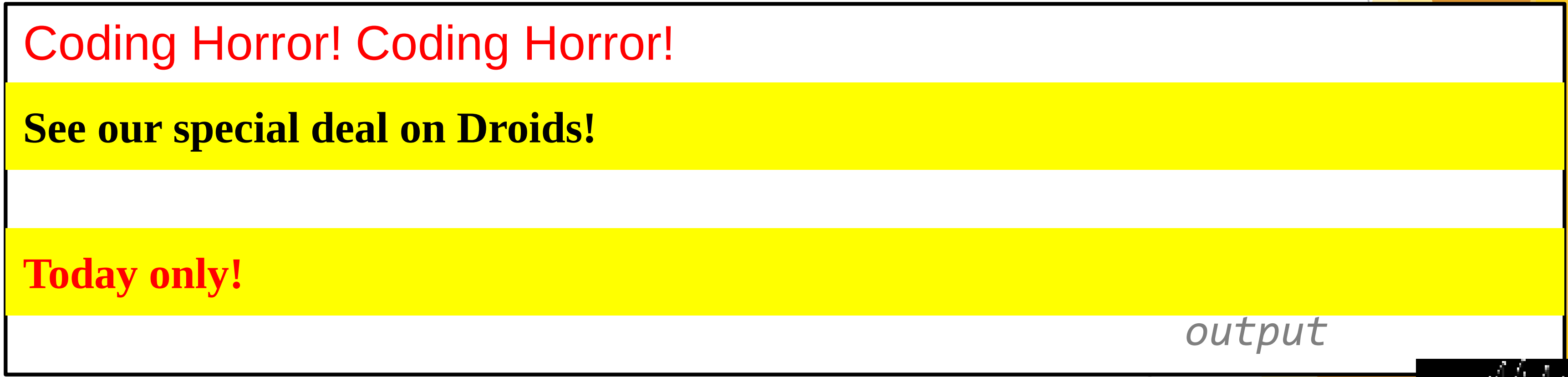
Today only!

output

CSS class selectors

```html
<p class="shout">Coding Horror! Coding Horror!</p>
<p class="special">See our special deal on Droids!</p>

<p class="special shout">Today only!</p>
```

HTML



Coding Horror! Coding Horror!

See our special deal on Droids!

Today only!

output

CSS ID selectors

```
a:link { color: #FF0000; } /* unvisited link */
a:visited { color: #00FF00; } /* visited link */
a:hover { color: #FF00FF; } /* mouse over link */
```

CSS

Buy Early Buy Often!

output

CSS ID selectors

class	description
:active	an activated or selected element
:focus	an element that has the keyboard focus
:hover	an element that has the mouse over it
:link	a link that has not been visited
:visited	a link that has already been visited
:first-letter	the first letter of text inside an element
:first-line	the first line of text inside an element
:first-child	an element that is the first one to appear inside another

Styling Page Sections

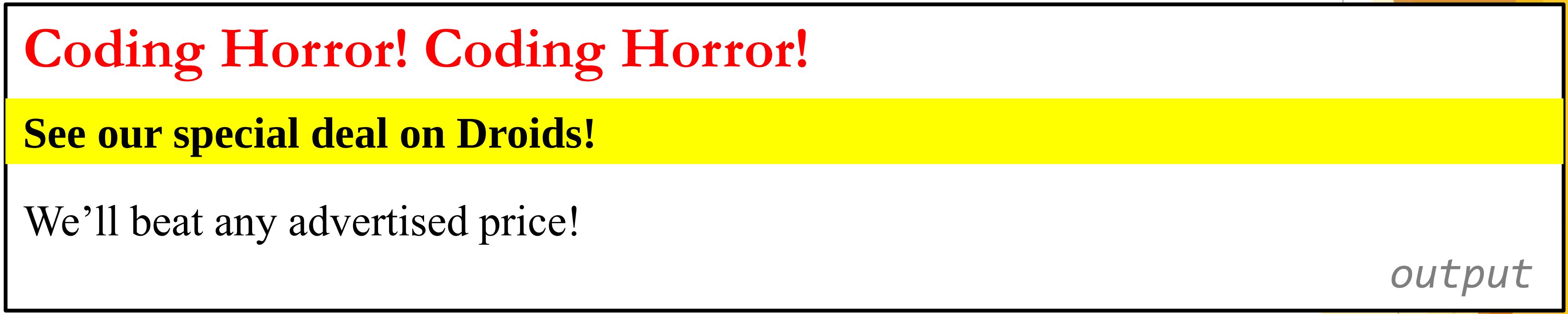
Why do we need page sections?

- Style individual elements, groups of elements, sections of text or of the page
- Create complex page layouts

Sections of a page <div>

```
<div class="shout">
<h2>Coding Horror! Coding Horror!</h2>
<p class="special">See our special deal on Droids!</p>
<p>We'll beat any advertised price!</p>
</div>
```

HTML



Coding Horror! Coding Horror!

See our special deal on Droids!

We’ll beat any advertised price!

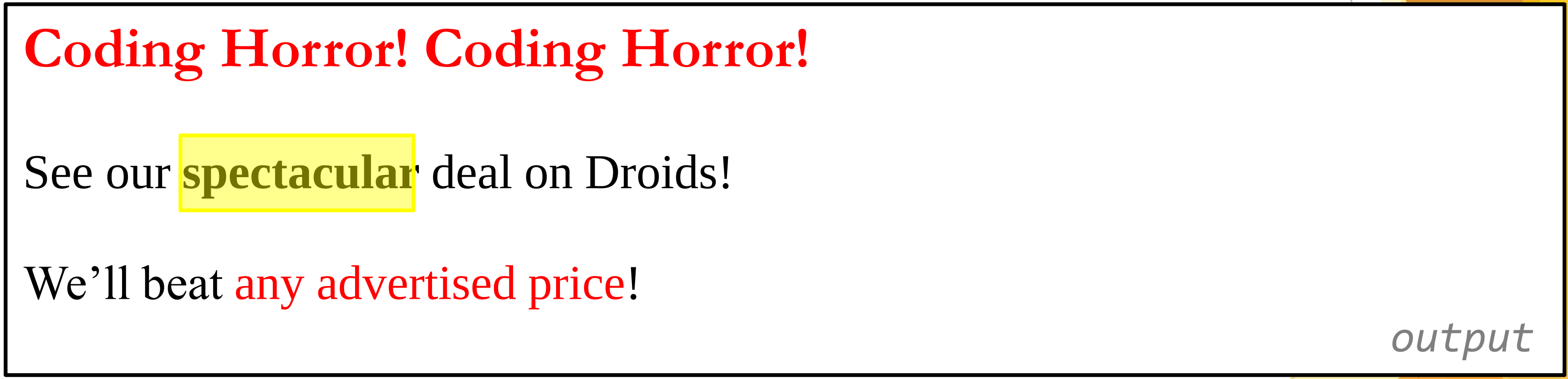
output

- Tag used to indicate a logical section or area of a page
- Has no appearance by default, but you can apply styles to it

Inline Sections <span>

```
<h2>Coding Horror! Coding Horror!</h2>
<p>See our <span class="special“>spectacular</span> deal
on Droids!</p>
<p>We'll beat <span class="shout“> any advertised
price</span>!</p>
```

HTML



Coding Horror! Coding Horror!

See our **spectacular** deal on Droids!

We’ll beat any advertised price!

output

- has no onscreen appearance, but you can apply a style or ID to it, which will be applied to the text inside span

CSS context selectors

```
selector1 selector2 {
properties
}
```

CSS

- applies the given properties to selector2 only if it is inside a selector1 on the page

```
selector1 > selector2 {
properties
}
```

CSS

□ applies the given properties to selector2 only if it is *directly* inside a selector1 on the page

Context selector example

```
<p>Eat at <strong>Greasy's Burger</strong>...</p>
<ul>
<li>The <strong>greasiest</strong> burgers in town!</li>
<li>Yummy and greasy at the same time!</li>
</ul>
```

HTML

```
li strong { text-decoration: underline; }
```

CSS

Eat at **Greasy's Burger**…

- The **greasiest** burgers in town!
- Yummy and greasy at the same time!

output

More complex example

```
<div id="ad">
<p>Eat at <strong>Greasy's Burger</strong>...</p>
<ul>
<li class="important">The <strong>greasiest</strong>
burgers in town!</li>
<li>Yummy and <strong>greasy at the same time
</strong>!</li>
</ul>
</div>
```

HTML

```
#ad li.important strong { text-decoration: underline; }
```

CSS

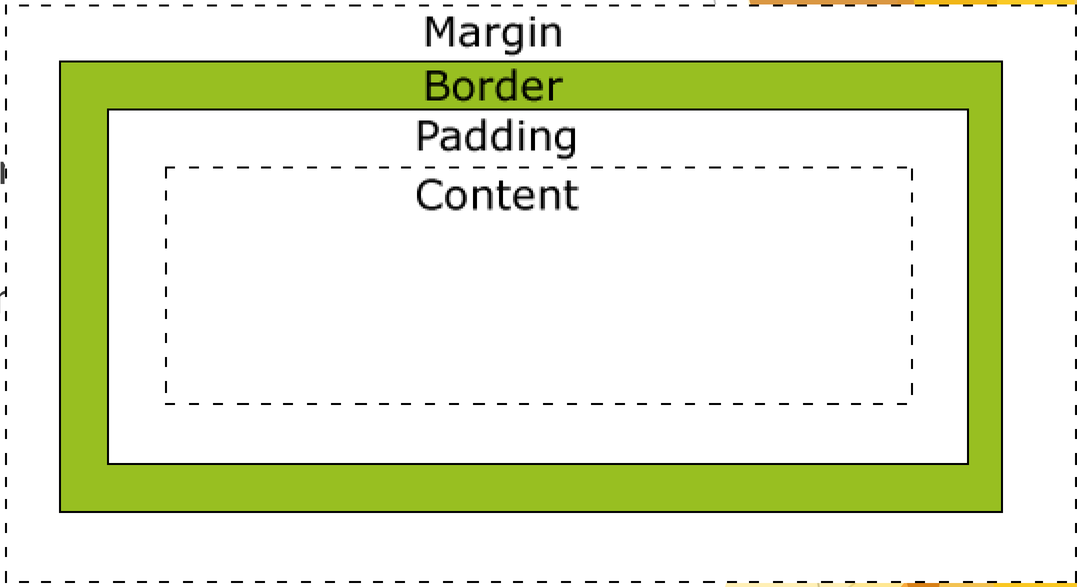
Eat at **Greasy’s Burger**…

- The **greasiest** burgers in town!
- Yummy and **greasy at the same time**!

output

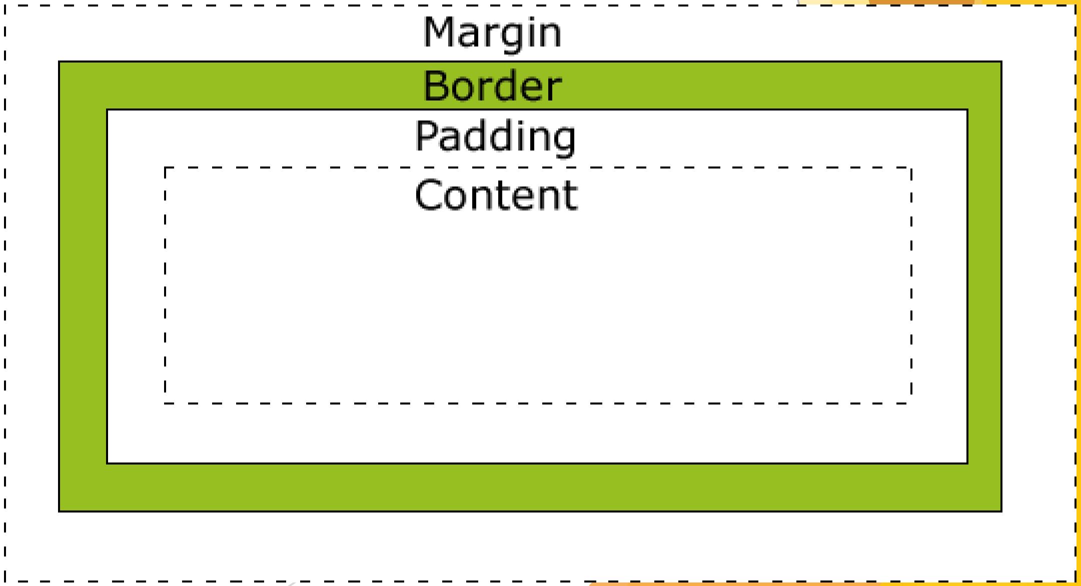
The CSS Box Model

- Every element composed of:
  - content
  - a border around the element
  - padding between the content ar the border
  - a margin between the border ar other content

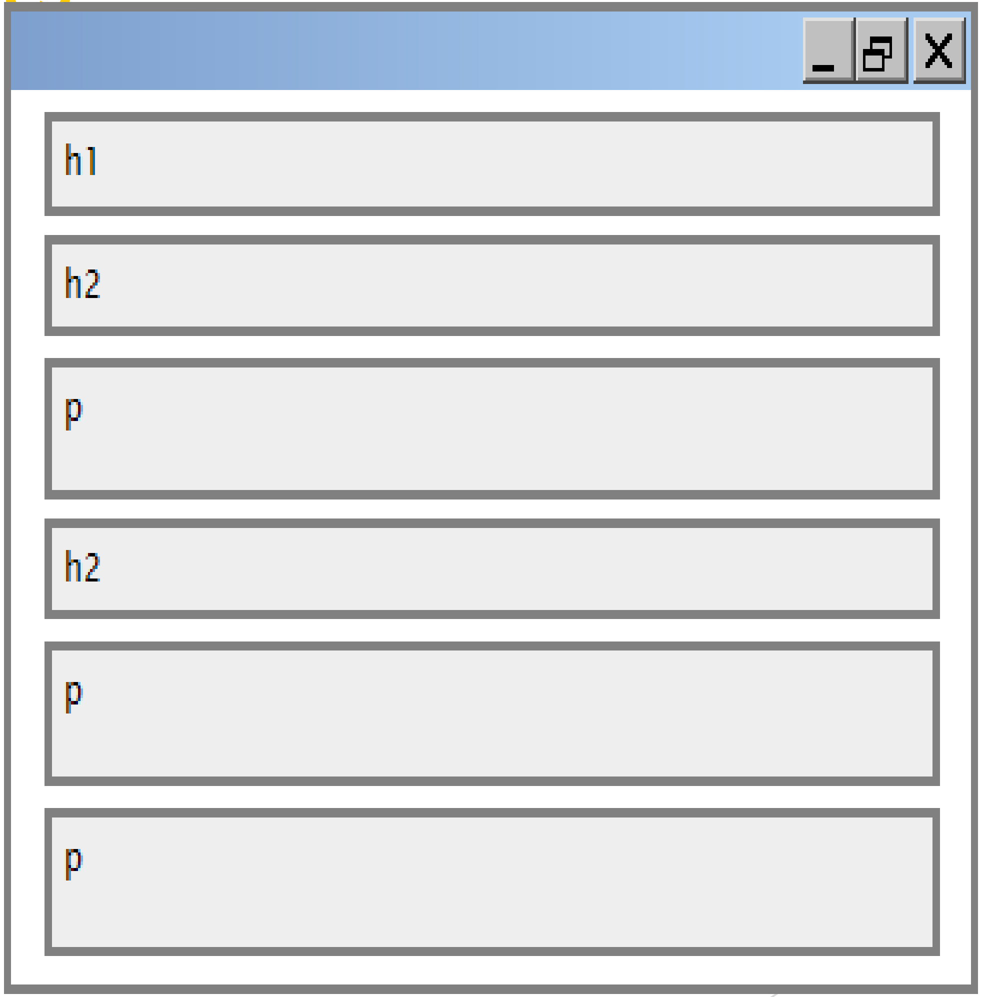


The CSS Box Model (cont.)

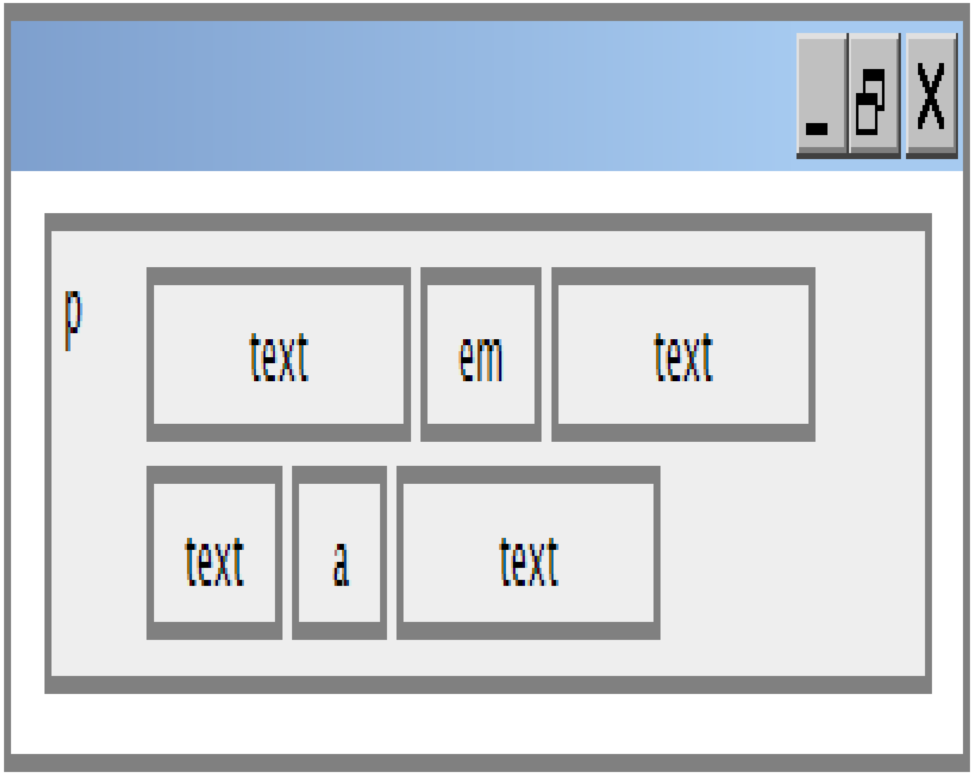
- width = content width + L/R padding + L/R border + L/R margin
- height = content height + T/B padding + T/B border + T/B margin
- IE6 doesn't do this right



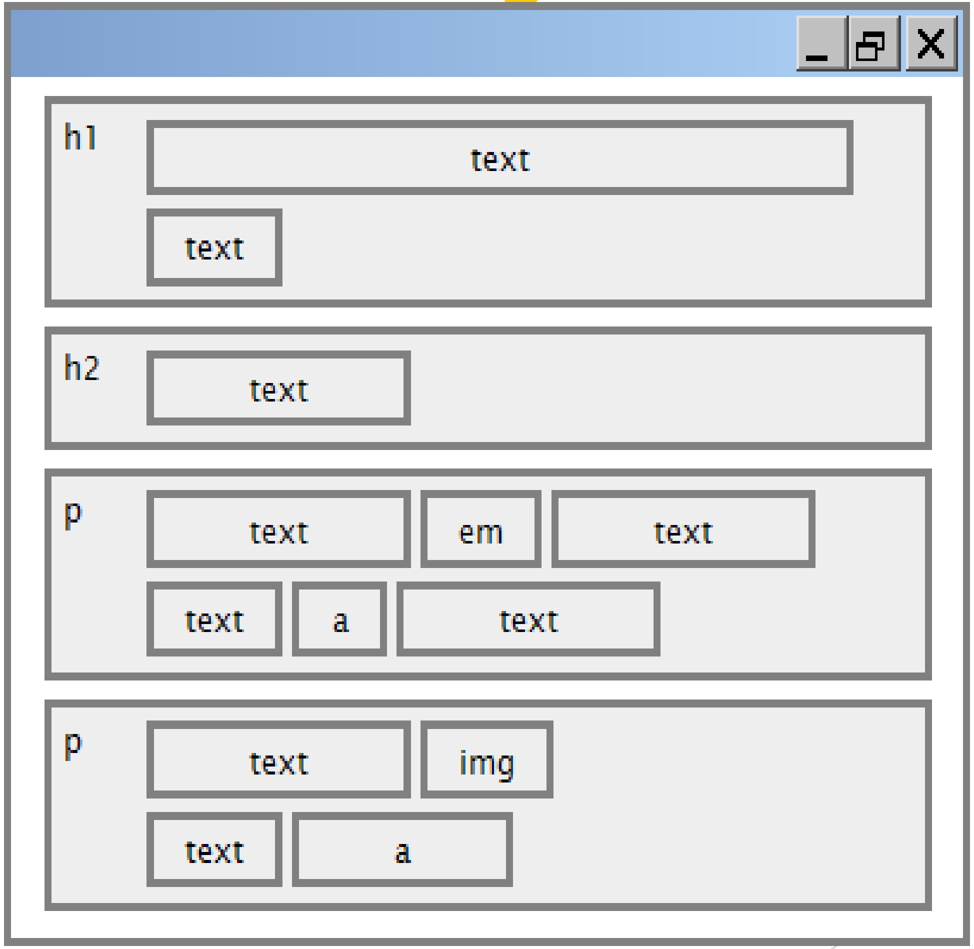
Document Flow – block elements



Document flow - inline elements



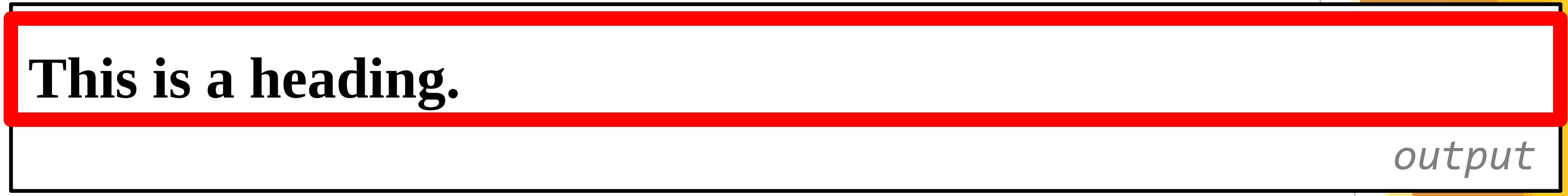
Document flow - a larger



CSS properties for borders

```
h2 { border: 5px solid red; }
```

CSS



This is a heading.

output

property	description
border	thickness/style/size of border on all 4 sides

- □ Thickness: `px, pt, em,` or `thin, medium, thick`
- □ Style: `none, hidden, dotted, dashed, double, groove, inset, outset, ridge, solid`
- □ color

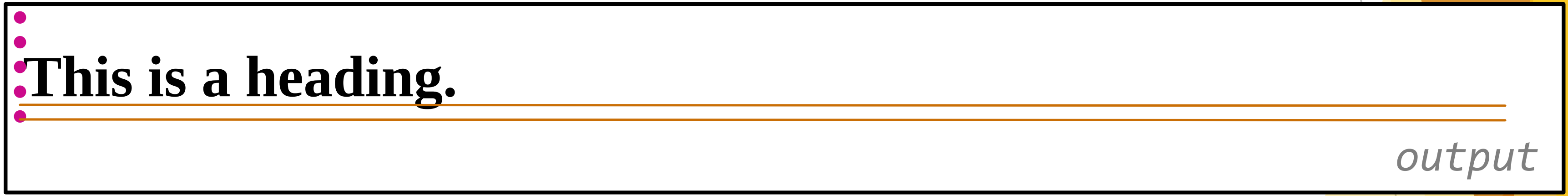
More border properties

property	description
border-color, border-width, border-style	specific properties of border on all 4 sides
border-bottom, border-left, border-right, border-top	all properties of border on a particular side
border-bottom-color, border-bottom-style, border-bottom-width, border-left-color, border-left-style, border-left-width, border-right-color, border-right-style, border-right-width, border-top-color, border-top-style, border-top-width	properties of border on a particular side
Complete list of border properties http://www.w3schools.com/css/css_reference.asp#border	

Another border example

```
h2 {
border-left: thick dotted #CC0088;
border-bottom-color: rgb(0, 128, 128);
border-bottom-style: double;
}
```

CSS



This is a heading.

output

- □ each side's border properties can be set individually
- □ if you omit some properties, they receive default

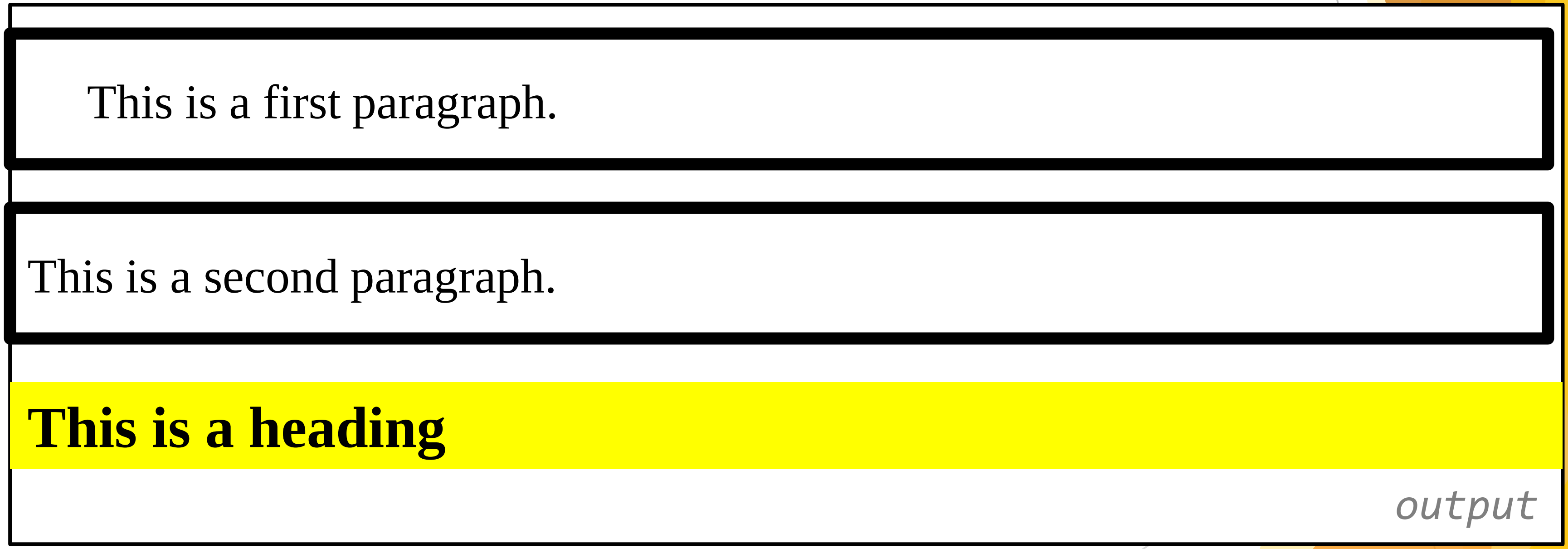
CSS properties for padding

property	description
padding	padding on all 4 sides
padding-bottom	padding on bottom side only
padding-left	padding on left side only
padding-right	padding on right side only
padding-top	padding on top side only
Complete list of padding properties http://www.w3schools.com/css/css_reference.asp#padding	

Padding example 1

```css
p { padding: 20px; border: 3px solid black; }
h2 { padding: 0px; background-color: yellow; }
```

CSS



This is a first paragraph.

This is a second paragraph.

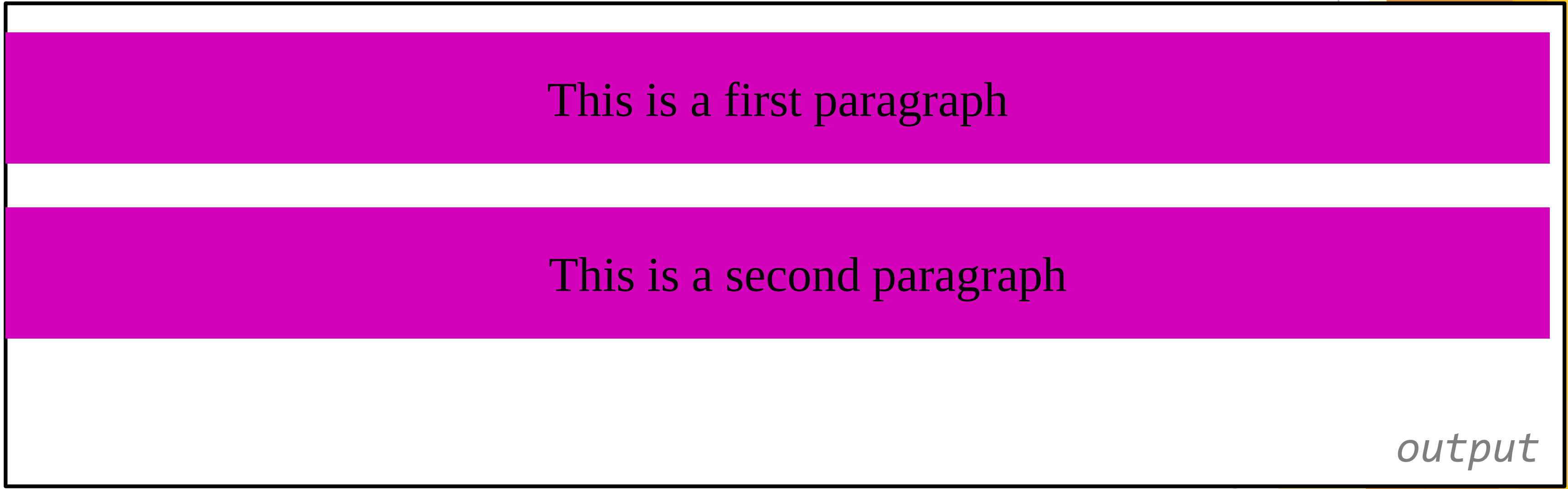
This is a heading

output

Padding example 2

```
p {
padding-left: 200px; padding-top: 30px;
background-color: fuchsia;
}
```

CSS



This is a first paragraph

This is a second paragraph

output

- □ each side's padding can be set individually
- □ notice that padding shares the background color of the element

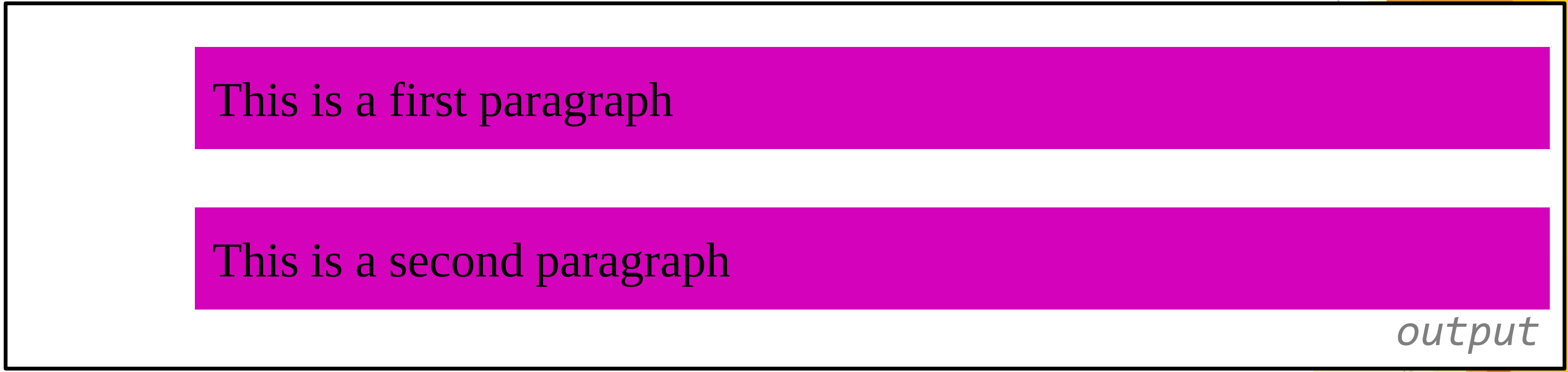
CSS properties for margins

property	description
margin	margin on all 4 sides
margin-bottom	margin on bottom side only
margin-left	margin on left side only
margin-right	margin on right side only
margin-top	margin on top side only
Complete list of margin properties http://www.w3schools.com/css/css_reference.asp#margin	

Margin example 1

```
p {
margin: 50px;
background-color: fuchsia;
}
```

CSS



This is a first paragraph

This is a second paragraph

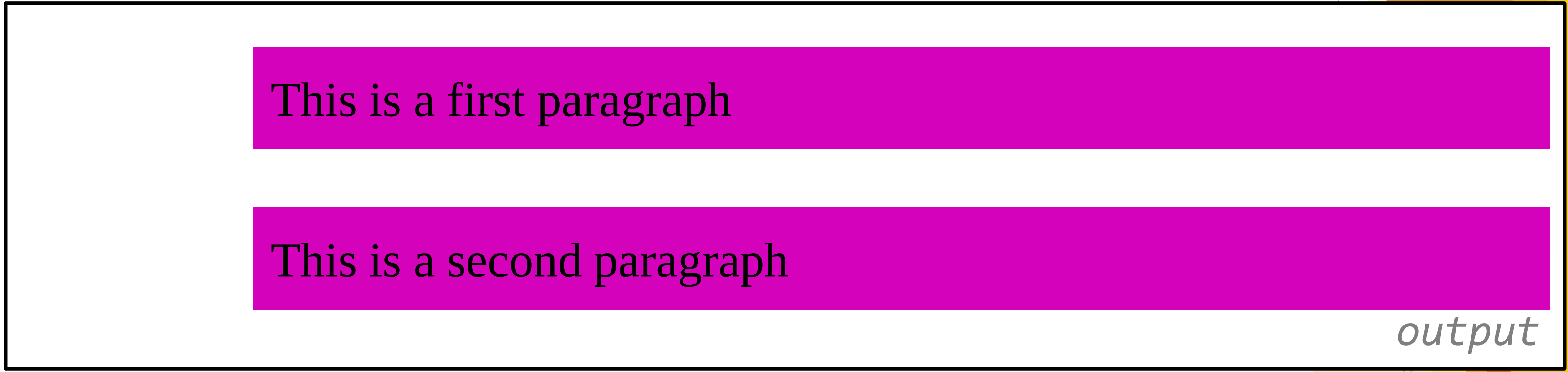
output

- □ notice that margins are always transparent

Margin example 2

```
p {
margin-left: 8em;
background-color: fuchsia;
}
```

CSS



This is a first paragraph

This is a second paragraph

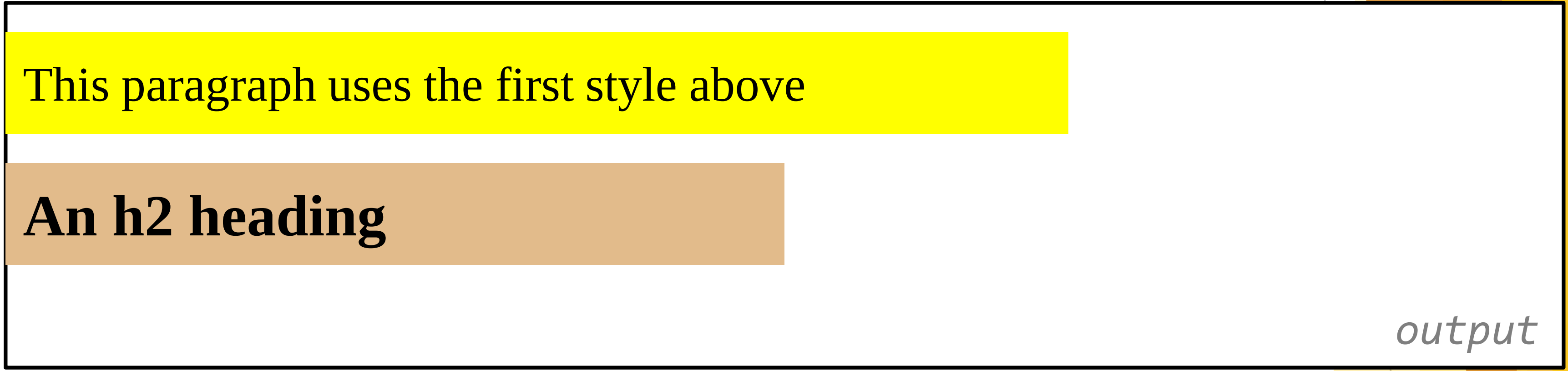
output

- each side's margin can be set individually

CSS properties for dimensions

```
p { width: 350px; background-color: yellow; }
h2 { width: 50%; background-color: aqua; }
```

CSS



This paragraph uses the first style above

An h2 heading

output

property	description
width, height	how wide or tall to make this element (block elements only)
max-width, max-height, min-width, min-height	max/min size of this element in given dimension

Centering a block element: auto margins

```
p {
margin-left: auto;
margin-right: auto;
width: 750px;
}
```

CSS

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

output

- works best if `width` is set (otherwise, may occupy entire width of page)
- to center inline elements within a block element, use `text-align: center;`