

# INFO 151

# Web Systems and Services

Week 3 (T1)

Dr Philip Moore

Dr Zhili Zhao

# Cascading Style Sheets

## 将英语转换为中文

# Review

- In this tutorial we will introduce:
  - HTML and cascading style sheets (CSS)
  - The motivation for CSS
  - The benefits of CSS
  - Methods of adding style using CSS and the order of precedence
  - Practical implementations of simple CSS in the NetBeans IDE
  - Writing and Running HTML with external CSS Files in the NetBeans IDE

# Introduction to Cascading Style Sheets

# HTML and Cascading Style Sheets

- HTML was never intended to contain tags for formatting a web page
- HTML was created only to describe the content of a web page:  
`<h1>This is a heading</h1>`  
`<p>This is a paragraph.</p>`
- HTML 3.2 added tags such as `<font>` and colour attributes (but)
  - There were problems in the design, development, and maintenance of large websites
  - Adding fonts and colour information to every single page became a long, expensive, and error prone process
- To solve this problem the World Wide Web Consortium (W3C) created Cascading Style Sheets (generally termed **CSS**)
- CSS changed the way style is used in HTML

# Motivation for CSS

- A growing problem for web site designers is the large variation in computing devices which range from computers to tablets and mobile phones
- Web-sites must be adaptable to the various devices while maintaining a consistent format (or 'style')
- The layout and formatting of the web-page must change to suit the different devices

# Why Use CSS?

- CSS provide the capability to set 'style rules'
  - 'Style rules' provide an effective basis to define a consistent 'style' across multiple web-pages and web-sites
- Web sites must accommodate a range of computing devices
  - Each device will have different screen resolution, profile, and size
  - Web-sites must be adaptable to the various devices while maintaining a consistent 'style'
  - The layout of the web-page must change to suit the different devices
  - CSS with layout managers provide a basis for adaptable web-sites

# The Benefits of CSS

- A CSS describes how HTML elements are formatted and displayed:
  - In a graphical user interface (screen)
  - On paper (printed in a hard copy) and in other media
- For example with an external CSS:
  - The look and formatting of an entire website(s) can be changed using one CSS file
- CSS files promote:
  - code re-use
  - Minimise formatting errors
  - Make web-site maintenance and updating simpler
  - Software maintenance generally represents 70% of the software life cycle



# Style and CSS

- An HTML web-page (or web-site) may use multiple approaches to introduce 'style' formats
- All 'styles' will "cascade" into a new "virtual" style sheet
- Based on the following 'style rules' (where number one has the highest priority):
  1. Inline style (inside an HTML element)
  2. External and internal style sheets (defined in the head section of the web page)
  3. Browser default
- An inline style (inside a specific HTML element) has the highest priority – it will override
  - A style defined inside the <head> tag
  - An external CSS style sheet
  - A browser default value

# CSS Priority (1)

- If properties have been defined for the same selector (element) in different style sheets the value from the **last read** style sheet will be used
- For example: assume that an external style sheet has the following style for the

```
<h1> element:  
  h1 {  
    color: navy;  
  }
```

- Then assume that an internal style sheet also has the following style for the

```
<h1> element:  
  h1 {  
    color: orange;  
  }
```

## CSS Priority (2)

- If the internal style is defined after the link to the external style sheet the `<h1>` elements will be "orange":

```
<head>  
<link rel="stylesheet" type="text/css" href="mystyle.css">  
<style>  
h1 {  
    color: orange;  
}  
</style>  
</head>
```

# CSS Priority (3)

- If the internal style is defined before the link to the external style sheet the <h1> elements will be "navy":

```
<head>
  <style>
    h1 {
      color: orange;
    }
  </style>
  <link rel="stylesheet" type="text/css" href="mystyle.css">
</head>
```

# CSS Priority (4)

- From the precedence applied to CSS style rules applied to HTML elements we can see that:
  - Careful attention must be applied to ensure that the **style logic (or priority)** is **understood** and **applied** to achieve the planned web-page format
- In practice it is good programming practice
  - To use a CSS for all web-site development
  - This maintains consistency in HTML element styles and format
  - Reduces the potential for errors
  - Makes updating and maintaining web-pages and web-sites easier

# CSS Syntax and Selectors

- A CSS rule-set consists of a **selector** and a **declaration block**:

```
<h1> {color:blue; font-size:12px;}
```

- The declaration block contains one or more declarations separated by semicolons
  - Each **declaration** includes a CSS property *name* and a *value*, separated by a *colon*
  - A CSS declaration always ends with a semicolon, and declaration blocks are surrounded by curly braces
- In the following example all **<p>** elements (in an HTML document) will be **'center-aligned'** with text formatted colour **'red'**:

```
p {  
    color: red;  
    text-align: center;  
}
```

# Example of the Syntax for CSS

```
<!DOCTYPE html>
<html>
  <head>
    <style>
      p {
        color: red;
        text-align:center;
      }
    </style>
  </head>
  <body>
    <p>Enter text here</p>
  </body>
</html>
```

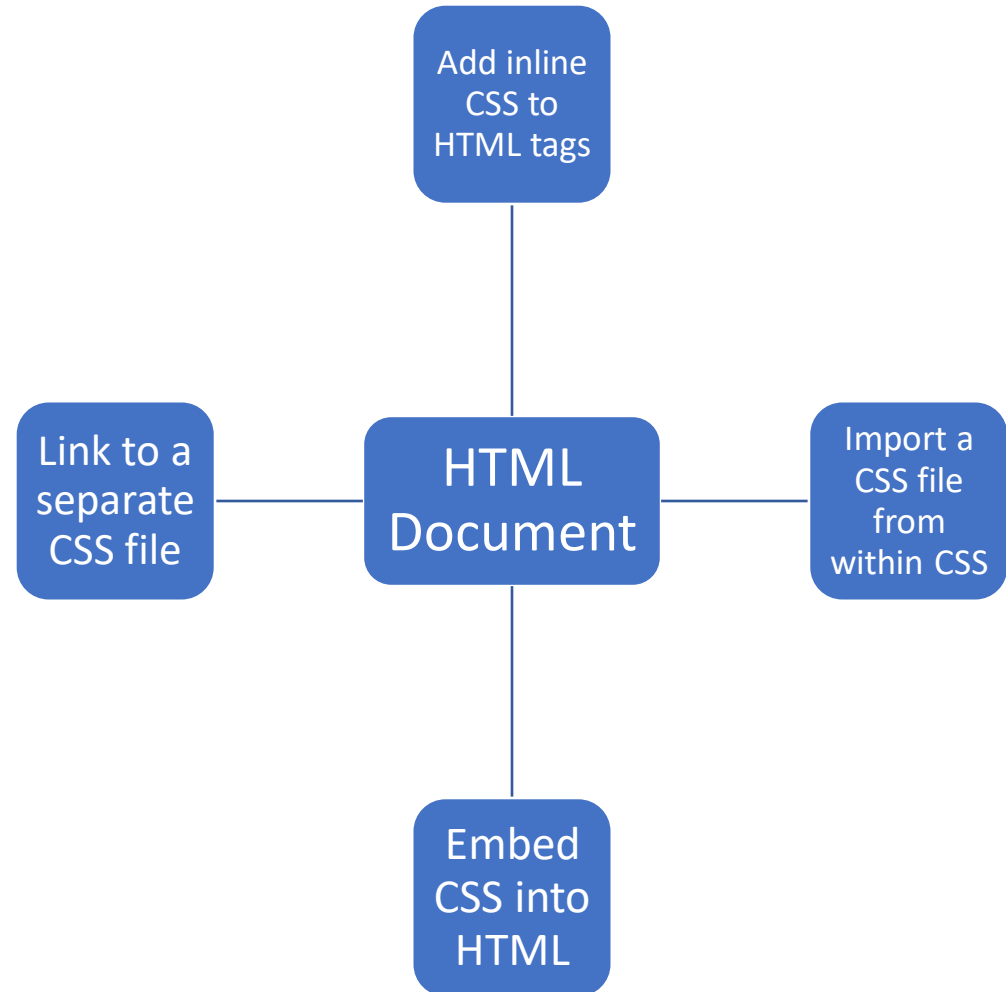
- Note:
  - The example shows the 'style' specified in the <head> of the HTML
  - The example also shows the correct way to write HTML
  - The indented style improves the readability of the HTML
- The indented code applies to all programming languages

# Methods for Adding Style



# Adding Style

- There are 4 methods of adding style to an HTML document
  1. Adding inline CSS to individual HTML tags
  2. Linking to a separate CSS file
  3. Embedding CSS into the HTML
  4. Import a CSS file from within CSS
- In this course we will focus on methods 1, 2, and 3.



# Adding Style

- In the following slides show examples of CSS files and their use
- We will demonstrate:
  - Inline CSS (inline)
  - Linking to a Separate CSS (external)
  - Embedding CSS into the HTML (internal)
- For examples showing how to use CSS see the course resources

# Adding Style Method One

# Inline CSS (method 1)

- **Style rules** can also be added directly to any HTML element
  - Simply add a **style parameter** to the **element** and enter your **style rules** as the value.
  - Here is an example of a heading with red text and a black background:

```
<h2 style="color:red;background:black;">
```

This is a red heading with a black background

```
</h2>
```

- This is not a very good method: it will cause 'bloat' in the HTML
- More importantly: inline CSS makes web-site updating and maintenance difficult and error prone

# Inline CSS (method 1)

- However inline CSS can be useful in some situations
  - An example may be where a system is used which has no access to the CSS file
  - In such a case add the 'style rules' directly to the elements
- Other potential uses for inline CSS include
  - An internal style sheet may be used if one single page has a unique style
  - Internal styles are defined within the **<style>** element, inside the **<head> ... </head>** section of an HTML page
  - An inline style may be used to apply a unique style for a single element
  - To use inline styles, add the style attribute to the relevant element – the style attribute can contain any CSS property
  - The example below shows how to change the colour of the **<h1>** element

# Using Inline CSS (method 1)

- This is HTML without 'style':

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1> This is a  
Heading</h1>
```

```
</body>
```

```
</html>
```

- This is HTML with 'style':

```
<!DOCTYPE html>
```

```
<html>
```

```
<body>
```

```
<h1 style="color:blue;">  
This is a Blue  
Heading</h1>
```

```
</body>
```

```
</html>
```

# This is a Heading

Output without added style  
The Default colour is black



## This is a Blue Heading

Output with added style  
The text has added colour  
This example uses inline style



# Adding Style Method Two

# Linking to a Separate (external) CSS (method 2)

- With this method all 'style' rules are contained in a single text file that is saved with the \*.CSS extension (where \* is the file name)
- This file is saved on the server and is linked to each HTML file
- The link is a line of HTML in the <head> ... </head> section of an HTML document as follows:

```
<link rel="stylesheet" type="text/css"  
href="mystyles.css" media="screen" />
```

# Linking to a Separate (external) CSS (method 2)

- This is the recommended method as there are many advantages to linking to a separate CSS file including:
  - Web-site updating and maintenance is simpler and more reliable using a single CSS file
  - A complete re-design of an entire web-site (the look and style) again requires only changing a single CSS file
- Each page must include a reference to the external style sheet file inside the `<link>` element. The `<link>` element goes inside the `<head> ... </head>` section:
  - An external style sheet can be written in any text editor
  - The file should not contain any html tags
  - The style sheet file must be saved with a `*.css` extension

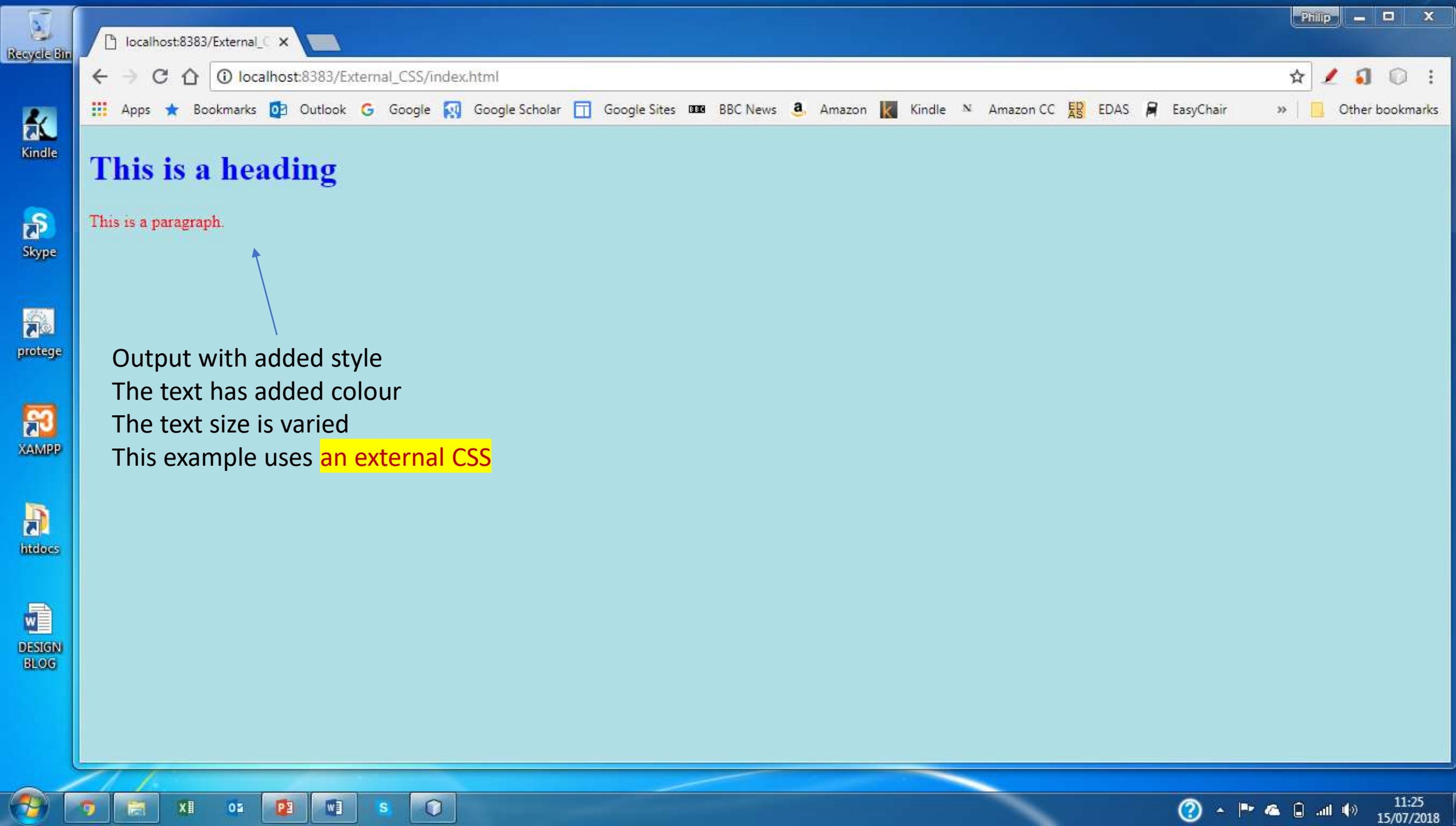
# Linking to a Separate (external) CSS (method 2)

- The index.html file

```
<!DOCTYPE html>
<html>
<head>
<link rel="stylesheet"
href="styles.css">
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

- The "styles.css" file

```
body {
    background color: powderblue;
}
h1 {
    color: blue;
}
p {
    color: red;
}
```



# This is a heading

This is a paragraph.

Output with added style

The text has added colour

The text size is varied

This example uses an external CSS

# Adding Style Method Three

# Embedded CSS (method 3)

- CSS 'style rules' may be embedded directly into any HTML web-page
- To embed CSS rules into an HTML document add the following code to the **<head>** of an HTML document:

```
<style media="screen" type="text/css">  
    Add style rules here  
</style>
```

- All CSS rules are placed between the **<style>** ... **</style>** tags.
- The media can be "screen" for your computer screen or "print" for printing
- The disadvantage with embedding is the styles must be downloaded every time someone visits the page
- However there are a advantages
  - Because the CSS is part of the HTML document the **whole web-page exists as just one file**
  - **This can be useful if the web-page is a template**

# Embedding CSS into the HTML

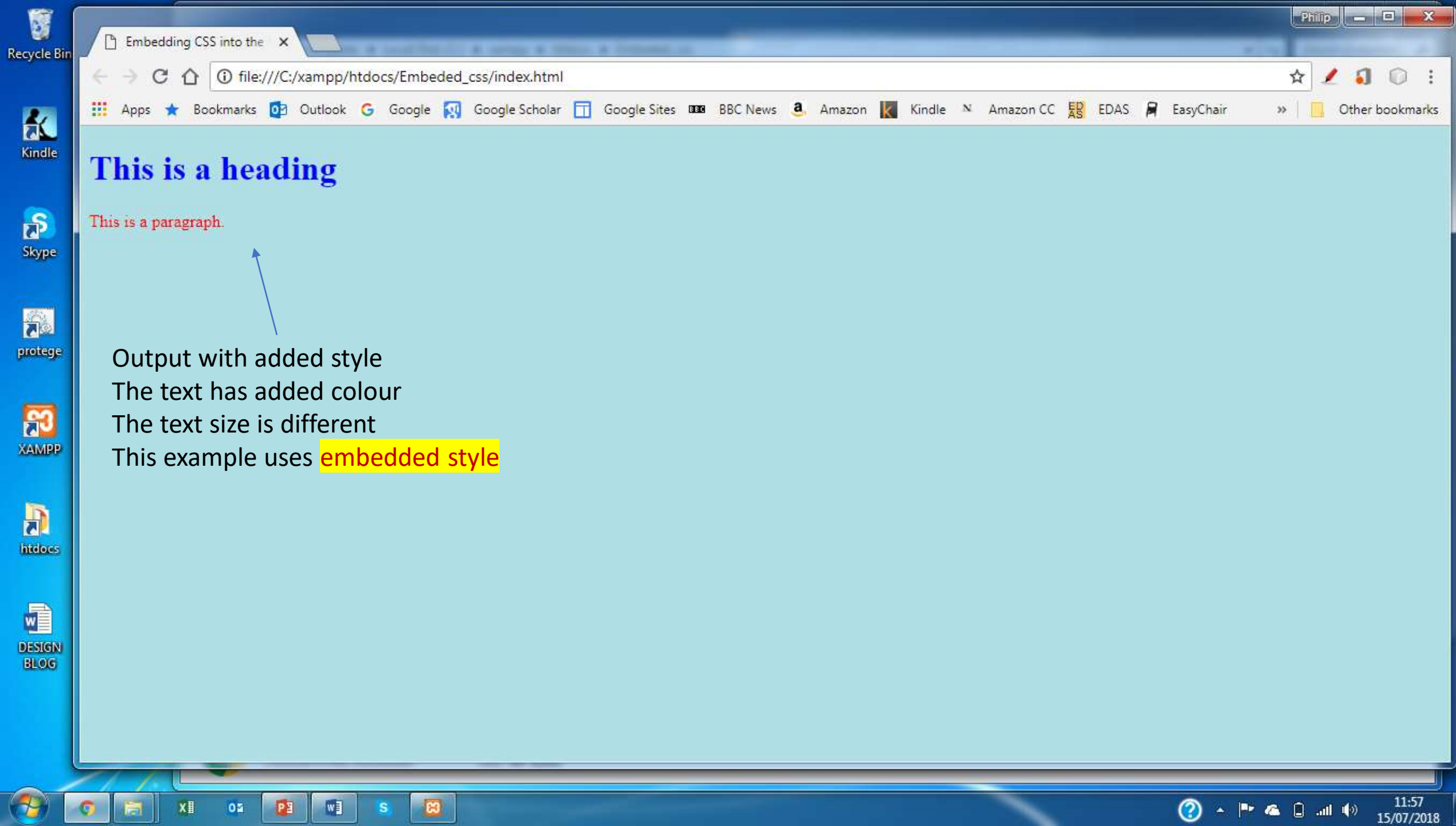
- The document template:

```
<!DOCTYPE html>
<html>
<head>
<!-- enter the CSS here -->
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```

- The following example shows the HTML

```
<!DOCTYPE html>
<html>
<head>
<style>
body {background-
color: powderblue;}
h1    {color: blue;}
p     {color: red;}
</style>
</head>
<body>
<h1>This is a heading</h1>
<p>This is a paragraph.</p>
</body>
</html>
```





This is a heading

This is a paragraph.

Output with added style

The text has added colour

The text size is different

This example uses embedded style

# Adding Style Method Four

# Import CSS file from within CSS (method 4)

- An interesting method of adding CSS rules to a HTML page is to use the import rule.
- The import rule enables the attachment of a new CSS file from within CSS itself.
- Consider a practical example of when you might use this method.
  - To import a new CSS file from within CSS simply use the following rule:  
**`@import "newstyles.css"`**
- Remember the path is relative to the current CSS file that we are using
- Where 'style' is embedded into the HTML page the path is relative to the HTML file

# Writing and Running HTML with External CSS Files

# Writing HTML and CSS

- Preparing HTML and style sheet files
  - Use filename.html (the convention is index.html)
  - Use filename.css
- The files may be written using a text editor
  - In a Windows system 'Notepad'
  - Do not use Microsoft Word
- However
  - The preferred method is to use an IDE
  - In this course we will use NetBeans
  - In NetBeans a .css template file can be created using the file menu
  - The following slides show the HTML and CSS files with the output



External\_CSS - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Files Services

External\_CSS

- Site Root
  - index.html
  - styles.css
- Unit Tests

Navigator

- CSS
  - At-Rules
    - @import "styles.css";
  - Imports
    - styles.css
- HTML
  - html
    - head
      - link
      - title
    - body
      - h1
      - p

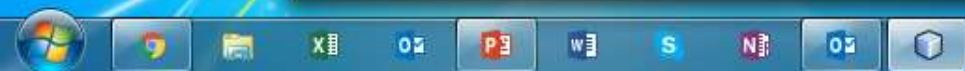
styles.css index.html

Source History

```
1 <!DOCTYPE html>
2 <html>
3 <head>
4   <link rel="stylesheet" href="styles.css">
5   <title>External Cascading Style Sheet Example</title>
6 </head>
7 <body>
8   <h1>This is a heading</h1>
9   <p>This is a paragraph.</p>
10 </body>
11 </html>
12
```

Output

12:1 INS





External\_CSS - NetBeans IDE 8.2

File Edit View Navigate Source Refactor Run Debug Profile Team Tools Window Help

Search (Ctrl+I)

Projects Files Services

- External\_CSS
  - Site Root
    - index.html
    - styles.css
  - Unit Tests

Navigator

- Elements
  - body
  - h1
  - p
- Rules
  - body
  - h1
  - p

styles.css index.html

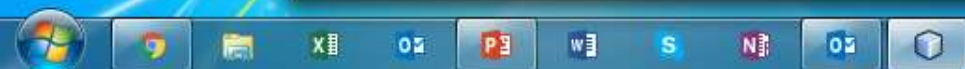
Source History

```
1 body {  
2     background-color: powderblue;  
3 }  
4 h1 {  
5     color: blue;  
6 }  
7 p {  
8     color: red;  
9 }
```

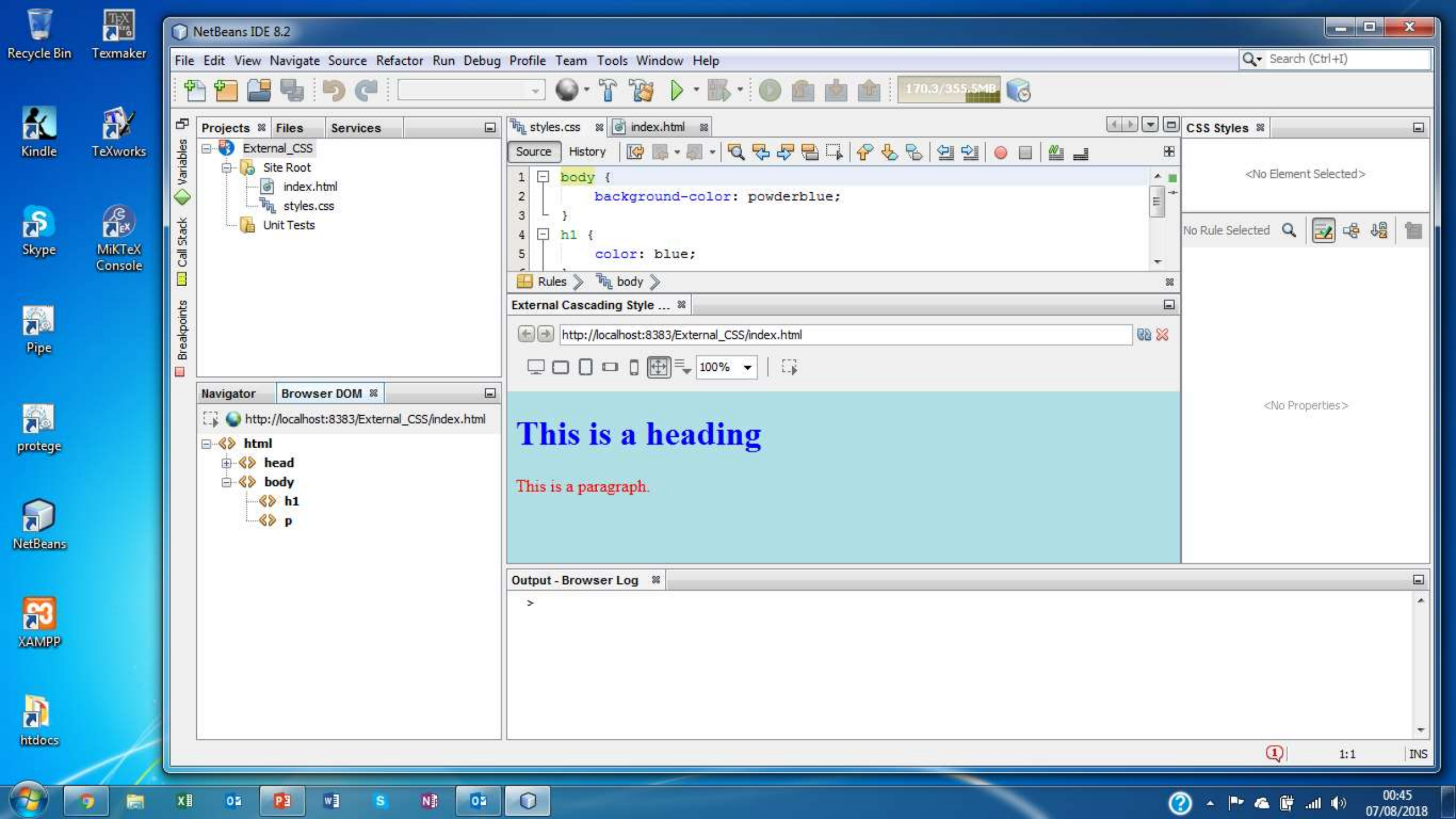
Rules > body >

Output

1:1 INS











Philip

Index of C:\xampp\htdocs - X

file:///C:/xampp/htdocs/

Apps Bookmarks Outlook Google Google Scholar Google Sites BBC News Amazon Kindle Amazon CC EDAS EasyChair Other bookmarks

# Index of C:\xampp\htdocs\

[parent directory]

Name	Size	Date Modified
dashboard/		31/07/2018, 13:59:03
img/		31/07/2018, 13:59:03
Php_HTML/		31/07/2018, 19:42:17
Php_HTML_1/		31/07/2018, 19:44:38
Php_test_1/		31/07/2018, 18:39:18
Php_Test_2/		31/07/2018, 18:41:38
Php_Test_3/		31/07/2018, 18:47:15
Php_Test_4/		31/07/2018, 18:52:05
Php_Test_5/		31/07/2018, 18:54:01
Php_Test_6/		31/07/2018, 19:01:59
webalizer/		31/07/2018, 13:58:58
xampp/		31/07/2018, 13:59:03
applications.html	3.7 kB	06/07/2018, 10:30:40
bitnami.css	177 B	27/02/2017, 09:36:00
favicon.ico	30.2 kB	16/07/2015, 16:32:32
index.php	260 B	16/07/2015, 16:32:32

# NetBeans IDE

- From the previous slides we can see:
  - Written in the IDE editor we can see:
    - The `index.html` HTML code
    - The `styles.css` code:
  - The navigator showing
    - The `index.html` file import and HTML elements
    - The `styles.css` style rules and HTML elements
- The previous slides show
  - The output (using the NetBeans embedded web kit browser)
  - The projects file structure in the '`htdocs`' folder (the `Site Root`)

# NetBeans IDE

- From the previous slides we can see:
  - Written in the IDE editor we can see
    - The index.html HTML code
    - The styles.css code
  - The projects file structure
  - The navigator showing
    - The index.html file import and HTML elements
    - The styles.css style rules and HTML elements
- The previous slide shows
  - The output (using the NetBeans embedded browser)
  - The navigator showing the 'Browser DOM', url, and elements

# Running the External CSS Method (1)

- We have seen the creation of
  - The “*index.html*” file
  - The simple “*styles.css*” file
- To run these files
  - The “*index.html*” file will be found by the browser and run in the normal way
  - The “*styles.css*” file will be stored in a known location where the path is known to the *index.html* file

# Running the External CSS Method (2)

- It is the usual (and best) practice
  - To store the “styles.css” file in the same directory (file) as the “index.html” file
  - For example: the “htdocs” file in the XAMPP web-server)
- The path (the location of the file) must be in the form (using a Windows computer):
  - C:\xampp\htdocs\css\_test\styles.css
- If the “styles.css” file is stored in a different location the full path must be specified
- Running a web-page requires the url to the index.html file which is:
  - [http://localhost:8383/External\\_CSS/index.html](http://localhost:8383/External_CSS/index.html)

# bringing it all together

- Knowing how to create HTML files and run web-pages / web-sites does not address the need for good web-site design
- Bringing it all together is the process of design where
  - The requirements are investigated
  - The appropriate features are selected
  - The features selected are combined into a web-page / web-site that is
    - Attractive
    - Easy to use
    - Effective in presenting the information
- It is the task of the designer to
  - Meet the user requirement
  - Provide a multi-page web-site with clear navigation that users want to visit

# Review

- In this session I have introduced:
  - Cascading Style Sheets (CSS)
  - Multiple CSS
  - Introduced the order of precedence for CSS
  - The methods of introducing style
  - Practical implementations of simple CSS
- CSS provide greater formatting and styling capability in HTML5 over HTML4.
- In HTML5 Cascading Style Sheets are the preferred method of formatting web pages and web sites