

# WEB DEVELOPMENT

---

## Lecture 03 Introductory XHTML 2

# Basic XHTML Elements

---

- Document Structure
- Text
- Links
- **Lists**
- Generic Elements

# List elements

---

There are three types of list in HTML:

- Unordered lists
- Ordered lists
- Terms and Definitions

They are usually displayed as a bulleted indented list by default.

By using CSS you can change the appearance of the list in many ways – even making it display horizontally instead of vertically.

# Unordered Lists and Ordered Lists

---

```
<ul>
```

```
  <li>2001 a space odyssey</li>
```

```
  <li>Blade Runner</li>
```

```
  <li>Alien</li>
```

```
</ul>
```

```
<ol start="4">
```

```
  <li>X Files</li>
```

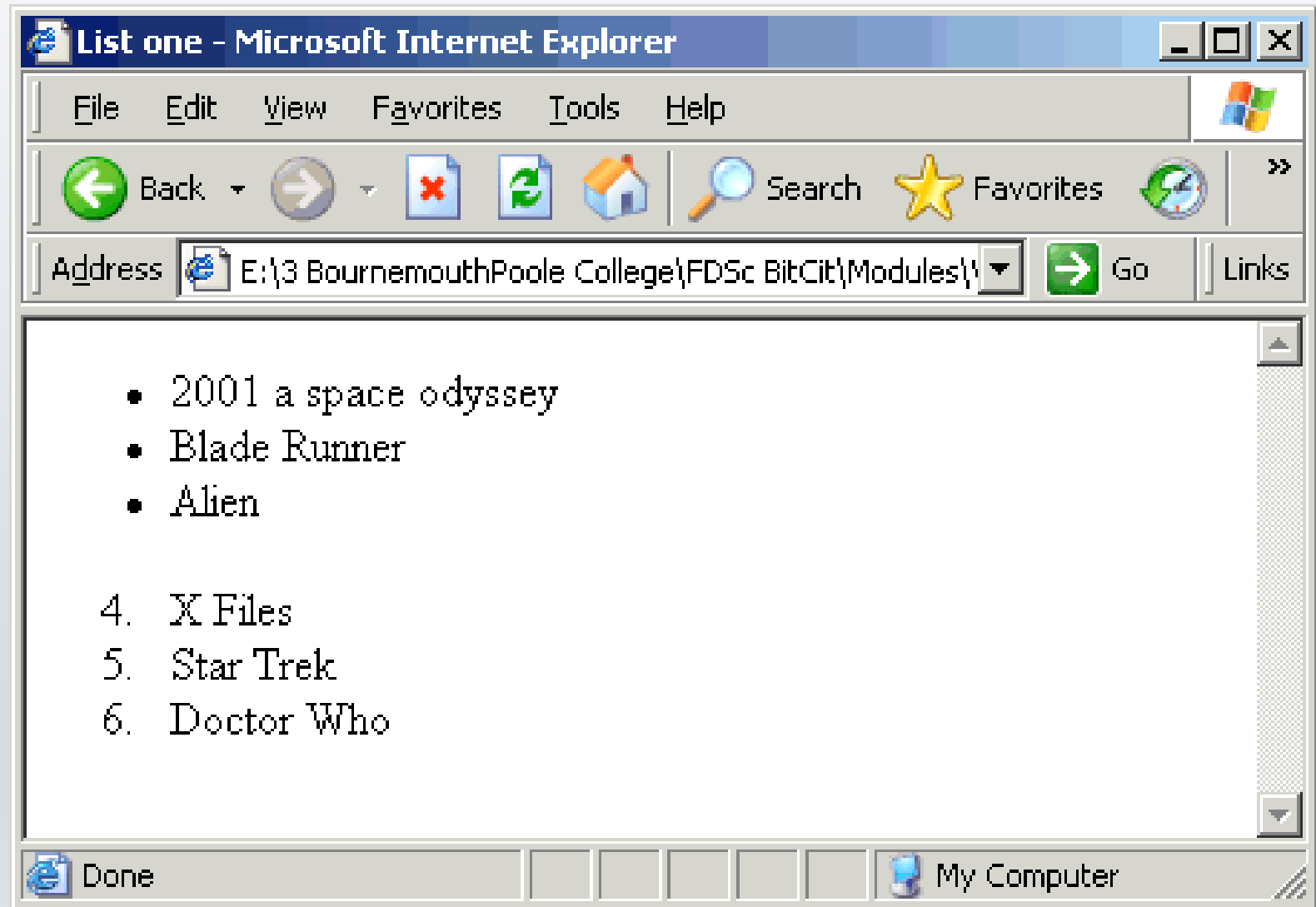
```
  <li>Star Trek</li>
```

```
  <li>Doctor Who</li>
```

```
</ol>
```

# Default Style - Internet Explorer

---



# Definition Lists

---

Displays a list of terms along with their definitions.

```
<dl>
```

```
  <dt>Porcupine Tree</dt>
```

```
  <dd>A progressive group dating back to the early  
1990s</dd>
```

```
  <dt>Can</dt>
```

```
  <dd>A German avant-garde rock group from the  
1970s</dd>
```

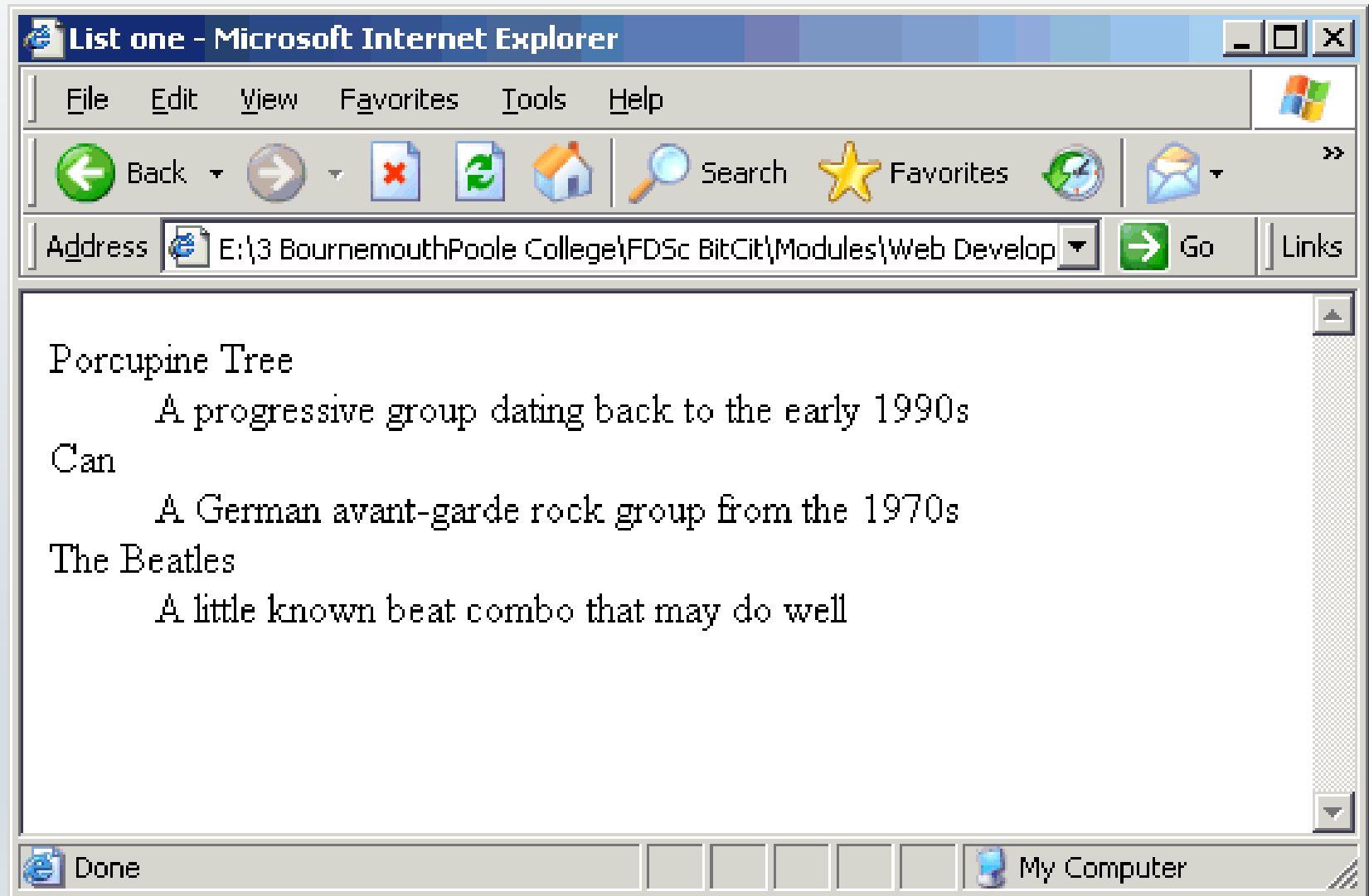
```
  <dt>The Beatles</dt>
```

```
  <dd>A little known beat combo that may do  
well</dd>
```

```
</dl>
```

# Default Style – Internet Explorer

---



# CSS properties for Lists

---

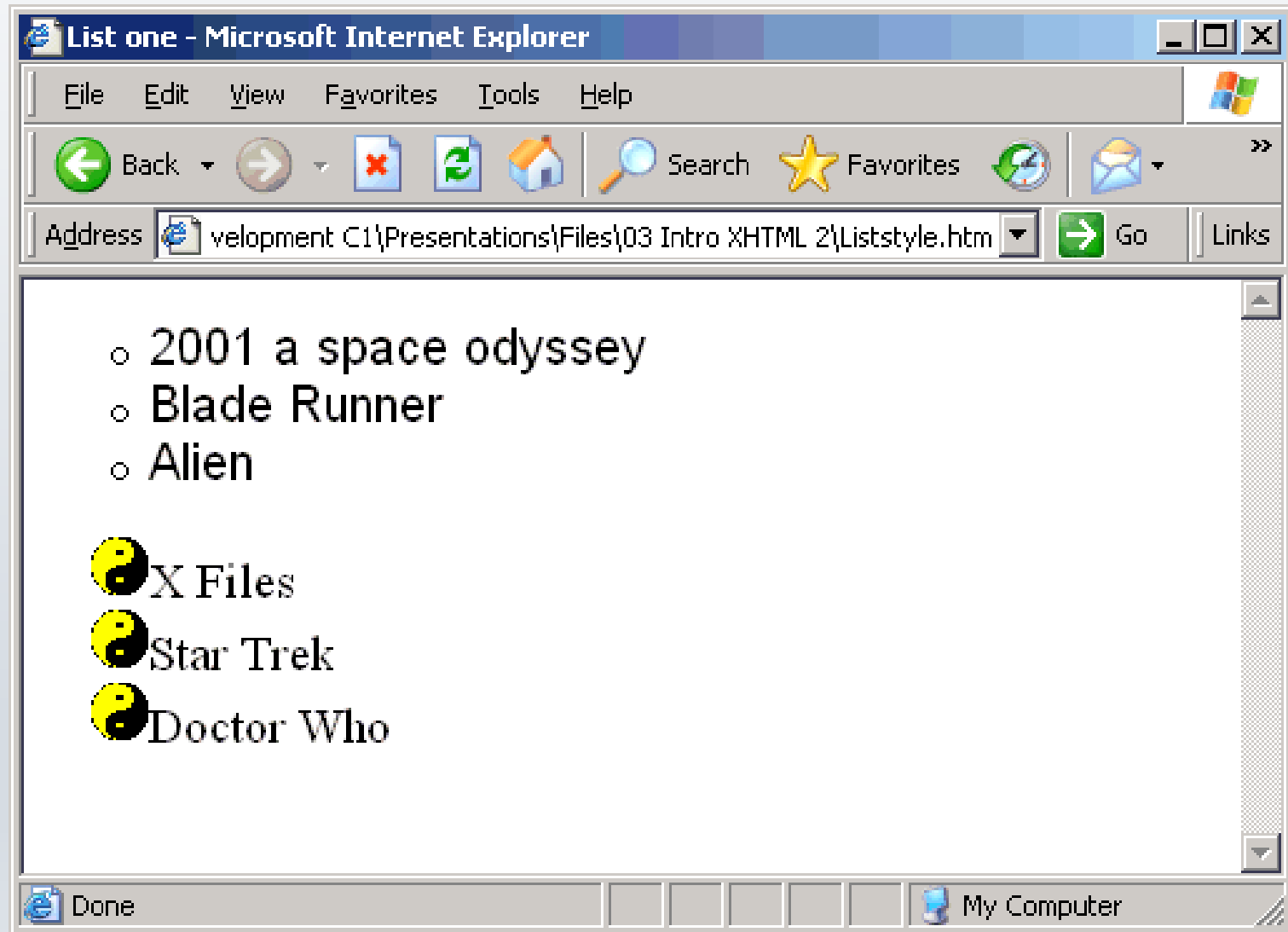
```
ul
{
list-style-type: circle;
font-family: Arial;
font-size: 20px;
}
```

```
ol
{
list-style-image: url(yinyang.gif) ;
font-family: serif;
}
```



# CSS styles for Lists

---



# list-style-type: valid keywords

---

disc



circle



square



decimal

1, 2, 3, 4, 5 ...

decimal-leading-zero

01, 02, 03, 04 ...

lower-alpha

a, b, c, d, e ...

upper-alpha

A, B, C, D, E ...

lower-roman

i, ii, iii, iv, v ...

upper-roman

I, II, III, IV, V ...

lower-greek

$\alpha$ ,  $\beta$ ,  $\chi$ ,  $\delta$ ,  $\varepsilon$

# Problem

---

What happens if we have two sections of text, both enclosed in the same tags – say `<p>`.

But we want to give them different styles?

What we do is use generic elements.

They enable you to create custom HTML elements to suit your own requirements.

# Generic Elements

---

- `<span>` An inline element – they can be placed into the middle of text. They tend to be used to modify individual words or sentences.
- `<div>` A block-level element like `<p>` – contents will start and end on a new line. They can be used as containers for sections of HTML.

They are both used in conjunction with element identifier attributes:

- `class` Used for grouping similar elements. Multiple elements on the same page may be given the same class name.
- `id` Used for giving elements a unique name. There can only be one element on a page with a given id name

# Example of <span>

---

## HTML code

```
<p>Work number: <span class="telephone">0122 096483</span> but  
don't use it after 5:00</p>
```

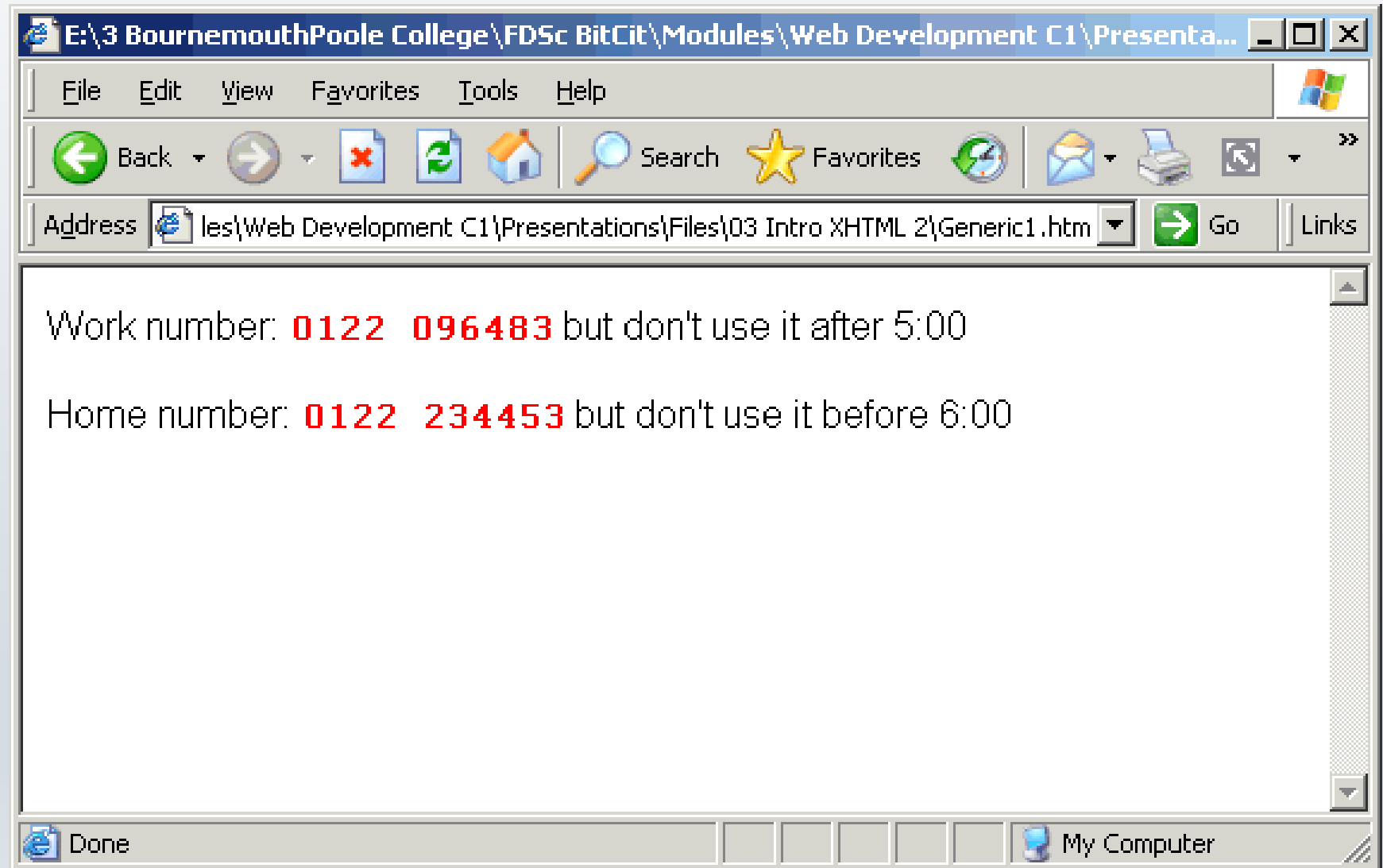
```
<p>Home number: <span class="telephone">0122 234453</span> but  
don't use it before 6:00</p>
```

## Stylesheet – note use of dot as the class selector

```
p  
{  
font-family: arial;  
}
```

```
span.telephone  
{  
font-family: courier;  
font-weight: bold;  
color: red;  
}
```

# Example of <span>



# Example of <div>

---

## HTML code

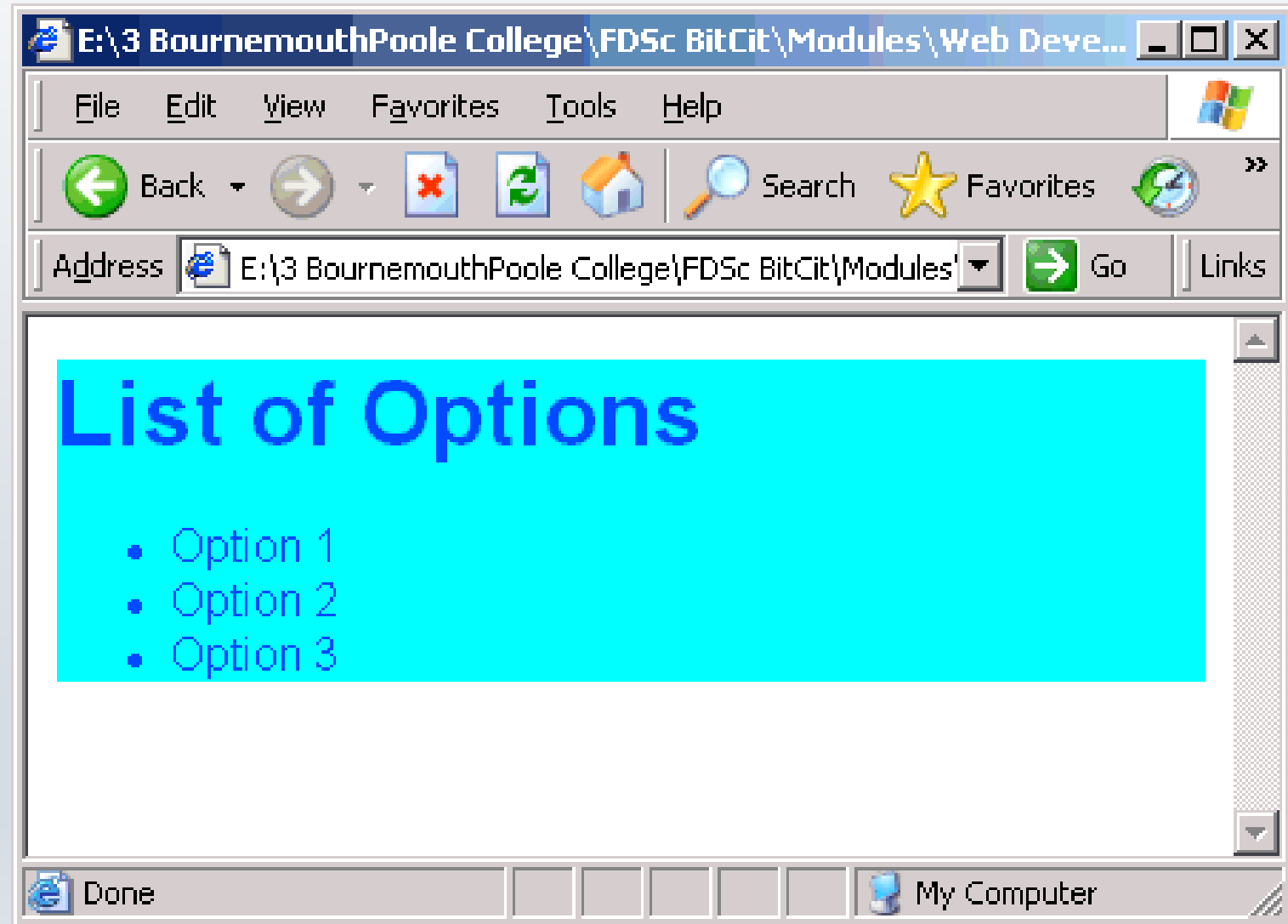
```
<div id="topmenu">
  <h1>List of Options</h1>
  <ul>
    <li>Option 1</li>
    <li>Option 2</li>
    <li>Option 3</li>
  </ul>
</div>
```

## Stylesheet – note use of hash as id selector

```
div#topmenu
{
font-family: Arial;
background-color: aqua;
color: blue;
}
```

# Example of <div>

---





# Notes on using class and id selectors

---

Remember that `class` can be also be used with `<div>` and `id` can also be used with `<span>`.

They can both be used with almost all other HTML elements.

Keep class and id names meaningful. They should describe the content, not the appearance.

As well as acting as stylesheet selectors, they can also be used as:

- targets for links
- a way for a script to refer to a particular element
- data which can be accessed by search engines