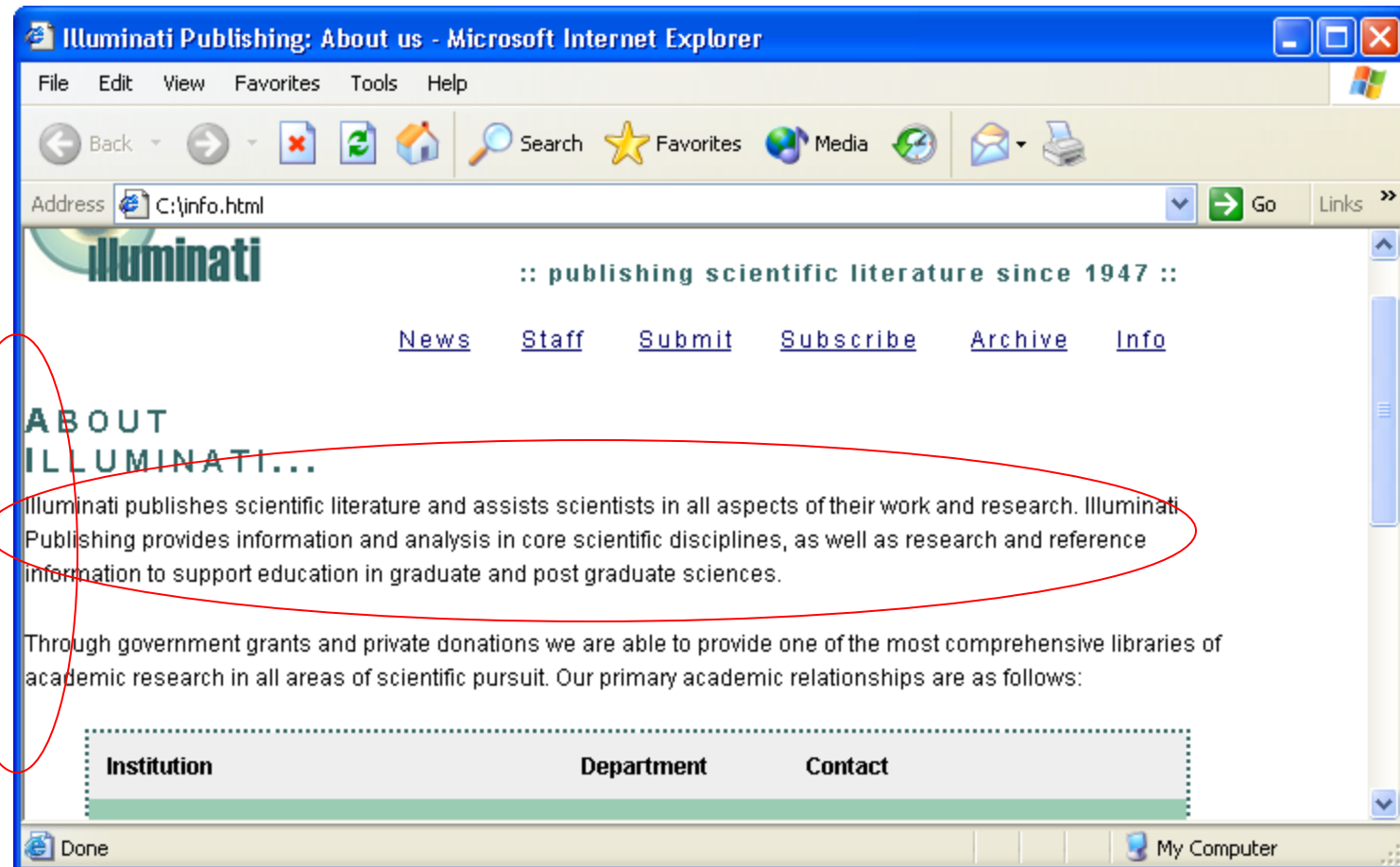


Designing Content Sections (p.99)

- ❑ 控制邊緣和填充 • Control Margins and Padding
- ❑ 控制邊框 • Create Borders
- ❑ 控制元素尺寸 • Control Element Dimensions
- ❑ 建立浮動元素 • Create Floating Elements
- ❑ 控制內文溢位 • Control Content Overflow

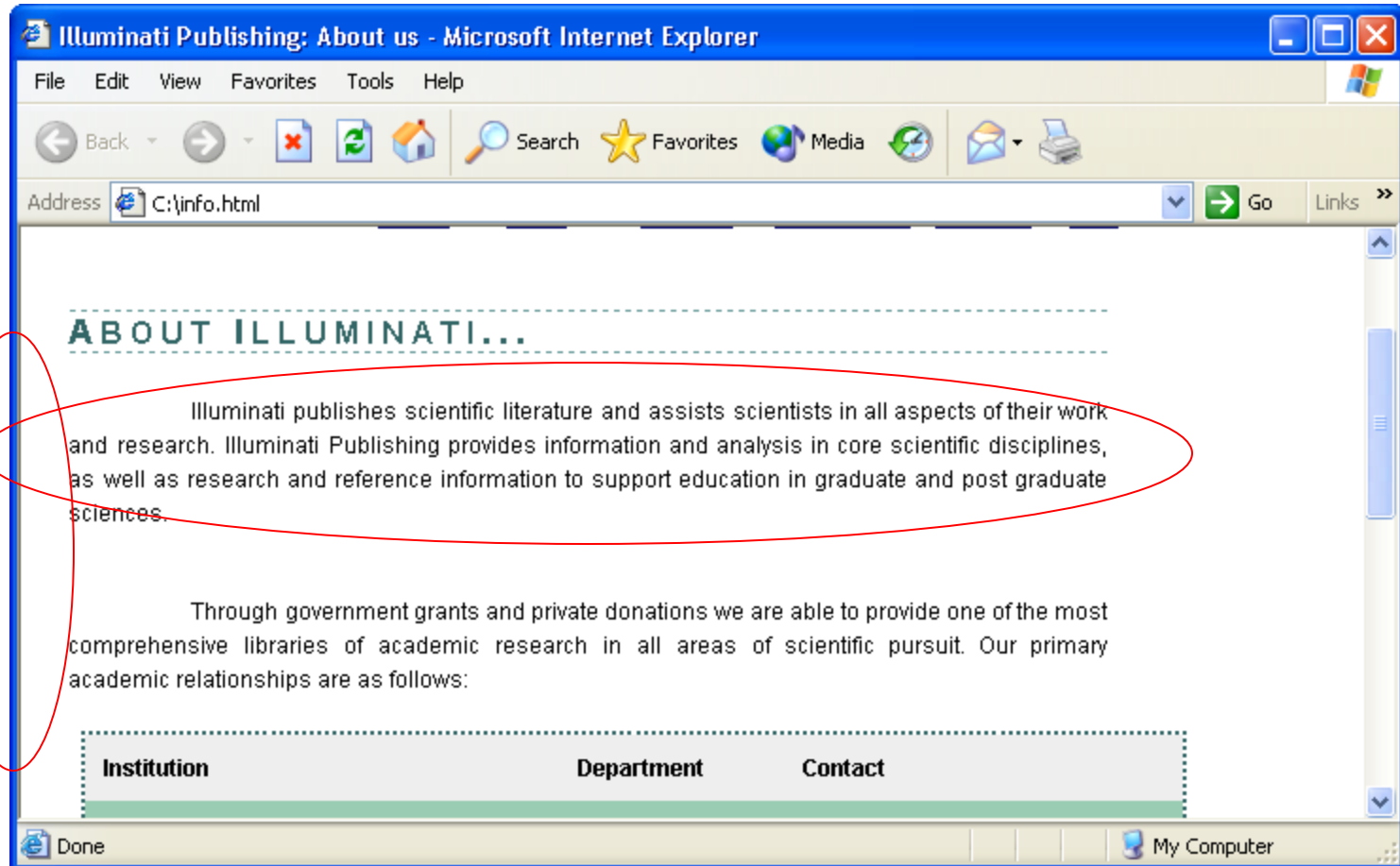
當元素所佔的空間和尺寸不是我們希望的樣式, 可以透過**CSS**來修改

Control Margins and Padding (p.100)



Web page with default spacing

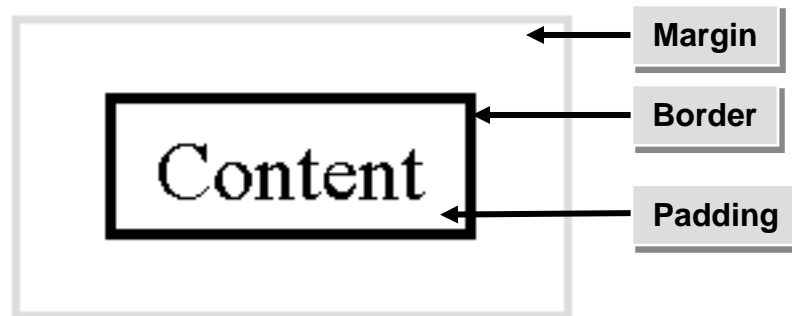
Control Margins and Padding (p.101)



Web page with increased margin spacing

The Box Model (p.101)

- 盒子模型是**CSS**的格式模型(formatting model), 它為元素建立長方形的盒子
- 盒子模型有三個元件：
 - 邊緣(margin)
 - 邊框(border)
 - 填充(padding)



Margin (p.102)

- **margin**是盒子模型定義元素盒子和父元素之間的空間
- 你可以使用量測單位, 如 **px**, **%**, **in**, **cm**, 或 **em**, ...
例 : `element { margin: 20px; }`
- 若使用絕對量測單位如**px**或**in**, 則**margin**會被固定, 不管瀏覽器的大小
- 若使用相對量測單位如**%**或**em**, 則**margin**會隨瀏覽器的大小縮放. 這就是所謂的 流動邊緣(fluid margin)
- 頁面邊緣 • Page Margins
 - 頁面的畫布是由 `<body>` 元素所定義
 - 要設定頁面邊緣, 只要用 `body` 選擇器做設定
 - 例 : 將頁面設定邊緣為 **20px** (請參考教材 p.102 圖示)
 - 由於不同瀏覽器預設的頁面邊緣不同(請參考教材 p.103 表格), 可以使用 **margin**覆寫瀏覽器的預設值
- 元素邊緣 • Element Margins
 - 如同頁面邊緣, 建議先覆寫元素邊緣的預設值
 - 範例 : (請參考教材 p.103 圖示)

- margin有5個屬性
 - margin-top
 - margin-right
 - margin-bottom
 - margin-left
 - margin : 簡寫, 可一次設定4個邊緣
- 設定邊緣 • Setting the Margin Property
 - 使用margin簡寫, 可以順序設定 top, right, bottom, left
 - 例
 - `p { margin: 10px; } /* 4邊都是10px */`
 - `p { margin: 5px 8px; } /* 上下是5px, 左右是8px */`
 - `div { margin: 2px 5px 10px; } /* 上2px, 左右5px, 下10px */`
 - `p { maring: 10px 20px 15px 5px; } /*上10px, 右20px, 下15px, 左5px */`
- 邊緣合併 • Collapsing Margins
 - 當CSS的邊緣在垂直方向上相鄰, 會發生合併(請參考教材p.105圖示)
 - 水平方向不會發生合併

- padding是盒子模型中定義元素的內容和盒子間的空間

This is a paragraph with no padding. The content of the paragraph is displayed flush against its background color.

← No Padding

This is more like it. This paragraph has 12 pixels of padding on all four sides of its box. The extra space between the edge of the background color and the content tends to make the text more readable.

← Padding Applied

填充的屬性

Padding Properties (p.105)

- padding有5個屬性
 - padding-top
 - padding-right
 - padding-bottom
 - padding-left
 - padding : 簡寫, 可一次設定4邊的增充
- padding簡寫屬性 • The padding Shorthand Property
 - 使用padding簡寫, 可以順序設定 top, right, bottom, left
- 填充表格單元 • Padding Table Cells
 - 填充表格單元通常用HTML cellpadding 屬性, 但cellpadding 只能讓表格單元四邊的值都相同
 - 使用CSS, 只要將padding套用至<td>元素, 例 : `td { padding: 2px 8px; }`
(請參考教材 p.106 表格)
- 負值margins • Negative Margins
 - 若希望將元素離開它的容器邊緣, 可以使用負值margin
 - `body { margin : 20px; }`
 - `h1 { margin-left: -20px; }`

Activity 3-1 Setting Margins (p.107)

- 你發現預設的 **page margin**, 在不同的瀏覽器有些微不同. 你和專案team決定要移除頁面邊緣以建立跨瀏覽器統一的顯示, 你需要變更**style.css**並符合以下需求
 - 調整文字內容左右邊緣和 **page canvas** 的間距
 - 增加段落到 **page canvas** 右邊之間的空白, 讓文字的長度縮短, 方便閱讀
 - 調整 **copyright** 和 瀏覽器底部的距離
 - 確認在不同的瀏覽器都有一致的 **margin length**.
- 1. 使用**margin**簡寫移除頁面全部的**margin**
- 2. 為什麼將頁面**margin**設為0有助於建立精準的元素**margin** ?
- 3. 對所有的段落用絕對量測單位調整左側邊緣(**left margin**)
- 4. 套用相同的左側邊緣(**left margin**)到 **h1**, **h2**, 和 **address** 元素
- 5. 確認不同的瀏覽器大小不會變更邊緣長度
- 6. 增加段落右側邊緣要超過左側
- 7. 在版權聲明之後增加底部邊緣
- 8. **margin-bottom** 對原本在<**body**>的**margin**簡寫裏設定 **margin:0px** 有什麼影響?
- 9. **margin:0px** 和 **margin-left:25px** 如果順序對調, 會有什麼結果?

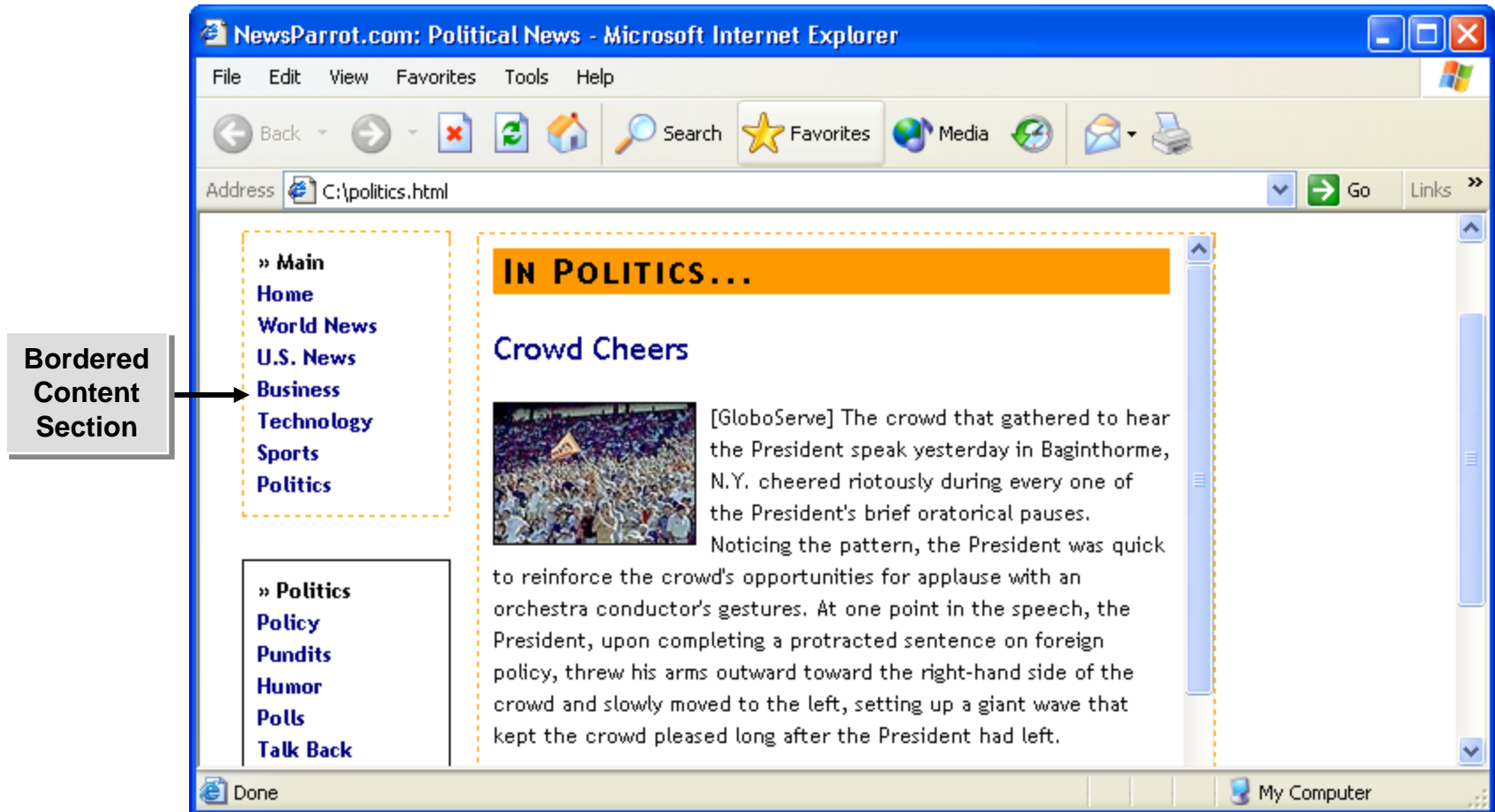
Activity 3-2 Creating Negative Margins (p.111)

- 在style.css中，page canvas 和 content paragraphs 的 margin 已經都設定好了。專案 team 希望縮小每一個標題底部的空白
 1. 減少標題底部的白色空白
 2. 為什麼需要負值邊緣

Activity 3-3 Modifying Element Padding (p.112)

- 專案 team 希望所有有背景色的 paragraphs 能增加 content 和 box 之間的空白。這樣會提高這些 elements 裏文字的可讀性，增加設計感
- 使用 style.css 進行專案 team 的要求。
 1. 給定 “note” 段落的四邊有足夠的padding來增加可讀性
 2. 建立規則在infomation表格的<th>和<td>增加足夠padding, 提供文字和表格單元間合適的分隔

- 你可以在任何元素上建立框線，並且控制它的樣式，寬度，和顏色



- CSS 框線(borders) 是顯示在元素四周, 不同的樣式, 顏色及寬度的線段
- Border的屬性 • The Border Properties
 - border-top-width
 - border-right-width
 - border-bottom-width
 - border-left-width
 - border-width
 - border-top-style
 - border-right-style
 - border-bottom-style
 - border-left-style
 - border-style
 - border-top-color
 - border-right-color
 - border-bottom-color
 - border-left-color
 - border-color

- 範例：

```
p.first {  
  border-width: 2px;  
  border-style: dashed;  
  border-color: black;  
  background-color: #eee;  
  padding: 8px; }
```

```
p.second {  
  border-top-width: 4px;  
  border-top-style : double;  
  border-top-color: black;  
  background-color: #eee;  
  padding: 8px; }
```

(效果請參考教材 p.116圖示)

- 框線簡寫屬性 • Border Shorthand Properties

- border-top
- border-right
- border-bottom
- border-left
- border
- 以下兩個範例意義相同

```
div { border: 2px ridge #eee; }
```

```
div {  
    border-top: 2px ridge #eee;  
    border-right: 2px ridge #eee;  
    border-bottom: 2px ridge #eee;  
    border-left: 2px ridge #eee; }
```

- 框線樣式 • Border Styles (請參考教材 p.117 列表)

- 邊框寬度 • Border Widths
 - 可使用任何長度的單位設定邊框的寬度
 - px 是最常使用的單位
 - 有3個內建的關鍵字可用於邊框的寬度
 - thin (= 2px)
 - medium (= 4px)
 - thick (= 6px)
- 邊框和行內元素 • Borders and Inline Elements
 - 邊框可套用於 block-level elements 和 inline elements
 - 例：

```
span { border: 2px dotted gray; }
```

(效果請參考教材 p.118 圖示)

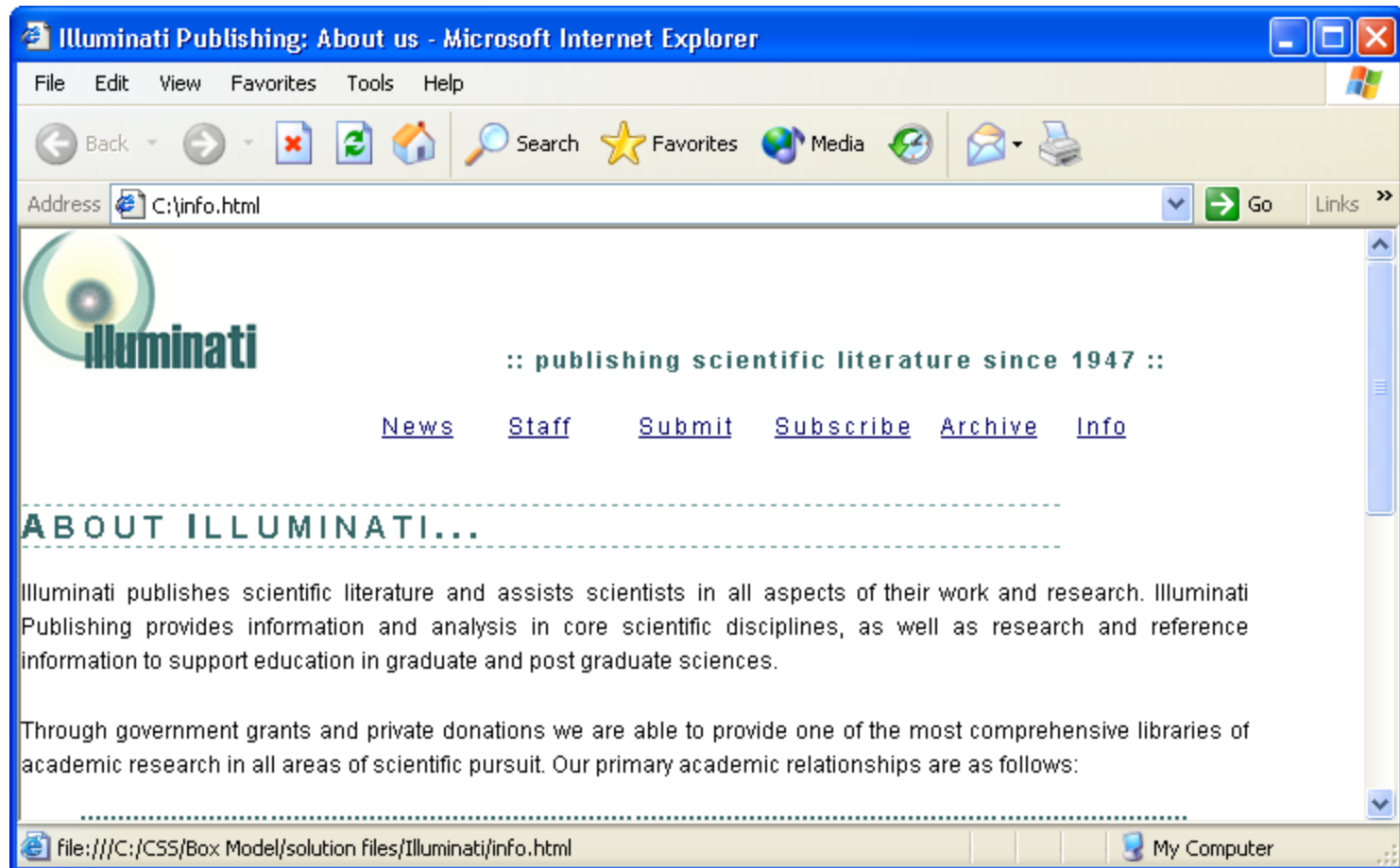
Activity 3-4 Creating Borders (p.119)

- 你和專案 team 決定使用 borders 加強網站的呈現以及突顯重要的內容
- 透過 style.css，建立borders來有效區隔 第一層標題，內部的content tables，以及“note”類別的段落
 1. 在第一層標題的上下兩邊建立框線
 2. 在 information表格建立框線
 3. 在“note”類別的段落建立框線

Control Element Dimensions (p.121)

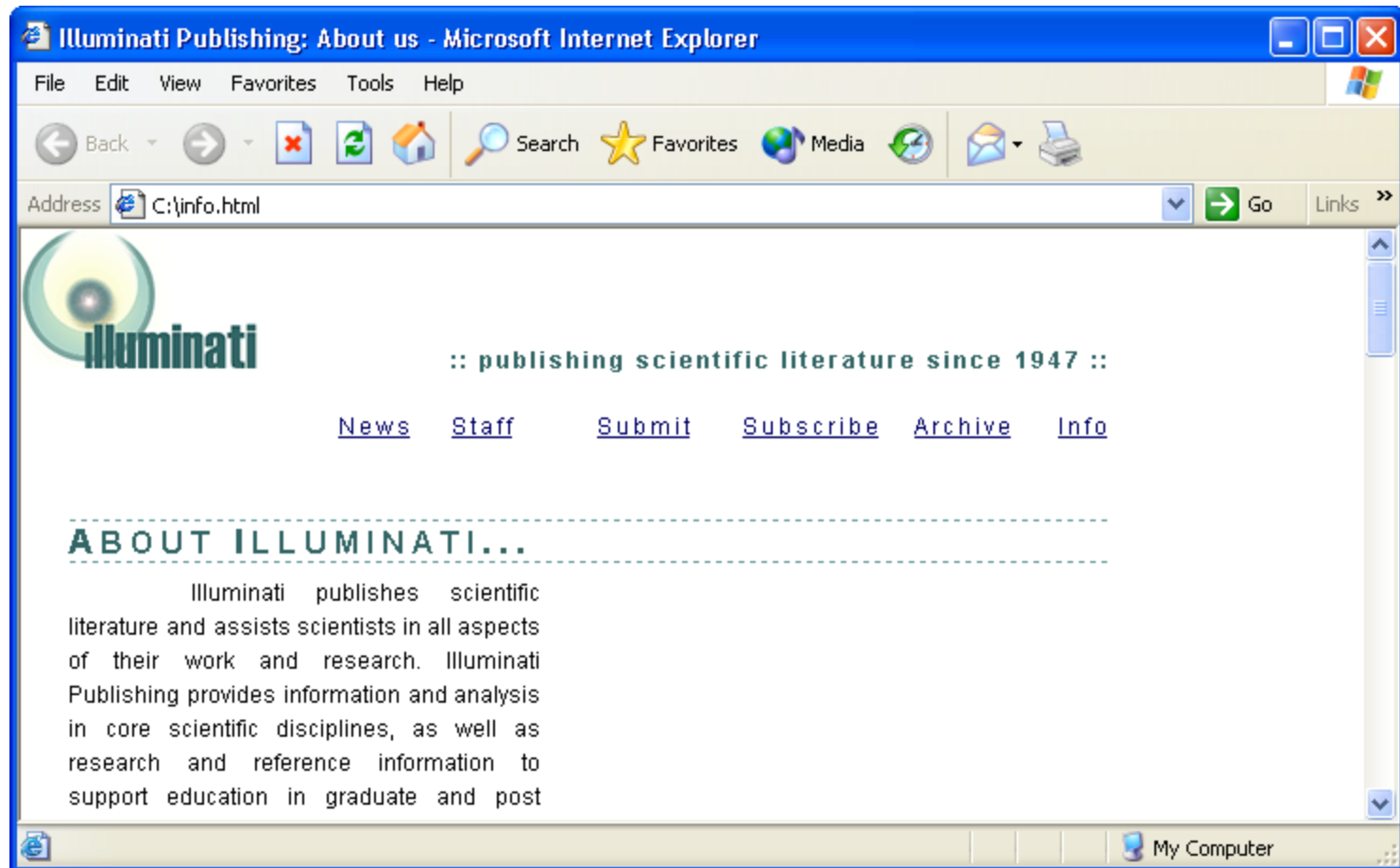
- 所有的元素都有預設的尺寸
- 在**CSS**中, 可以精準地控制 **height** 和 **width**
 - 在有限的頁面畫布中, 可用於節省空間, 安排元素
 - 在不同的頁面中保持一致的寬度
- 段落預設尺寸(請參考下頁圖示)
- 段落自訂尺寸(請參考下頁圖示)

Control Element Dimensions (p.121)



Paragraphs with default dimension

Control Element Dimensions (p.122)



Paragraph with customized height and width

Element Dimensions (p.122)

- 高度和寬度的屬性 • The height and width Properties
 - 如果使用絕對量測單位, 則元素的尺寸將會固定, 無法隨著瀏覽器的大小變動
 - 如果使用相對量測單位, 則元素的尺寸將會隨著瀏覽器的大小自動調整
 - 若只指定元素的寬度, 則元素的高度會自動隨著內容長度做調整
例: `div { border : 2px double gray; width : 250px; }`
(請參考教材 p.122 圖示2)
 - 如果設定元素的高度超過內容所需的高度
例: `div { border:2px double gray; width:250px; height:350px; }`
(請參考教材 p.123圖示1)
- 計算元素的尺寸 • Calculating an Element's Dimensions
 - 當宣告一個元素的寬度時, 它只定義內容的寬度, 並不包括 padding 及 border 的寬度.
 - 例: `div { width:200px; border: 4px solid black; padding: 8px; }`
則元素的總寬度是224px
(請參考教材 p.123 圖示2)

Activity 3-5 Controlling an Element's Width (p.124)

- Illuminati 網站專案進行得很順利，你仍需要利用 `styles.css` 在一些 `elements` 的 `dimensions` 做一點小改變。
- 特別地，你需要將 `heading` 的右邊 和 `paragraphs` 的右邊對齊
- 你也需要將相同的寬度套在 `copyright`，並且增加 `copyright` 的高度
 1. 使用 `width` 將第一階標題的框線對齊段落的右緣
 2. 將相同的寬度指定給版權頁尾
 3. 增加版權頁尾的高
 4. 將版權的文字填塞到盒子的中間

Create Floating Elements (p.127)

- 浮動元素 • Floating Elements

- 浮動元素意思是使用 **float** 屬性將元素對齊相鄰接元素的兩側
 - **left** : 強制浮動元素到相鄰接元素的左側
 - **right** : 強制浮動元素到相鄰接元素的右側

- **float**屬性 • The float Property

- 當一個<div>再接著另一個<div>, 預設值會上下排列
- 使用**float**屬性, 可以將一個元素浮動到另一個的左右
- 範例:

```
div1 { width: 250px;  
  border: 5px solid black;  
  padding: 5px;  
  float: left; }
```

(結果請參考教材 p.127圖示)

- 浮動子元素 • Floating Child Elements

- 當把子元素加上浮動, 則子元素會浮動在父元素內的一邊
(請參考教材 p.128 圖1)

The Clear Property (p.128)

- **clear**屬性控制一個元素是否允許有浮動元素在它的左右
 - 當使用**left**屬性值, 則沒有元素可以出現在它的左側
 - 當使用**right**屬性值, 則沒有元素可以出現在它的左側
 - 當使用**both**屬性值, 則沒有元素可以出現在它的左右

- 例

```
div p.first, div p.second { float : right; }  
div p.second { clear: right; }
```

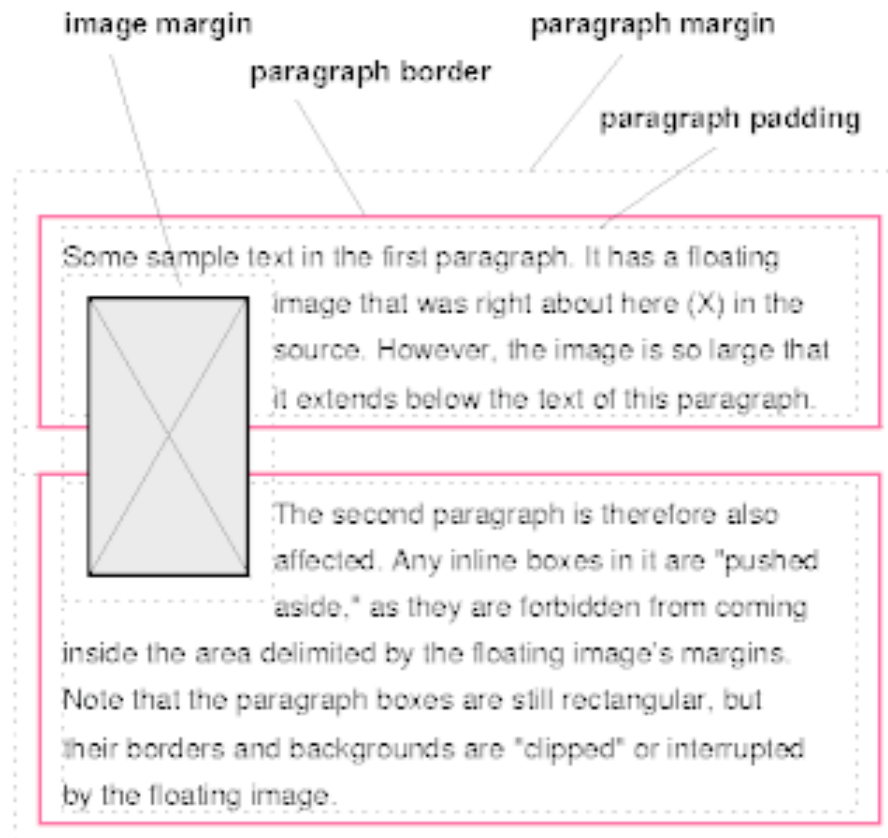
(結果請參考教材 p.128 圖2)

Activity 3-6 Arranging Elements with the Float Property (p.129)

- 你和專案 team 希望將 code-heavy outer layout tables 的方式(註：使用table做layout)，改為用 box properties 的方式來控制
- 專案 team 希望將 Top Story 的內容區段顯示在兩個 navigation box 的右側，且 Top Story 區段內需要做到文繞圖，文字在右的效果。
 1. 安排2個導覽區垂直相鄰, 並且在主要Top Story區段的左側
 2. 將圖像之後的文字圍繞在圖像的右側
 3. 增加圖像右側的 padding 這樣圖像和文字間才有足夠的空間

補充 : *float overlaps borders of elements in the normal flow*

- <http://www.w3.org/TR/CSS21/visuren.html#floats>
- Here is another illustration, showing what happens when a float overlaps borders of elements in the normal flow.



Control Content Overflow (p.132)

- 若你要強制指定元素的尺寸, 不管它包含的內容有多長, 則你可以該如何顯示溢的部份
- 內文溢位 • **Content Overflow**
 - 溢位發生在元素指定的高度不足以包含它的內容



The overflow property (p.132)

- overflow屬性 • The overflow Property
 - 屬性：overflow ; overflow-x ; overflow-y
 - 屬性值
 - scroll：捲軸 (即使內文沒有溢位, 也會出現捲軸)
 - hidden：溢位的部份隱藏
 - visible：溢位的部份照常顯示
 - auto：由瀏覽器自行判斷, 內文溢位就會出現捲軸)
 - inherit：繼承父元素的值
- 可捲動內文的區段 • Scrollable Content Sections
 - 可捲動內文的區段, 是指可以使用捲軸存取溢位內容的區段
 - 並不是所有的元素都可以運作. <div>是跨瀏覽器運作最穩定的元素
 - 範例：可捲動的表格(請參考下頁圖示)
- 套用overflow:auto 屬性 • Applying the overflow:auto Property
 - 套用 overflow:auto, 可以讓使用者存取資料, 又可以保持所希望的版面

Scrollable Content Sections (p.133)

Academic Schedule: (All Courses)

Course	Dept.	Instructor	Meets
Exobiology and Evolution	BIO	Dr. Kelly	T-TH
Why Shark Week is Cool	BIO	Joe Smith	Friday
Surviving Natural Selection	BIO	Dr. Fittest	M-W-F
DNA Made Easy	BIO	Dr. Wigginbotham	M-W-F
Introduction to Finger Counting	MAT	Dr. Fermat	T-TH
Advanced Finger Counting	MAT	Dr. Fermat	M-W-F

Activity 3-7 Creating a Scrollable Content (p.135)

- 設計指示：所有 Top Story 段落，必須有固定的寬和高，不管內容的長短。
- 在 `parrot.css` 中，你需要限制 Top Story 的高度，讓使用者利用scroll bar 讀取過長的內容
 1. 調整主要內容區段的高度, 讓它的頂部和第1個導覽區的頂部對齊, 它的底部和第2個導覽區的底部對齊
 2. 當需要的時候出現捲軸

Reflective Questions (p.136)

1. Analyze a website of your choice that does not have appropriately designed content sections. What properties would you apply and why?
2. Explain how margins and padding will improve the design?