

## Response物件輸出網頁內容

資料系  
林偉川

## 輸出HTML標籤與文字內容

- Response.Write()方法可以在ASP.NET程式將任何資料型態輸出到瀏覽程式顯示，換句話說，輸出的內容可以是字串、HTML標籤或變數，而且不論變數資料為何，都是使用字串形式輸出到瀏覽程式，如下所示：

```
Response.Write(title & "<br>")
```

```
Response.Write("歡迎使用ASP.NET設計動態網頁<br>")
```

## 輸出HTML標籤與文字內容

- Response.Write()方法擁有另一種形式，可以直接插入HTML標籤，以便更改標籤的屬性或內容，如下所示：

```
<%=title %>
```

- 「<%=」和「%>」符號間就是輸出變數title內容，屬於Response.Write()方法的簡化寫法。

3

## 轉向到其它的網頁或ASP.NET程式

- Response.Redirect()方法可以在執行ASP.NET程式時，轉向到其它的網址或網站的其它網頁或ASP.NET程式，因為轉向到其它網頁，所以執行此方法後的指令或HTML標籤並不會顯示，如下所示：

```
Response.Redirect("Ch9-1-1.aspx")
```

- 參數允許使用完整的URL網址或同一個網站檔案的相對路徑，也可以是其它HTML文件，如下：

```
Response.Redirect("../Ch08/Ch8-1-1.aspx")
```

```
Response.Redirect("http://www.hinet.net")
```

4

## 中斷ASP.NET程式的執行

- ASP.NET程式主要是使用Response.Write()方法輸出網頁內容，如果程式需要中斷輸出，只需在程式的中斷點插入Response.End()方法，其程式碼如下：

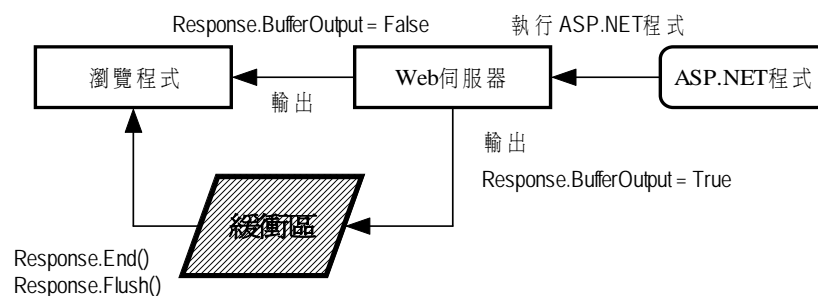
```
.....  
Response.End()  
.....
```

- 輸出指令前的HTML標籤或執行的ASP.NET程式，Response.End()方法後的程式碼或HTML標籤就不會執行或顯示，因為ASP.NET會使用緩衝區，此方法會將緩衝區的內容送到瀏覽程式顯示。

5

## ASP.NET緩衝區的用途

- Web伺服器在執行ASP.NET程式碼時，可以選擇將處理的結果馬上輸出到瀏覽程式，或是先輸出到暫放的緩衝區，等到ASP.NET程式執行完後，才送到瀏覽程式顯示，如下圖所示：



6

## ASP.NET緩衝區的用途-輸出到緩衝區

- 當遇到下列的3種情況才會將輸出的內容送到瀏覽程式，如下所示：
  - 執行完ASP.NET程式後。
  - 執行到Response.End()方法中斷ASP.NET程式的執行，此時已執行的部分就會送出。
  - 執行到Response.Flush()方法馬上將緩衝區的內容送出。

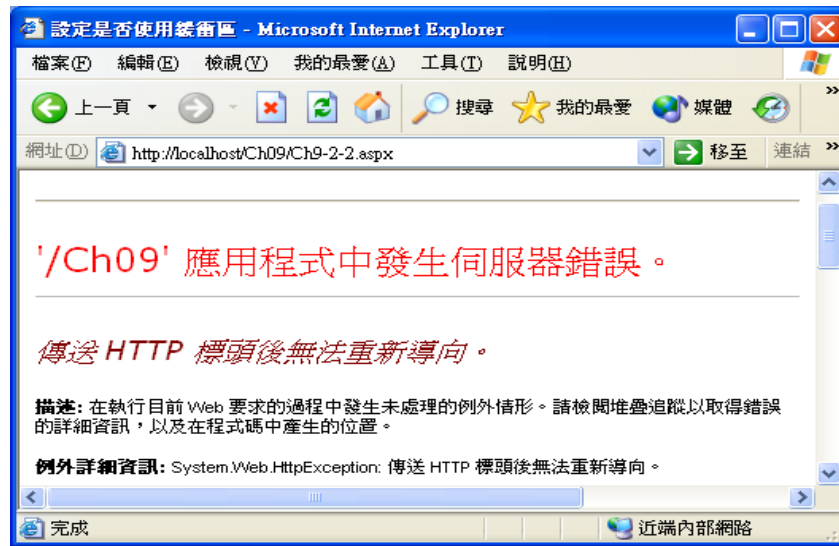
7

## 設定是否使用緩衝區

- Response.BufferOutput屬性可以指定是否使用緩衝區，預設值True，表示使用，如果不使用緩衝區，請設為False，如下所示：  
`Response.BufferOutput = False`
- 程式碼中若不使用緩衝區，一旦程式在執行轉向時，就會出現標題寫入錯誤。

8

## 設定是否使用緩衝區-錯誤訊息



## 馬上輸出緩衝區的內容

- ASP.NET程式預設使用緩衝區，其輸出的內容是先送到緩衝區，等到程式執行完成後，才將緩衝區的內容一起送到瀏覽程式顯示。如果需要將目前的緩衝區內容，在尚未執行完ASP.NET程式前強迫輸出到瀏覽程式，可以使用**Response.Flush()**方法。
- 當ASP.NET程式執行到上述方法，馬上就會將目前緩衝區的內容都送到瀏覽程式顯示。

## 清除緩衝區的內容

- 如果伺服器緩衝區的內容已經不需要，或是根本不需要將緩衝區內容輸出到瀏覽程式顯示，我們可以在ASP.NET程式使用 **Response.Clear()** 方法清除緩衝區內容，不過這並不含HTTP通訊協定的標頭資料。
- 在執行上述Response.Clear()方法後，和上一次執行Response.Flush()方法間輸出的網頁內容就會被清除掉。

11

## ASP.NET輸出快取的處理

- 「快取」(Caching) 是一種技術可以將網頁輸出或是應用程式本身的資料暫時儲存在客戶或伺服器端，以便當重複請求相同資訊的網頁時，可以加速網頁的顯示，特別適用在多次網頁請求，而且都是請求相同網頁內容的情況。

12

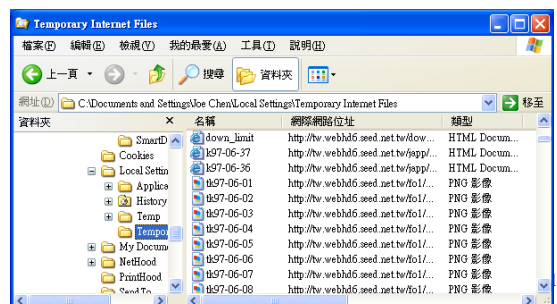
## 輸出快取的用途

- 輸出快取是當第一次編譯執行ASP.NET程式時，先將執行結果的網頁內容儲存在輸出快取，然後再輸出到瀏覽程式，當有其它請求時，如果程式沒有更改或已經是快取中已經存在網頁，就直接從快取將網頁資料送到客戶端，如此可以增加網頁瀏覽的效率。

13

## 輸出快取的用途-儲存位置

- 輸出快取可以將資料儲存在伺服器端、客戶端或Proxy伺服器，以客戶端快取為例，是將網頁內容保留在使用者電腦「C:\Documents and Settings\Joe Chen\Local Settings\Temporary Internet Files」的快取資料夾。



4

## 設定保留在快取的時間

- `Response.Expires` 屬性可以設定和取得網頁保留在瀏覽程式快取的分鐘數，如果使用者在保留期限前請求此網頁，顯示的是儲存在快取的網頁，如果超過期限，才會連線伺服器取得最新的網頁內容，其程式碼如下所示：

```
Response.Expires = 3
```

- 程式碼設定保留的時間為3分鐘，如果設為0表示不保留在快取。

15

## 設定在快取保留的期限

- `Response.ExpiresAbsolute` 屬性可以指定和取得網頁保留期限的日期和時間，其程式碼如下所示：

```
Response.ExpiresAbsolute = #12/31/2003#
```

- 屬性值是 `Date` 資料型態，請使用 # 符號括起，上述的保留期限是2003年底。

16



## ASP.NET的快取處理

- 在ASP.NET程式可以使用**Response.Cache**屬性取得網頁輸出快取的**HttpCachePolicy**物件，內含數種方法可以控制ASP.NET網頁的輸出快取，如下表所示：

方法	說明
SetExpires(DateTime)	指定快取的期限
SetCacheability(HttpCacheability)	指定快取控制的方式

17

## ASP.NET的快取處理

- ASP.NET程式的快取設定需要位在Page物件的Page\_Load事件處理程序，如下所示：

```
Sub Page_Load(Sender As Object, e As EventArgs)
    Dim dtDay As Date = Today
    Response.Cache.SetExpires(DateAdd("D", 5, dtDay))
    Response.Cache.SetCacheability(HttpCacheability.Public)
End Sub
```

- Page\_Load事件處理程序使用Response.Cache屬性取得物件，然後使用**SetExpires()**方法設定期限，使用DateAdd()函數新增5天，期限是目前日期加上5天。

18

## ASP.NET的快取處理- SetCacheability()方法參數

參數值	說明
HttpCacheability.NoCache	使用伺服器端或 Proxy 伺服器的快取
HttpCacheability.Private	只使用客戶端快取，不包括 Proxy 伺服器，此為預設值
HttpCacheability.Public	可以使用客戶端或 Proxy 伺服器的快取
HttpCacheability.Server	只使用伺服器端快取

19

## 設定輸出文件內容的資料類型

- **Response**物件是使用**HTTP通訊協定**傳送資料到瀏覽程式，雖然通訊協定與資料類型並沒有關係，不過在傳送資料前，需要設定資料種類 **Response.ContentType**屬性，其值是**MIME資料類型**。
- **text/html**為預設值，如果需要輸出成其它的文件類型時，我們可以設定**MIME資料類型**，其程式碼如下所示：  

```
Response.ContentType = "text/xml"
```
- 程式碼指定傳送的內容為XML文件。

20

## 設定輸出文件內容的資料類型- MIME資料類型

MIME 資料類型	說明
text/html	HTML 文件的網頁檔案
text/xml	XML 文件的檔案
text/plain	ASCII 一般的文字檔，瀏覽程式並不會直譯內容
image/jpeg	JPEG 格式的圖片檔
image/gif	GIF 格式的圖片檔

21

## 檢查客戶端是否正在連線

- HTTP通訊協定並不會持續保持伺服器 and 瀏覽程式間的連線，而是每當瀏覽程式向伺服器提出請求時才會建立伺服器和瀏覽程式間的連線，Response物件的isClientConnected屬性能夠檢查目前客戶端的連線狀態，其檢查的程式碼，如下所示：

```
If Response.IsClientConnected = True Then
    Response.Write("使用者正在連線!")
Else
    Response.Write("使用者已經斷線!")
End If
```

22

## 取得網頁的編碼方式

- Response物件的Charset和ContentEncoding屬性都可以取得或指定網頁的編碼方式，如下所示：

```
Response.Write(Response.Charset & "<br>")
```

```
Response.Write(Response.ContentEncoding.EncodingName)
```

- 程式碼分別可以顯示編碼方式的字元集，其差異在第2列程式的ContentEncoding屬性可以取得Encoding物件，EncodingName屬性是完整的編碼名稱。

23

## ASP.NET程式的除錯模式

- 在錯誤的ASP.NET程式檔案開頭加上Page指引的Debug屬性，如下所示：

```
<%@ Page Language="VB" Debug="True" %>
```

- 程式碼指出使用的程式語法是VB.NET，Debug屬性True開啟除錯模式，當執行ASP.NET程式就可以顯示進一步的除錯資料。

24