

10

靜態和動態網頁

目標

在完成這章後，你將能夠

- ◆ 區別靜態網頁和動態網頁
- ◆ 知道除 HTML 之外，其他代碼亦可在一個網頁檔案中存在
- ◆ 體會動態網頁的好處
- ◆ 了解客戶端和伺服器端處理的不同
- ◆ 了解使用伺服器端處理可保障數據傳輸的安全

10.1 靜態網頁

靜態網頁 (Static web page) 只有文本、圖形等多媒體元素及一些連結，用戶毋需輸入任何數據，因此靜態網頁通常不會包括任何填寫式表格。靜態網頁只有少量的互動性，與一般印刷刊物無異。

靜態網頁檔案包含的全是 HTML 代碼，並由客戶端電腦進行解譯，所以傳送到客戶端電腦的檔案，與存貯在網站伺服器中的，並無分別。

10.2 動態網頁

動態網頁 (Dynamic web page) 卻含有更多互動性，根據用戶的輸入，進行分析及作出回應。由於動態網頁涉及電腦程式，因此用戶可利用它作計算，甚至更新數據庫。運算的結果通常透過網頁，傳回給用戶。

提示

注意：

不可把「經編譯的程式」混為「編譯程式」，後者是翻譯器的一種。見核心課程第二冊頁 89。

A. 經編譯的程式

動態網頁使用的程式可能是經過編譯的，例如 Flash[®] 和 Java 都是經編譯後的獨立程式。這些程式會連同其他多媒體元素檔案一起傳送到客戶，由客戶端電腦執行，客戶的瀏覽器必須裝有適當的外掛軟件（例如 Flash[®] 播放器）才可順利執行這些程式；CGI 則是長駐於網站伺服器經編譯的程式，由網站伺服器來執行。

B. 指令碼

除上述經編譯的程式外，動態網頁所使用的程式也可以和 HTML 代碼混合。這些程式稱為**指令碼 (Script)**，意謂它們並非由電腦直接執行，而是由瀏覽器或伺服器軟件來執行。例子是 JavaScript、VBScript、ASP 和 PHP。

換言之，這些網頁檔案並不屬於純 HTML。與 HTML 混合使用的程式編寫稱為「動態 HTML」或 DHTML。

10.3 靜態的和動態網頁之間的比較

	靜態網頁	動態網頁
部件	HTML 程式碼	HTML 程式碼及程序
程序編寫	不用編寫程序	需要編寫程序
互動性	低	高
數據輸入	不用輸入數據	輸入數據通常由表格收集

表 1 靜態的和動態網頁之間的比較

10.4 動態網頁的種類

動態網頁可根據執行其程式的電腦的位置，分為兩類型：**客戶端 (Client-side)** 和**伺服器端 (Server-side)**。

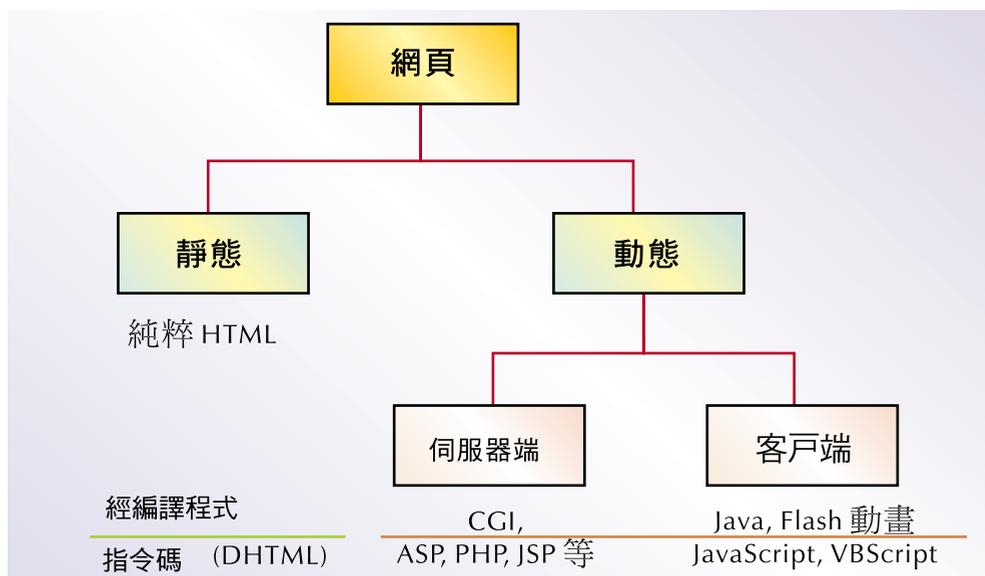


圖 1 網頁的類型

A. 客戶端處理

客戶端處理 (Client-side processing) 意謂程式由客戶電腦執行。常用的客戶端程式編寫語言包括 JavaScript 和 VBScript，由於這些程式的源碼包含在 HTML 檔案中，用戶是可以檢視這些程式的。

Flash® 動畫和 Java 也屬於客戶端處理的。但是，由於它們已被編譯，用戶是不能夠檢視這些程式的源碼。

1. 回應速度

客戶端動態網頁的回應是比較快捷的，原因如下：

1. 客戶端電腦比網站伺服器較清閒
2. 減少數據經互聯網傳輸

客戶端處理	
1.	Flash 動畫
2.	Java
3.	JavaScript
4.	VBScript

表 2 客戶端處理的例子

2. 客戶端處理的應用

客戶端程式可以為網頁增加多媒體效果，例如動畫圖形、捲動訊息、日曆和廣告橫幅等，令網頁更具生氣。

客戶端程式亦可以對輸入數據在傳送前，作**有效性檢驗 (Validation)**，例如檢驗數據的類型、範圍和完整性等。



圖 2 Flash 是客戶端動態網頁的一個例子。注意當游標移動到文字之上時，它們如何改變。



圖 3 客戶端指令碼可以對輸入數據作「有效性檢驗」，檢驗數據的完整性（在這例子中，成員號碼 member ID 受到檢驗）



B. 伺服器端處理

伺服器端處理 (Server-side processing) 意謂程式由網站伺服器來執行。這些網頁，通常利用填寫式表格來收集數據。用戶在按一下「送出」按鈕後，數據會傳送給伺服器進行處理。最後傳回給客戶的，只是一個純 HTML 檔案，並不包含程式源碼。

伺服器的程式可以是經編譯的 (例如 CGI)，也可以是混合 HTML 的指令碼，並存貯在文字檔中，例如 ASP、JSP 或 PHP。

任何情況下，用戶均無法檢視這些程式的源碼。

	縮寫	英文全名
1.	CGI	Common Gateway Interface
2.	ASP	Active Server Page
3.	JSP	JavaServer Page
4.	PHP	PHP: Hypertext Preprocessor

表 3 伺服器端處理的例子

1. 伺服器端處理的安全性

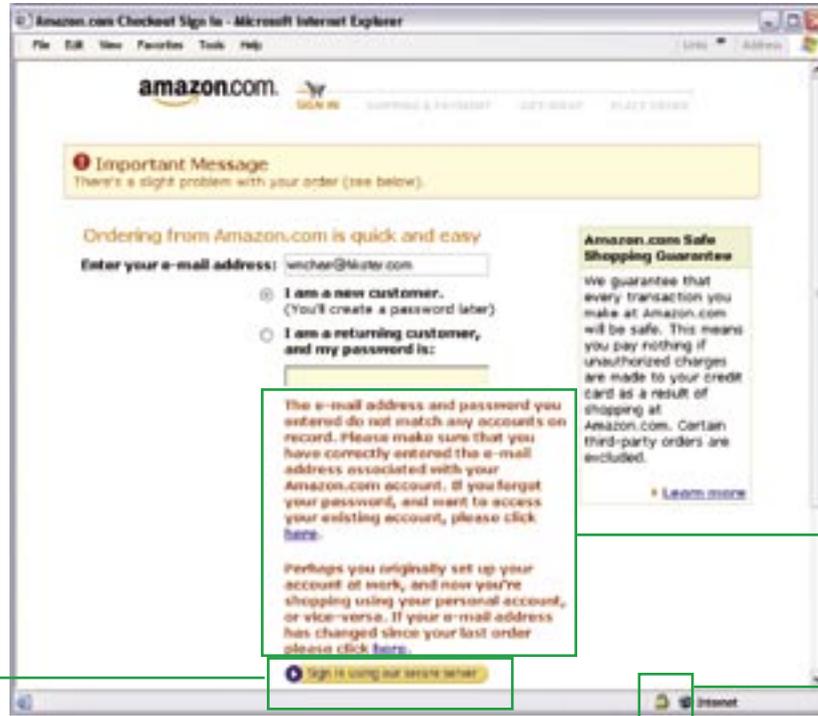
用戶在檢視 ASP 網頁的源碼時，是不會看到任何程式編寫的蹤跡，即使傳回的檔擴展名是 ".asp"。由於程式源碼是不公開的，伺服器端處理較為安全。另外，經加密的數據必須由伺服器解密及處理，所以伺服器端處理讓數據加密得以實踐，從而提高數據傳輸的安全性。

2. 伺服器端處理的應用

伺服器端處理在商業上的應用非常廣泛，理由如下：

1. 有高度的安全性 (如上述)，
2. 能存取網站伺服器的數據庫。

鑒別 (Authentication) 是伺服器端處理眾多應用中，最普遍的一種，當用戶提交登入號碼和密碼後，伺服器會從數據庫中，搜尋用戶的記錄，經核實後，才讓用戶進入系統。



安全的伺服器意謂只有這個伺服器才能將那些已加密的數據解密

檢查已經在伺服器上進行了。這些警告訊息是由伺服器編寫的。

閉鎖圖像顯示數據在送出前會進行加密

Fig.4 伺服器端處理用於電子商務上，鑒別用戶。

C. 客戶端處理和伺服器端處理的比較

	客戶端	伺服器端
進行處理的電腦	客戶端電腦	網伺服器
回應速度	快速	較慢
伺服器的工作量	低	高
安全性	低	高
存取伺服器數據庫	不可以	可以
使用者可檢視程式碼	指令碼：是 經編譯程式：否	指令碼：否 經編譯程式：否
例子	JavaScript, VBScript, Flash 動 畫, Java	CGI, ASP, JSP, PHP

表 4 客戶端處理和伺服器端處理之間的比較



Dynamic Web Page
示範



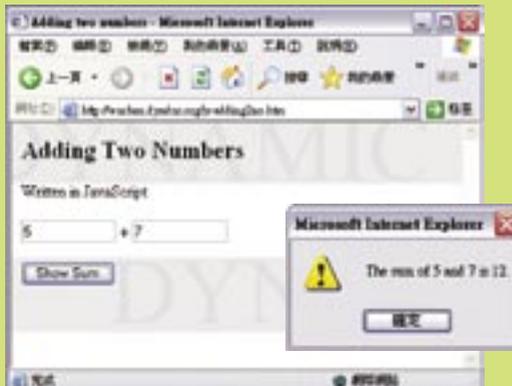
JavaScript：兩數相加

我們將要編寫一個網頁，利用 JavaScript 將兩數相加，如圖甲所示：

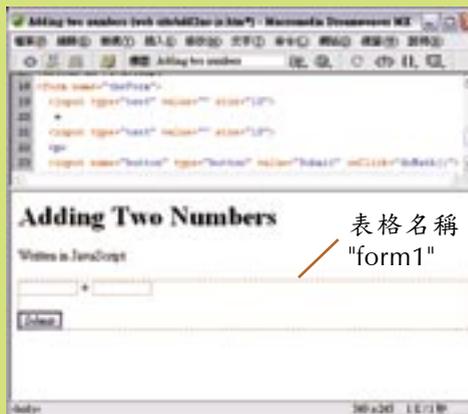
1. 啟動 DreamWeaver。
2. 建立一個新的文件，並加入如圖乙所示的表格物件。
 - 將表格命名為 "form1" (預設)
3. 加入 JavaScript：
 - 轉到「程式碼」(Code) 檢視
 - 鍵入圖丙紅色框內的文字

注意：由於 JavaScript 對大小寫是有分別的，所以當心使用大小寫。

4. 存檔，並命名為 "add2no-js.htm"
5. 測試結果。
 - 按 F12 鍵



圖甲



圖乙

```

<html>
<head>
<title>Untitled Document</title>
<SCRIPT LANGUAGE="JavaScript">
function doMath(){
    var x1 = eval(this.form1.elements[0].value)
    var x2 = eval(this.form1.elements[1].value)
    var y = x1+x2
    alert("The sum of "+x1+" and "+x2+" is "+y+".")
}
</SCRIPT>
</head>
<body>
<h2>Adding Two Numbers</h2>
<p>Written in JavaScript </p>
<form action="" method="post" name="form1" id="form1"><p>
    <input name="textfield" type="text">
    +
    <input name="textfield2" type="text"> </p><p>
    <input type="submit" name="Submit" value="Submit" onClick="doMath()"> </p>
</form>
</body>
</html>
    
```

將這些文字加入到
你的網頁

圖丙

Dynamic Web Page
示範



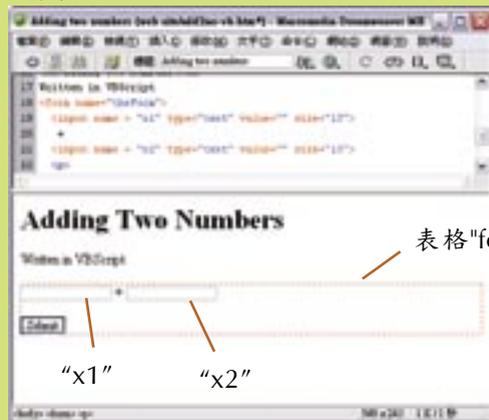
VBScript : 兩數相加

我們將要編寫一個網頁，利用 VBScript 將兩數相加，如圖甲所示：

1. 啟動 DreamWeaver。
2. 建立一個新的文件，並加入如圖乙所示的表格物件。
 - 將表格命名為 "form1" (預設)
 - 將第一個文本欄位命名為 "x1"
 - 將第二個文本欄位命名為 "x2"
 - 將送出 (submit) 按鈕命名為 "submit" (預設)
3. 加入 VBScript :
 - 轉到「程式碼」(Code) 檢視
 - 鍵入圖丙紅色框內的文字
4. 存檔，並命名為 "add2no-vb.htm"
5. 測試結果。
 - 按 F12 鍵



圖甲



圖乙

```

<html>
<head>
<title>Untitled Document</title>
<SCRIPT LANGUAGE="VBScript">
sub submit_onClick
    x1 = form1.x1.value
    x2 = form1.x2.value
    y = clnt(x1) + clnt(x2)
    MsgBox "The sum of "&x1&" and "&x2&" is "&y&".
end sub
</SCRIPT>
</head>
<body>
<h2>Adding Two Numbers</h2>
<p>Written in VBScript</p>
<form name="form1" method="post" action=""> <p>
    <input name="x1" type="text" id="x1">
    +
    <input name="x2" type="text" id="x2"> </p><p>
    <input type="submit" name="Submit" value="Submit"> </p>
</form>
</body>
</html>
    
```

將這些文字加入
到你的網頁

圖丙

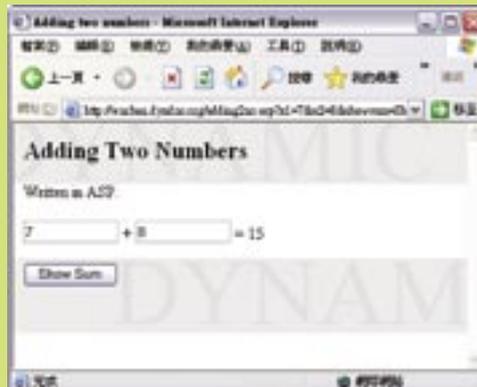
Dynamic Web Page
示範



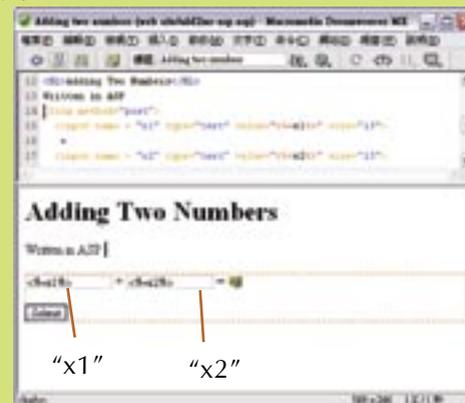
ASP：兩數相加

我們將要編寫一個網頁，利用 ASP 將兩數相加，如圖甲所示：

1. 啟動 DreamWeaver。
2. 建立一個新的文件，並加入如圖乙所示的表格物件。
 - 為表格的「行動」(Action) 屬性，鍵入 "add2no-asp.asp"
 - 為「方法」(Method) 屬性，選擇 GET
 - 將第一個文本欄位命名為 "x1"
 - 將 "<%=x1%>" 放進 x1 內
 - 將第二個文本欄位命名為 "x2"
 - 將 "<%=x2%>" 放進 x2 內
3. 加入 ASP script：
 - 轉到「程式碼」(Code) 檢視
 - 鍵入圖丙紅色框內的文字
4. 存檔，並命名為 "add2no-asp.asp"
5. 將網頁上載到網站伺服器上。使用你的瀏覽器進行測試。



圖甲



圖乙

```
<%
x1 = request("x1")
x2 = request("x2")
y = clnt(x1)+clnt(x2)
%>
```

將這些文字加入
到你的網頁

```
<html>
<head>
<title>Untitled Document</title>
</head>
<body>
<h2>Adding Two Numbers</h2>
<p>Written in ASP.</p>
<form name="form1" method="get" action="add2no-asp.asp"> <p>
  <input name="x1" type="text" id="x1" value="<%=x1%>">
  +
  <input name="x2" type="text" id="x2" value="<%=x2%>">
  = <%=y%>
</p>
  <p> <input type="submit" name="Submit" value="Submit"> </p>
</form>
</body>
</html>
```

圖丙



摘要

1. 靜態網頁只有文本、圖形和連結，並有小量互動性而已。
2. 動態網頁有更多互動性，並且包括執行一些電腦程式。
3. 動態網頁使用的程式可以是經過編譯或是指令碼。
4. 指令碼是由瀏覽器或伺服器軟件來執行的。
5. 與 HTML 混合使用的程式編寫稱為動態 HTML 或 DHTML。
6. 在客戶端處理中，程式是由客戶端電腦進行處理。例子有 JavaScript, VBScript, Flash 動畫及 Java。
7. 在伺服器端處理中，程式由網站伺服器執行。例子有 CGI, ASP, JSP 及 PHP。
8. 一般而言，客戶端處理比較快速，伺服器端處理則提供安全數據傳輸及鑒別。



練習

多項選擇題

1. 可打印版本通常是一個
 - A. 靜態網頁
 - B. 客戶端處理網頁
 - C. 伺服器端處理網頁
 - D. 以圖形為主的網頁
2. 下列哪項時常出現在動態網頁上？
 - A. 文本
 - B. 表格
 - C. 框
 - D. 按鈕
3. CGI 程式可能由下列哪項調用？
 - A. 靜態網頁
 - B. 客戶端處理網頁
 - C. 伺服器端處理網頁
 - D. 電子郵件軟件
4. 客戶端處理網頁
 - (1) 需要比較長的下載時間。
 - (2) 增加客戶端電腦的工作量。
 - (3) 增加網站伺服器的工作量。
 - A. 只有 (1)
 - B. 只有 (2)
 - C. 只有 (1) 和 (2)
 - D. 只有 (1) 和 (3)
5. 伺服器端處理網頁
 - (1) 能存取網站伺服器的數據庫。
 - (2) 把程式傳送到客戶端的瀏覽器中。
 - (3) 要求網站伺服器去執行一個程式。
 - A. 只有 (1)
 - B. 只有 (2)
 - C. 只有 (1) 和 (2)
 - D. 只有 (1) 和 (3)
6. 對於由客戶的瀏覽器送出加密的數據，下列哪項敘述是正確的？
 - (1) 加密和解密俱由網站伺服器進行。
 - (2) 加密後的數據由網站伺服器解密。
 - (3) 即將傳送的數據由客戶的瀏覽器加密。
 - A. 只有 (1)
 - B. 只有 (2)
 - C. 只有 (1) 和 (2)
 - D. 只有 (2) 和 (3)



7. 某個網頁懂得將兩個輸入的數字相加，但是當檢視網頁源碼時，只發現簡單的HTML代碼，並沒有任何程式編寫的痕跡。這網頁定必涉及
- 客戶端處理。
 - JavaScript。
 - 伺服器端處理。
 - ASP。
8. 假設用戶沒有為文本欄位鍵入任何的內容，就按下「送出」按鈕。在這情況下，一個錯誤訊息顯示了出來。這是由下列哪項完成？
- 客戶端處理
 - 伺服器端處理
 - DNS 伺服器
- 只有 (1)
 - 只有 (2)
 - 只有 (1) 和 (2)
 - 只有 (2) 和 (3)
9. 某填寫式表格使用了 GET 的方法。下列哪項敘述是正確的？
- 即將傳輸的數據經已加密
 - 數據可在網址欄位中看到
 - 表格內的數據將被重新設定
- 只有 (1)
 - 只有 (2)
 - 只有 (1) 和 (2)
 - 只有 (2) 和 (3)
10. 某個網頁聲稱可保證數據傳輸安全。
- 數據並不是由填寫式表格輸入
 - 傳輸規約是 https
 - 數據將被加密
- 只有 (1)
 - 只有 (2)
 - 只有 (1) 和 (2)
 - 只有 (2) 和 (3)

問答題

注意：部分題目可能涉及其他章節的內容。

- 某學校的網站是由校務處職員管理的。當校務處職員收到新的告示，他們會立即更新網站上的告示板。告示板分為三部分：學生、老師和家長，並出現在不同的網頁上。網站是開放予公眾的。任何人都可檢視他們喜歡的部分。
 - 校務處職員需要懂得什麼技術？ (2 分)
 - 靜態網站和動態網站之間主要的分別是什麼？ (1 分)
 - 上述的告示板是否涉及動態網頁？試解釋。 (2 分)
 - 相同的告示可能同時出現在上述三個部分。
 - 舉出這類告示的其中一個例子。 (1 分)
 - 試討論是否需要有三個相同的檔案。 (2 分)
 - 在教員室內，傳統的告示板上仍然貼滿告示。試提供一個理由。 (1 分)

最近，學校聘請了一名資訊科技技術員，為網站重新設計。其中一個改變是，校外人士再不能夠檢視告示板了。此外，家長和學生亦不再能存取老師那部分的資訊。

- 詳細討論網站如何區分用戶的類別。 (2 分)
- 網站伺服器現在的功能比以前更強大。試寫出兩個以前沒有的功能。 (2 分)
- 該資訊科技技術員必須為網頁編寫一些程式。舉出一個他可能會使用的語言，並解釋你的答。 (2 分)

2. 某銀行的網上銀行服務讓顧客透過家中電腦，處理過戶、投資和檢查結餘等財政事務。顧客必須鍵入正確的顧客帳號和密碼，才可使用這服務。

網上銀行服務充分利用動態網頁，包括伺服器端及客戶端處理。

- (a) 為上述每類動態網頁，舉出一個適當的程式語言例子。 (1 分)

- (b) 當用戶鍵入密碼時，所有鍵入的字元都以圓點形式來顯示。這是動態網頁的功能嗎？若是的話，它屬於伺服器端還是客戶端呢？ (1 分)

- (c) 在用戶的瀏覽器上，網頁的源碼包括下列的程式語句：

```
var errHead="Your request cannot be processed due to the following error(s):\n\n";  
var Error_01 = "User ID should not be blank. Please re-enter.";  
var Error_02 = "Password should not be blank. Please re-enter.";
```

- 這些程式屬於伺服器端，還是客戶端呢？ (1 分)

當用戶按一下「確定」按鈕時，瀏覽器上的帳號欄和密碼欄旋即變為空格，抹掉用戶所鍵入的資料，然後才顯示新的網頁。

- (d) 「確定」按鈕屬於那類型的表格物件？ (1 分)

- (e) 將欄的內容抹掉的功能屬於動態網頁嗎？若是的話，它是伺服器端還是客戶端？ (2 分)

- (f) 討論顧客的帳號和密碼被盜取的後果。因此，解釋為什麼要抹掉欄內的內容。 (2 分)

- (g) 顧客如何能知道傳送到網上銀行的數據將會受到保護？ (1 分)

- (h) 利用填寫式表格來送交數據，方法有兩個。

- i) 寫出這兩個方法，並作簡短的描述。 (2 分)

- ii) 網上銀行應該用那個方法？解釋。 (2 分)

- (i) 部分顧客在使用完服務後忘記登出，建議一個解決方案。 (2 分)

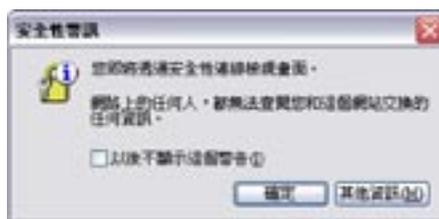


3. 「書齋」是一家大型網上書店。網上購買書本的程序如下：
1. 將貨品項目放入到購物籃
 2. 按一下「到付款處」
 3. 鍵入電郵位址和密碼
 4. 提供送貨地址
 5. 鍵入信用卡資料，並確認交易

購物籃記錄顧客有意購買的貨品。這些記錄可能存貯於顧客的電腦或伺服器中。

- (a) 討論將購物籃的記錄存貯在顧客電腦的**優點**。 (2 分)
- (b) 寫出**兩種**購物籃所存貯的資訊。 (2 分)

用戶在完成步驟 2 後，以下的訊息視窗將會彈出：



這個訊息說明數據傳輸時會受到「安全連線」的保護。

- (c) 「安全連線」的意義是什麼？ (1 分)
- (d) 舉出一個受保護的資訊。 (1 分)

由於初次光顧的客人並沒有任何密碼，他們必須先進行登記。在「書齋」中，電郵位址便是顧客記錄的關鍵字段。

- (e) 關鍵字段的用途是什麼？解釋電郵位址為什麼可用作關鍵字段。 (1 分)
- (f) 討論以電郵位址作為關鍵字段的**優點**和**缺點**。 (2 分)
- (g) 「書齋」允許用戶設定密碼。寫出一般密碼設定的規則。 (2 分)
- (h) 解釋中文字元為什麼**不應該**用於密碼上。 (1 分)
- (i) 解釋除了電郵位址之外，為什麼還需要密碼。 (1 分)
- (j) 當新的顧客登記註冊時，他們要提供居住地址。討論在顧客記錄內存貯顧客居住地址的**優點**。 (1 分)
- (k) 「書齋」並不存貯顧客的信用卡號碼。事實上，「書齋」甚至不知道這些號碼。顧客輸入的信用卡號碼傳送到那裏去？ (1 分)

4. 「快速遞送」提供國際性的遞送服務，從接收寄件人的貨物開始便不斷地追蹤貨物的位置。透過互聯網，寄件人可以隨時檢視貨物的位置。

(a) 提供**兩個**理由，解釋伺服器端處理為什麼是必需的。 (2 分)

(b) 為了更新伺服器內的資訊，送貨職員需要攜帶些什麼設備？ (2 分)

「快速遞送」的主頁是由專業的網頁設計師建立。用戶漫遊網站時，橫幅總是出現在瀏覽器相同的位置。網頁上資料很豐富，並恰當地排列在屏幕上。當滑鼠指標移近一些選單時，一列選單項目便會自動從選單放下。

(c) 寫出**兩個**可用於網頁版面配置的 HTML 功能。 (2 分)

(d) 舉出一個賦予網頁互動性的程式編寫語言。 (2 分)

在正式購買服務前，顧客可以自行估計所需費用。他們只需在網頁上輸入目的地、貨物重量和服務類別。

(e) 討論使用客戶端處理去完成費用估計的**優點**和**缺點**。 (2 分)

最近，「快速遞送」提供了一個服務：當貨物到達目的地後，寄件者可以看到收件人的簽名。

(f) 寫出存貯收件人的簽名最合適的格式，並解釋答案。 (2 分)

顧客必須使用信用卡付款，繳費後，顧客可檢視收據。

(g) 用於顯示收據的網頁與一般網頁是有分別的。寫出這種網頁的**三個**特徵。 (3 分)

