

---

# Java EE 7 Web 練習八 Solution

---

鄭安翔

ansel\_cheng@hotmail.com

# Lab 1 – 標準動作 UseBean

- 建立 ContactInfo 網路應用專案
- 撰寫 index.jsp
  - Form 表單, Action = setAddress.jsp
  - 五個請求參數 address1 / address2 / city / country / postcode
- 撰寫 domain.Address.java
  - 五個屬性 address1 / address2 / city / country / postcode

# Lab 1 – 標準動作 UseBean

## ■ 撰寫 setAddress.jsp

- 使用<jsp:useBean>標準動作標籤,來建立Java Bean Address
  - Scope設為session
- 使用<jsp:setProperty>標準動作標籤,取得請求參數,並設定Java Bean中同名的屬性
- 列印設定成功訊息
- 使用<jsp:getProperty>標準動作標籤顯示設定值

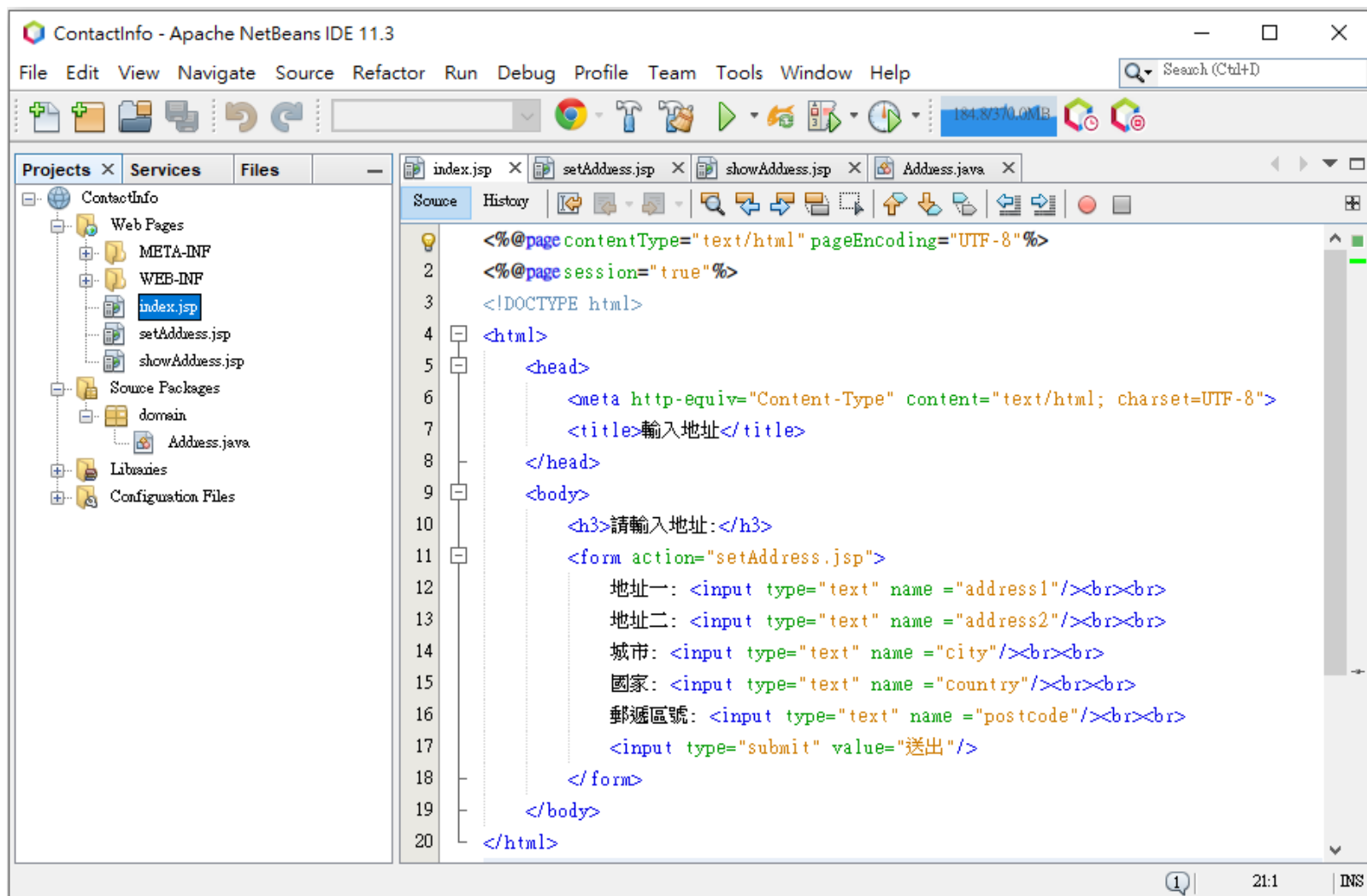
## ■ 撰寫 showAddress.jsp

- 使用<jsp:useBean>標準動作標籤,來取得Session中的 Address Java Bean
  - Scope設為session
- 使用<jsp:getProperty>標準動作標籤,顯示取得之Java Bean指定屬性值

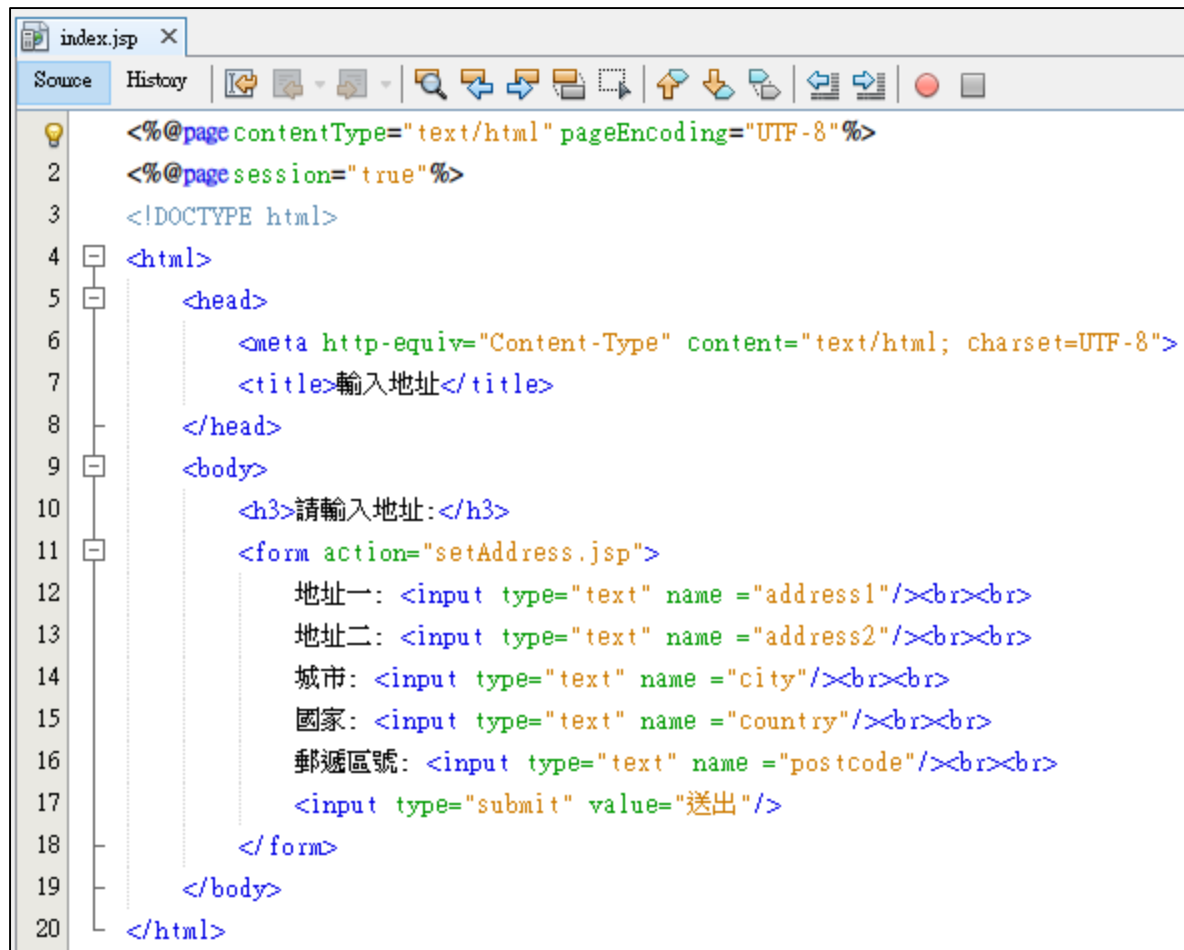
# Lab 1 – 標準動作 UseBean

- 測試、執行
  - <http://localhost:8080/ContactInfo/index.jsp> 輸入地址資訊
  - 送出請求至 [setAddress.jsp](#)
    - 檢視設定訊息
    - Session加入此 Address Java Bean 物件
  - <http://localhost:8080/ContactInfo/showAddress.jsp> 檢視Session中 Address Java Bean 物件

# 建立 ContactInfo 專案



# index.jsp



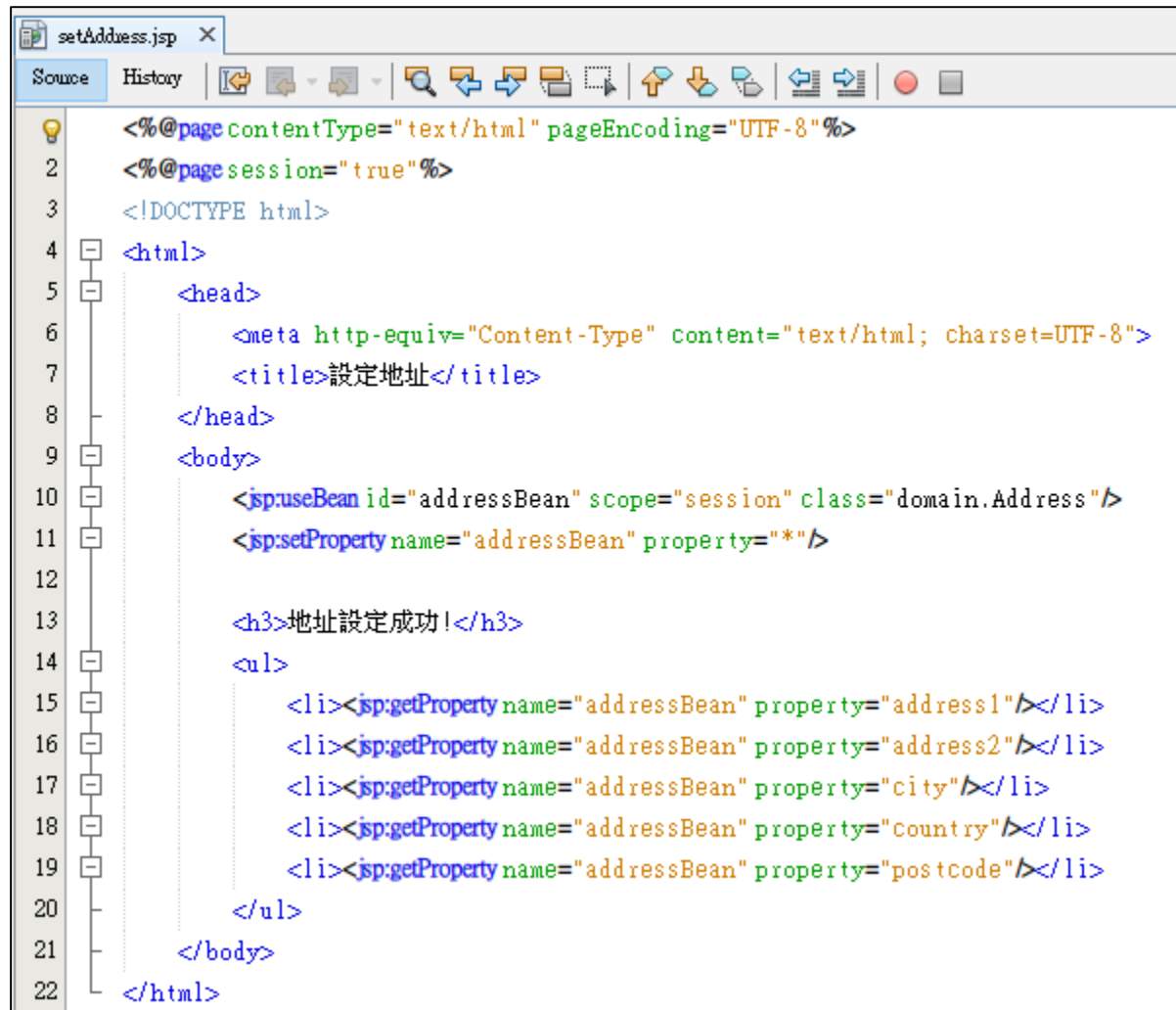
```
1 <%@page contentType="text/html" pageEncoding="UTF-8"%>
2 <%@page session="true"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7     <title>輸入地址</title>
8   </head>
9   <body>
10    <h3>請輸入地址:</h3>
11    <form action="setAddress.jsp">
12      地址一: <input type="text" name="address1"/><br><br>
13      地址二: <input type="text" name="address2"/><br><br>
14      城市: <input type="text" name="city"/><br><br>
15      國家: <input type="text" name="country"/><br><br>
16      郵遞區號: <input type="text" name="postcode"/><br><br>
17      <input type="submit" value="送出"/>
18    </form>
19  </body>
20 </html>
```

# domain.Address.java

```
Address.java
Source History
1 package domain;
2 import java.io.*;
3
4 public class Address implements Serializable{
5     private String address1;
6     private String address2;
7     private String city;
8     private String country;
9     private String postcode;
10
11     public Address() {
12     }
13
14
15     public Address(String address1, String address2,
16         String city, String country, String postcode) {
17         this.address1 = address1;
18         this.address2 = address2;
19         this.city = city;
20         this.country = country;
21         this.postcode = postcode;
22     }
23
24     public String getAddress1() {
25         return address1;
26     }
27
28     public void setAddress1(String address1) {
29         this.address1 = address1;
30     }
```

```
32     public String getAddress2() {
33         return address2;
34     }
35
36     public void setAddress2(String address2) {
37         this.address2 = address2;
38     }
39
40     public String getCity() {
41         return city;
42     }
43
44     public void setCity(String city) {
45         this.city = city;
46     }
47
48     public String getCountry() {
49         return country;
50     }
51
52     public void setCountry(String country){
53         this.country = country;
54     }
55
56     public String getPostcode() {
57         return postcode;
58     }
59
60     public void setPostcode(String postcode) {
61         this.postcode = postcode;
62     }
63
64 }
```

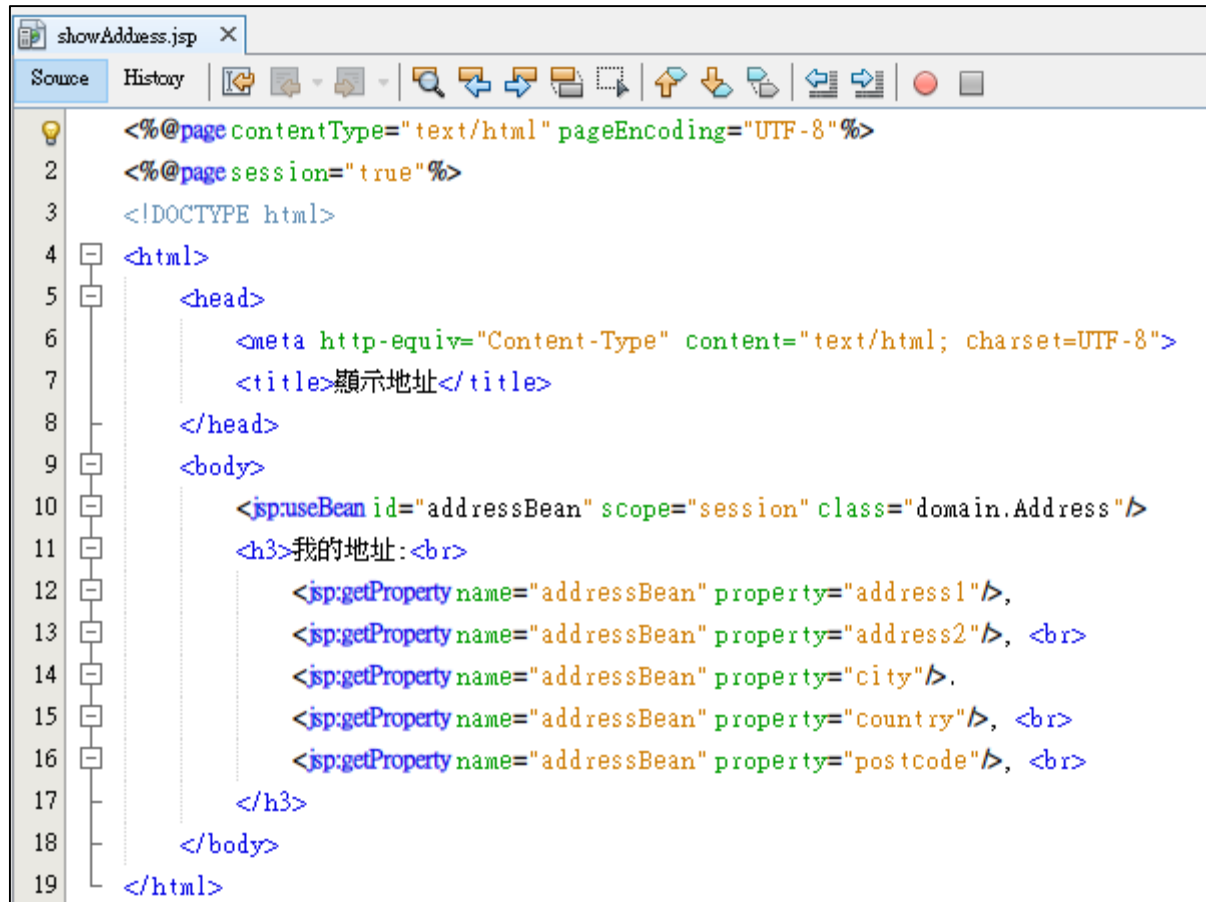
# setAddress.jsp



```
1  <%@page contentType="text/html" pageEncoding="UTF-8"%>
2  <%@page session="true"%>
3  <!DOCTYPE html>
4  <html>
5  <head>
6      <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7      <title>設定地址</title>
8  </head>
9  <body>
10     <jsp:useBean id="addressBean" scope="session" class="domain.Address"/>
11     <jsp:setProperty name="addressBean" property="*" />
12
13     <h3>地址設定成功!</h3>
14     <ul>
15         <li><jsp:getProperty name="addressBean" property="address1"/></li>
16         <li><jsp:getProperty name="addressBean" property="address2"/></li>
17         <li><jsp:getProperty name="addressBean" property="city"/></li>
18         <li><jsp:getProperty name="addressBean" property="country"/></li>
19         <li><jsp:getProperty name="addressBean" property="postcode"/></li>
20     </ul>
21 </body>
22 </html>
```



# showAddress.jsp



```
1 <%@page contentType="text/html" pageEncoding="UTF-8"%>
2 <%@page session="true"%>
3 <!DOCTYPE html>
4 <html>
5 <head>
6     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7     <title>顯示地址</title>
8 </head>
9 <body>
10     <jsp:useBean id="addressBean" scope="session" class="domain.Address"/>
11     <h3>我的地址:<br>
12         <jsp:getProperty name="addressBean" property="address1"/>,
13         <jsp:getProperty name="addressBean" property="address2"/>, <br>
14         <jsp:getProperty name="addressBean" property="city"/>.,
15         <jsp:getProperty name="addressBean" property="country"/>, <br>
16         <jsp:getProperty name="addressBean" property="postCode"/>, <br>
17     </h3>
18 </body>
19 </html>
```

# 測試、執行

輸入地址

localhost:8080/Cont...

請輸入地址:

地址一:

地址二:

城市:

國家:

郵遞區號:

設定地址

localhost:8080/Cont...

地址設定成功!

- 凱達格蘭大道一號
- 總統府廣場
- 台北市
- 台灣
- 100

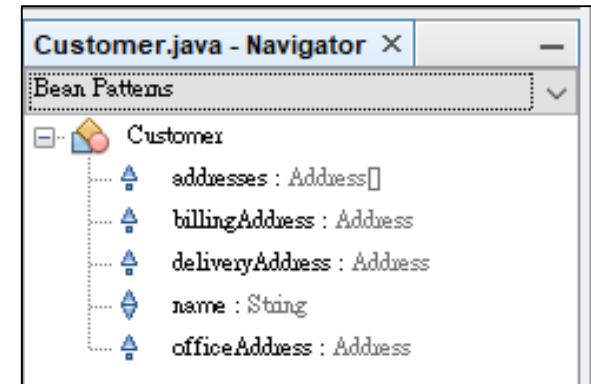
顯示地址

localhost:8080/ContactInfo/showAddress.jsp

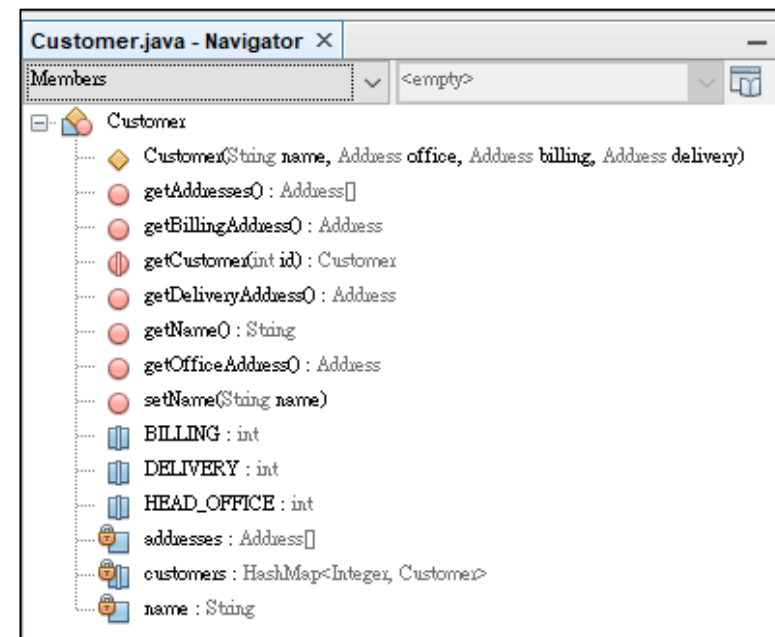
我的地址:  
凱達格蘭大道一號, 總統府廣場,  
台北市. 台灣,  
100,

# Lab 2

- 修改 ContactInfo 網路應用專案
- 檢視 domain.Address 模型元件
- 新增 domain.Customer.java 設計如右



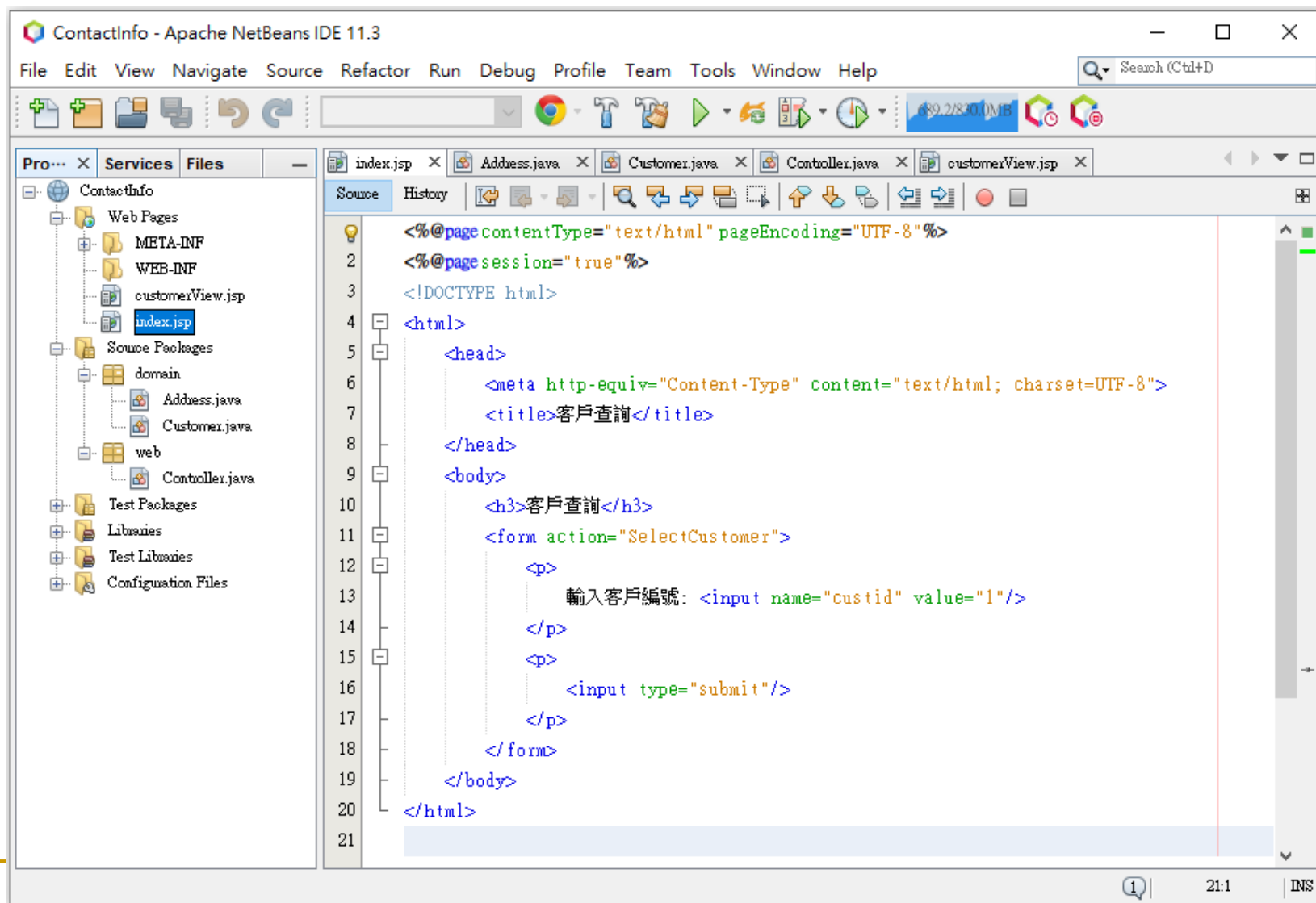
- 整數類別常數 HEAD\_OFFICE / BILLING / DELIVERY
- 字串屬性 name 及存取方法
- Address[ ]屬性 addresses 及 getter
- 建立靜態測試資料及方法
  - HashMap 類別屬性 customers
  - static block 初始化
  - Customer getCustomer(int id) 類別方法
- 修改 index.jsp
  - 呼叫 SelectCustomer 查詢客戶資料
  - 請求參數 custid



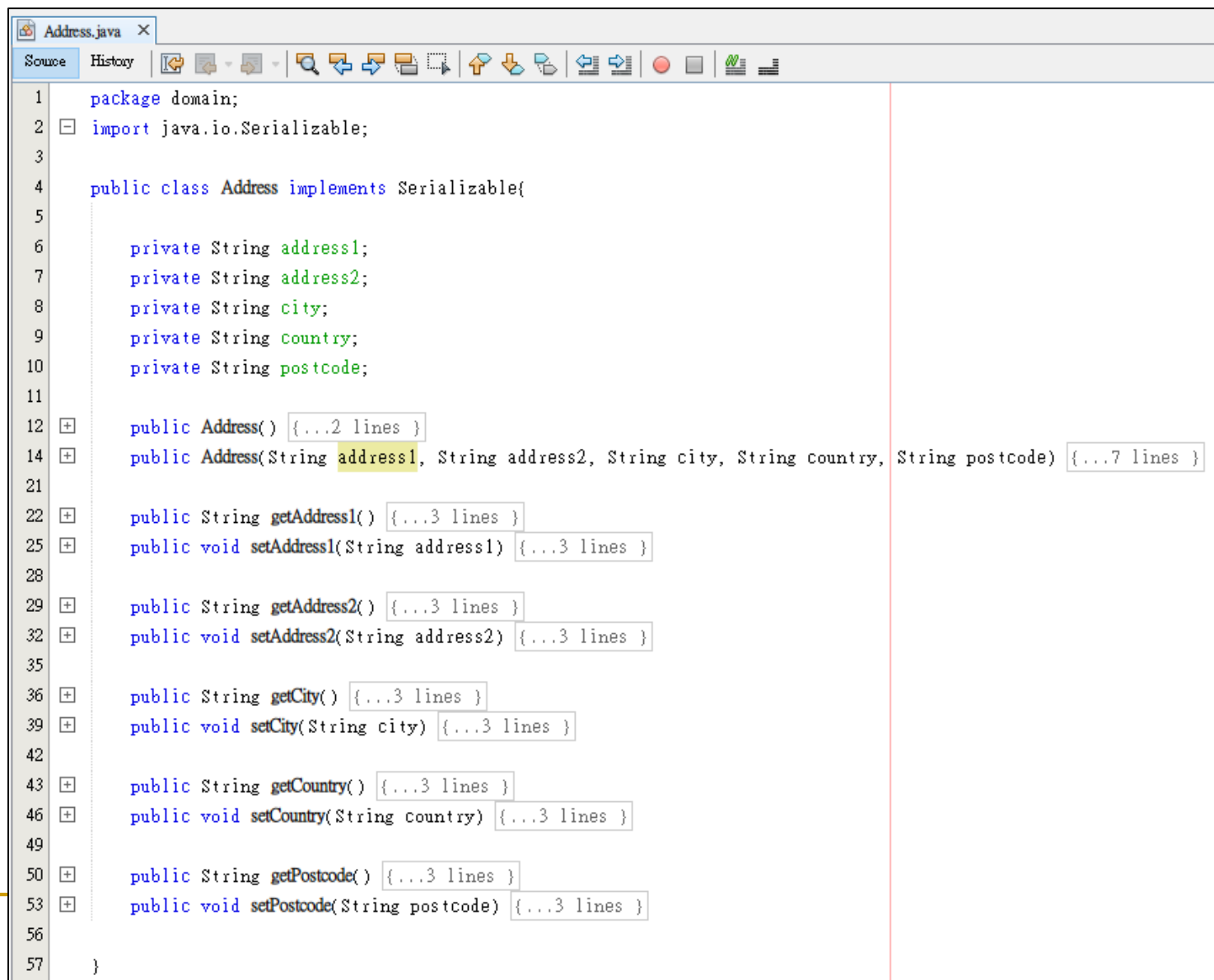
# Lab 2

- 撰寫 `web.Controller.java` 控制器元件
  - 取得請求參數 `custid`
  - `getCustomer()` 取得客戶物件
  - 轉送至 `customerView.jsp`
- 撰寫 `customerView.jsp`
  - 使用 `Expression Language(EL)`, 顯示客戶聯絡資料
  - 由請求中 `customer` 屬性取得客戶資料
  - `OfficeAddress`, `BillingAddress`, `DeliveryAddress` 屬性資料
  - `Addresses[0]`, `Addresses[1]`, `Addresses[2]` 資料
- 刪除 `setAddress.jsp` / `showAddress.jsp`
- 測試、執行

# ContactInfo 專案



# 檢視 domain.Address.java



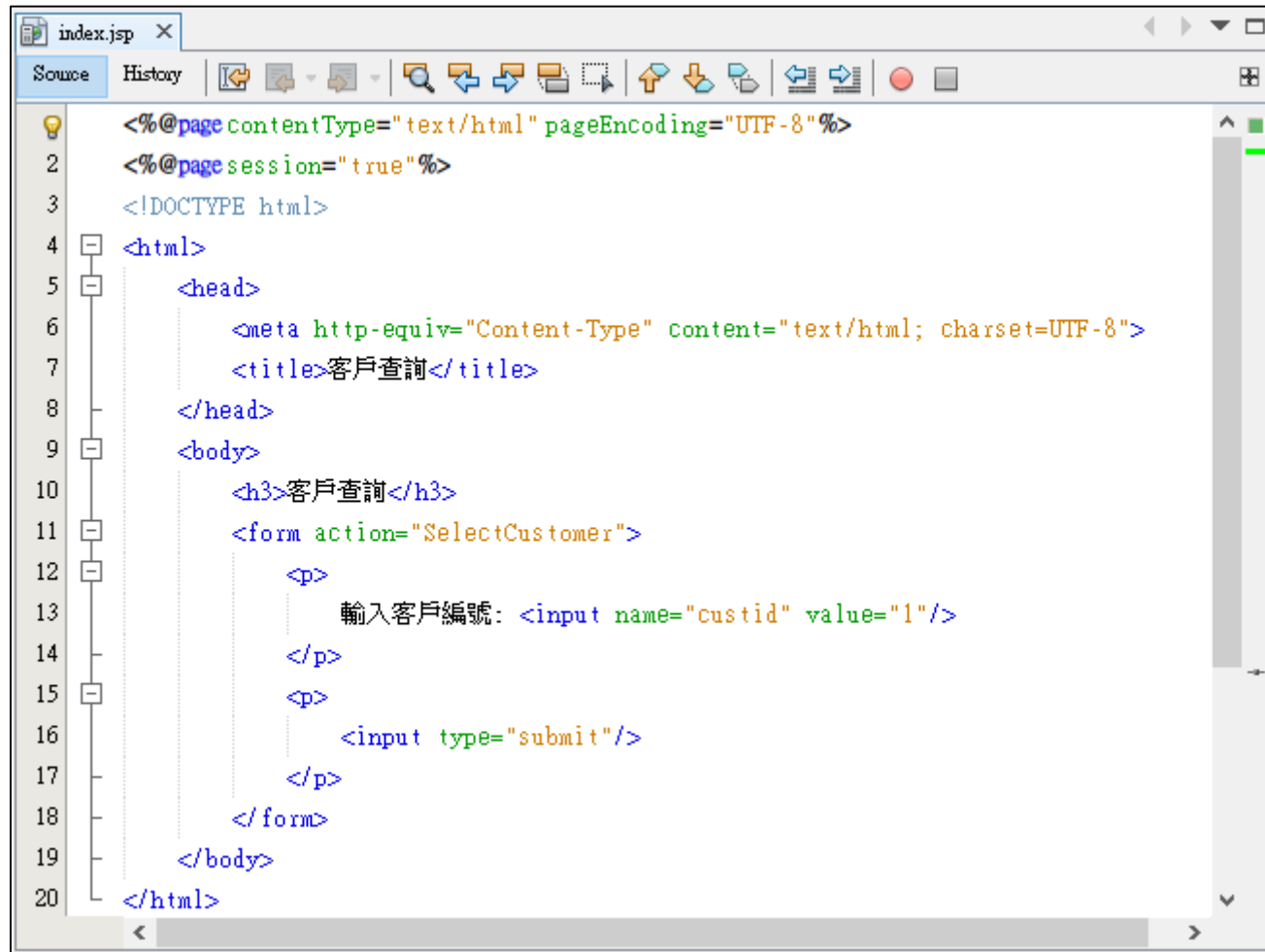
```
1 package domain;
2 import java.io.Serializable;
3
4 public class Address implements Serializable{
5
6     private String address1;
7     private String address2;
8     private String city;
9     private String country;
10    private String postcode;
11
12    public Address() {...2 lines }
13
14    public Address(String address1, String address2, String city, String country, String postcode) {...7 lines }
15
16
17
18
19
20
21
22    public String getAddress1() {...3 lines }
23
24    public void setAddress1(String address1) {...3 lines }
25
26
27
28
29    public String getAddress2() {...3 lines }
30
31    public void setAddress2(String address2) {...3 lines }
32
33
34
35
36    public String getCity() {...3 lines }
37
38    public void setCity(String city) {...3 lines }
39
40
41
42
43    public String getCountry() {...3 lines }
44
45    public void setCountry(String country) {...3 lines }
46
47
48
49
50    public String getPostcode() {...3 lines }
51
52    public void setPostcode(String postcode) {...3 lines }
53
54
55
56
57 }
```

# domain.Customer.java

```
Customer.java x
Source History
1 package domain;
2
3 import java.util.HashMap;
4
5 public class Customer {
6     private String name;
7     public static final int HEAD_OFFICE = 0;
8     public static final int BILLING = 1;
9     public static final int DELIVERY = 2;
10    private final Address [] addresses = new Address[3];
11
12    public Customer(String name, Address office, Address billing, Address delivery ){
13        this.name = name;
14        addresses[HEAD_OFFICE] = office;
15        addresses[BILLING] = billing;
16        addresses[DELIVERY] = delivery;
17    }
18
19    //測試資料
20    private static final HashMap<Integer, Customer> customers = new HashMap<Integer, Customer>();
21    static {
22        customers.put(1, new Customer("蔡英文",
23            new Address("凱達格蘭大道1號", "總統府廣場", "台北", "台灣", "100"),
24            new Address("重慶南路一段122號", "總統府", "台北", "台灣", "100"),
25            new Address("重慶南路二段100號", "總統官邸", "台北", "台灣", "100")));
26        customers.put(2, new Customer("柯文哲",
27            new Address("市府路1號", "台北市政府", "台北", "台灣", "110"),
28            new Address("仁愛路四段507號", "台北市議會", "台北", "台灣", "110"),
29            new Address("徐州路46號", "台北市長官邸", "台北", "台灣", "100")));
30    }
31
32    public static Customer getCustomer(int id) {
33        return customers.get(id);
34    }
```

```
35
36 public String getName() {
37     return name;
38 }
39
40 public void setName(String name){
41     this.name = name;
42 }
43
44 public Address getOfficeAddress() {
45     return addresses[HEAD_OFFICE];
46 }
47
48 public Address getBillingAddress() {
49     return addresses[BILLING];
50 }
51
52 public Address getDeliveryAddress() {
53     return addresses[DELIVERY];
54 }
55
56 public Address[] getAddresses() {
57     return addresses;
58 }
59
60 }
```

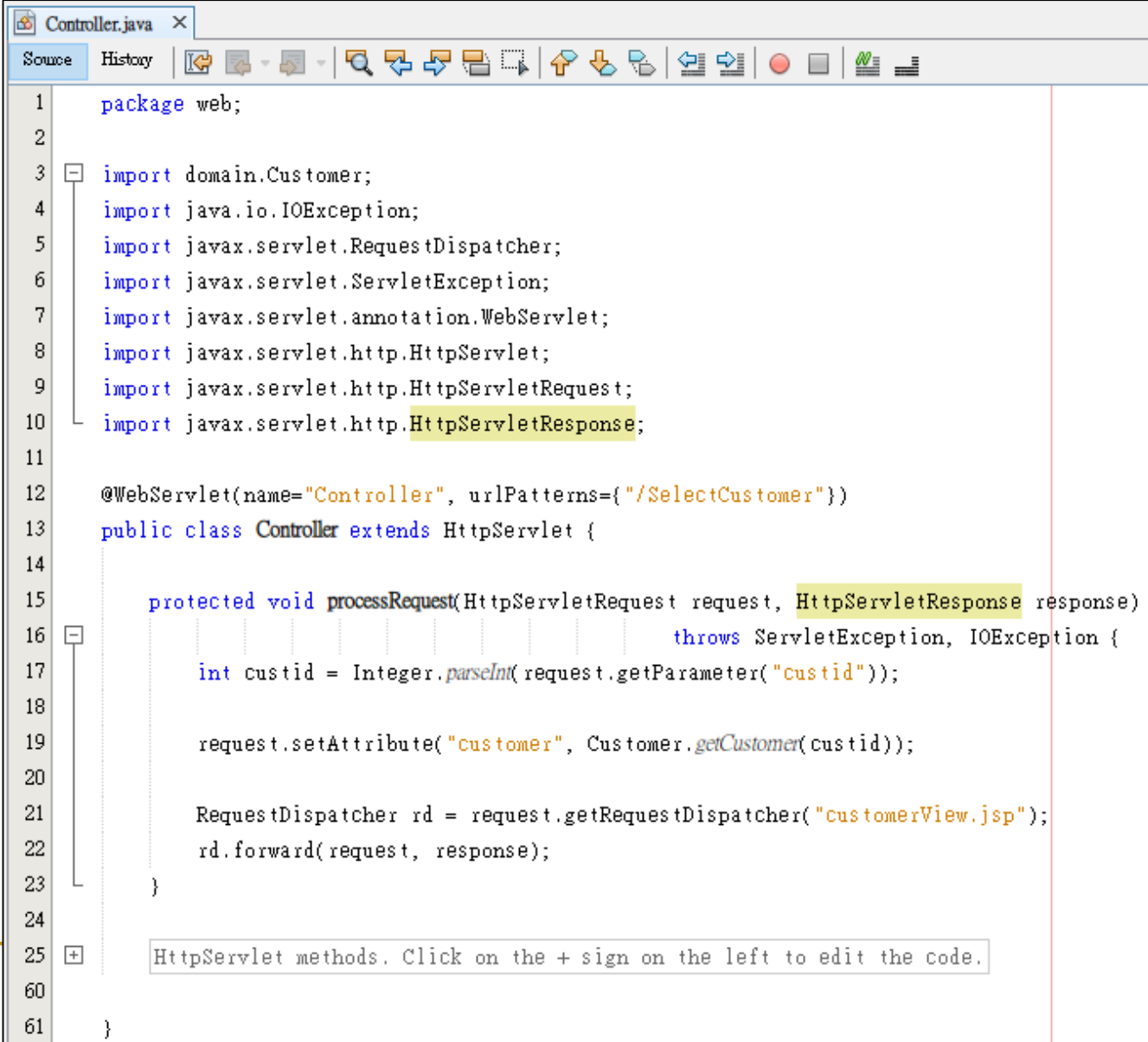
# index.jsp

A screenshot of a web development IDE showing the source code of a file named 'index.jsp'. The window has a title bar with 'index.jsp' and standard OS controls. Below the title bar is a menu bar with 'Source' and 'History'. A toolbar with various icons is located below the menu bar. The main editor area displays the JSP code, which is color-coded. On the left side of the editor, there is a vertical line with markers for each line of code, and a small tree view on the far left showing the document structure. The code is as follows:

```
1 <%@page contentType="text/html" pageEncoding="UTF-8"%>
2 <%@page session="true"%>
3 <!DOCTYPE html>
4 <html>
5   <head>
6     <meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
7     <title>客戶查詢</title>
8   </head>
9   <body>
10    <h3>客戶查詢</h3>
11    <form action="SelectCustomer">
12      <p>
13        輸入客戶編號: <input name="custid" value="1"/>
14      </p>
15      <p>
16        <input type="submit"/>
17      </p>
18    </form>
19  </body>
20 </html>
```



# web.Controller.java

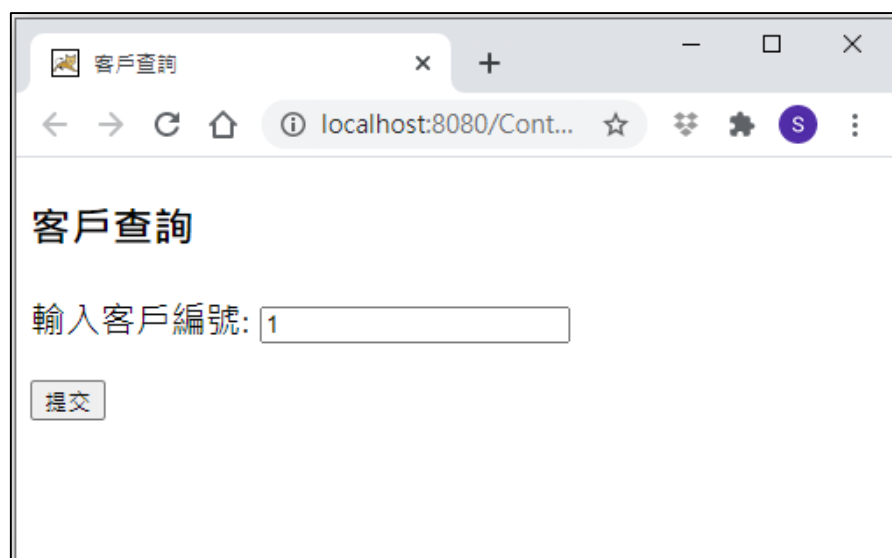


```
1 package web;
2
3 import domain.Customer;
4 import java.io.IOException;
5 import javax.servlet.RequestDispatcher;
6 import javax.servlet.ServletException;
7 import javax.servlet.annotation.WebServlet;
8 import javax.servlet.http.HttpServlet;
9 import javax.servlet.http.HttpServletRequest;
10 import javax.servlet.http.HttpServletResponse;
11
12 @WebServlet(name="Controller", urlPatterns={"/SelectCustomer"})
13 public class Controller extends HttpServlet {
14
15     protected void processRequest(HttpServletRequest request, HttpServletResponse response)
16         throws ServletException, IOException {
17         int custid = Integer.parseInt(request.getParameter("custid"));
18
19         request.setAttribute("customer", Customer.getCustomer(custid));
20
21         RequestDispatcher rd = request.getRequestDispatcher("customerView.jsp");
22         rd.forward(request, response);
23     }
24
25     HttpServlet methods. Click on the + sign on the left to edit the code.
26
27
28
29
30
31
32
33
34
35
36
37
38
39
40
41
42
43
44
45
46
47
48
49
50
51
52
53
54
55
56
57
58
59
60
61 }
```

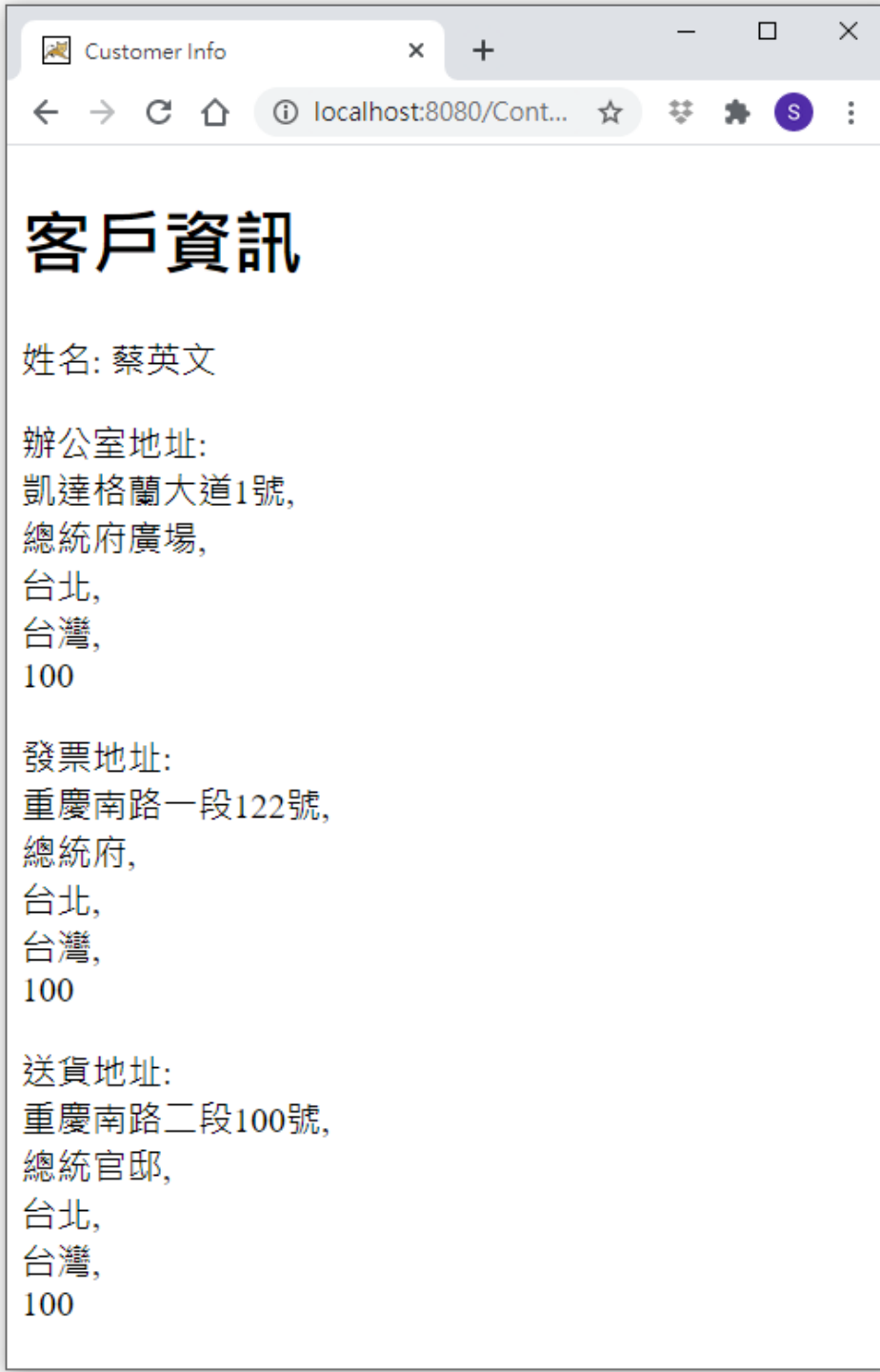
# CustomerView.jsp

```
customerView.jsp
Source History
<%@page contentType="text/html" pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">
<title>Customer Info</title>
</head>
<body>
<h1>客戶資訊</h1>
<p>姓名: ${customer.name}</p>
<p>辦公室地址:<br>
    ${customer.officeAddress.address1},<br>
    ${customer.officeAddress.address2},<br>
    ${customer.officeAddress.city},<br>
    ${customer.officeAddress.country},<br>
    ${customer.officeAddress.postcode}<br>
</p>
<p>發票地址:<br>
    ${customer["billingAddress"].address1},<br>
    ${customer["billingAddress"].address2},<br>
    ${customer["billingAddress"].city},<br>
    ${customer["billingAddress"].country},<br>
    ${customer["billingAddress"].postcode}<br>
</p>
<p>送貨地址:<br>
    ${customer.addresses[2].address1},<br>
    ${customer.addresses[2].address2},<br>
    ${customer.addresses[2].city},<br>
    ${customer["addresses"][2].country},<br>
    ${customer["addresses"][2].postcode}<br>
</p>
</body>
</html>
```

# 測試、執行



A screenshot of a web browser window titled "客戶查詢" (Customer Query). The browser's address bar shows "localhost:8080/Cont...". The page content includes the title "客戶查詢" and a form with the label "輸入客戶編號:" (Enter customer number:). The input field contains the number "1". Below the input field is a button labeled "提交" (Submit).



A screenshot of a web browser window titled "Customer Info". The browser's address bar shows "localhost:8080/Cont...". The page content includes the title "客戶資訊" (Customer Information) and the following details:

- 姓名: 蔡英文
- 辦公室地址:  
凱達格蘭大道1號,  
總統府廣場,  
台北,  
台灣,  
100
- 發票地址:  
重慶南路一段122號,  
總統府,  
台北,  
台灣,  
100
- 送貨地址:  
重慶南路二段100號,  
總統官邸,  
台北,  
台灣,  
100