



巨匠線上真人

Java Web OCE JWCD元件系統 開發認證

www.pcschoolonline.com.tw



巨匠線上真人

Java Web OCE JWCD元件系統開發認證

第八堂 JSP EL/JSTL 運算語言應用

本堂教學重點

- ◆ EL運算語言基本語法與運算子應用
- ◆ EL配合自訂函數應用
- ◆ JSTL-Java Standard Tag Library應用

何謂JSP EL(Expressions Language)

- ◆ JSP EL表達運算式，可以在特定語法上進行運算，或者邏輯比對等。甚至可以呼叫JSP封裝的Java Method，轉換的函數進行運算。
- ◆ EL一般具有輸出，可以輕鬆黏貼在HTML Tag進行運算與輸出。
- ◆ 具有內建物件，可以輕鬆運算不同範圍中的Attribute結果。
- ◆ 語法:
 - ◆ `${ EL表達式 }`
 - ◆ `${requestScope.userName}`
 - ◆ `${requestScope["user"]["userName"]}`
 - ◆ `${requestScope.member[0].["userName"]}`

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>簡易EL運算式</title>
</head>
<body>
(3X10)/2=${3*10/2}
</body>
</html>
```

← → ↺ ⓘ localhost:8080/mod08/simpleel.jsp

$(3 \times 10) / 2 = 15.0$

EL內建物件

◆ EL有11內建物件:

內建物件	描述
pageScope	參照出page層級參照的物件
requestScope	參照出ServletRequest層級參照的物件
sessionScope	參照出HttpSession層級參照的物件
applicationScope	照站出ServletContext應用系統層級參照的物件
param	參考出傳遞進來的參數名稱的內容
paramValues	參考出所有參數的內容(values)
header	取出HTTP Header的內容
headerValues	取出所有Header Collection的內容
initParam	參照出ServletContext初始化參數內容
cookie	參照出Cookie 值
pageContext	JSP PageContext物件為相對網頁物件

pageScope應用

- ◆ 在網頁層級中，透過pageContext進行setAttribute Method參照下來的物件。可以透過EL使用pageScope內建物件進行截取與輸出。
- ◆ EL執行發生錯誤時，具有例外管理，不會直接擲出錯誤。

pagescoedemo.jsp

```
<%  
    String com="巨匠電腦";  
    pageContext.setAttribute("company", com);  
%>  
  
<div>使用區域變數於EL: ${com}</div>  
<div>使用EL內建物件: ${pageScope.company}</div>
```

← → ↻ ⓘ localhost:8080/mod08/pagescoedemo.jsp

使用區域變數於EL:
使用EL內建物件:巨匠電腦

requestScope 應用

- ◆ requestScope可以取得ServletRequest.setAttribute()參照下來的物件。
- ◆ 一般用於RequestDispatcher或者是<jsp:forward>等派送到JSP處理UI時，需要在來源帶入狀態到JSP進行相關的處理，這時候JSP可以使用EL配合內建物件requestScope進行物件擷取。

sourcepage.jsp

```
<body>
  <%
    int rad=(int)(Math.random()*10000);
    //透過HttpServletRequest進行隨機值參考
    request.setAttribute("number",rad);
  %>
  <jsp:forward page="showvalue.jsp"></jsp:forward>
</body>
```

showvalue.jsp

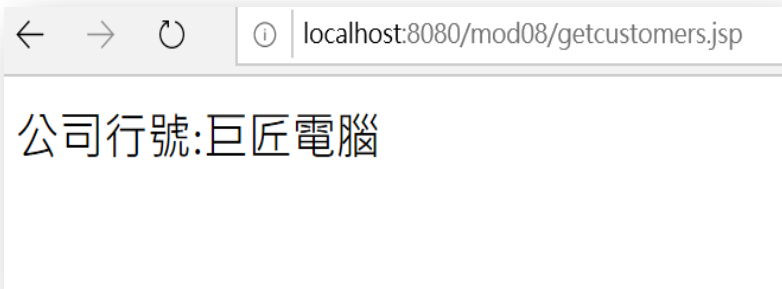
```
<body>
  <div>來源網頁的隨機值:${requestScope.number}</div>
</body>
```

← → ↻ ⓘ localhost:8080/mod08/sourcepage.jsp

來源網頁的隨機值:5098

sessionScope應用

- ◆ 在同一個網站系統，節合一個前端進行持續狀態管理用。如同使用 `HttpSession.setAttribute()` 進行物件參考，可以橫跨整個網站系統。
- ◆ 使用 `sessionScope` 將 `HttpSession` 範圍內的物件進行參照與運算。



sessionsource.jsp

```
<%  
    //建構一個客戶物件  
    Customers customers=new Customers();  
    customers.setCustomerId("0001");  
    customers.setCompanyName("巨匠電腦");  
    customers.setPhone("02-12345678");  
    session.setAttribute("customers",customers);  
%>  
  
<div>建構一個客戶物件!!</div>
```

getcustomers.jsp

```
<meta charset="UTF-8">  
<title>sessionScope應用</title>  
</head>  
<body>  
    公司行號: ${sessionScope.customers.companyName}  
</body>  
</html>
```


applicationScope應用

- ◆ 如同ServletContext.setAttribute() 參照物件之後，可以形成整個應用系統共用的資源。
- ◆ 可以透過EL applicationScope獲取ServletContext參照的物件。
- ◆ 如應用系統初始化設定的資源，我們可以使用applicationScope取出與列印。

ApplicationHandler.java

```
@WebListener
public class ApplicationHandler implements ServletContextListener {

    @Override
    public void contextDestroyed(ServletContextEvent sce) {
        // TODO Auto-generated method stub
        ServletContextListener.super.contextDestroyed(sce);
    }

    @Override
    public void contextInitialized(ServletContextEvent sce) {
        //聆聽應用系統初始化資訊
        sce.getServletContext().setAttribute("company", "巨匠電腦");
    }
}
```

applicationscopedemo.jsp

```
<body>
<div>公司行號:${applicationScope.company}</div>
</body>
```

EL配合自訂函數應用

- ◆ 撰寫具有**static Method**的類別，當作共用函數使用時，一般具有回應值。
- ◆ 可以在**JSP**使用**scriptlet**或者是**expressions**呼叫該函數。
- ◆ 可以透過**tld**檔的描述，並且配合**EL**進行函數呼叫。

設計具有**static**
Method類別

規劃**tld**檔進行**function**
描述

撰寫**JSP**引用**tld** 且使
用**EL**運算自訂函數

民國年轉換的static Method設計

- ◆ 設計一個類別具有static Method，獲取系統日期的民國年字串。

AppUtility.java

```
public class AppUtility {  
    //回應現在的民國日期  
    public static String chineseDate() {  
        Calendar calendar=Calendar.getInstance();  
        int year=calendar.get(Calendar.YEAR)-1911;  
        int month=calendar.get(Calendar.MONTH)+1;  
        String m=(month+"").length()==2?month+"":"0"+month;  
        int day=calendar.get(Calendar.DAY_OF_MONTH);  
        String d=(day+"").length()==2?day+"":"0"+day;  
        return year+"/"+m+"/"+d;  
    }  
}
```

規劃tld檔進行function描述

◆ Tld(Tag Library Descriptor)檔案建立

◆ <function>...</function>元素描述

appfunction.tld

```
<?xml version="1.0" encoding="UTF-8"?>
<taglib version="2.0" xmlns="http://java.sun.com/xml/ns/j2ee"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xsi:schemaLocation="http://java.sun.com/xml/ns/j2ee
    web-jsptagLibrary_2_0.xsd">
  <!-- 宣告的版本 -->
  <tlib-version>1.0</tlib-version>
  <!-- 精簡名稱 -->
  <short-name>RandomTags</short-name>
  <!--提供JSP引用的URI-->
  <uri>http://www.gjun.com.tw/functions</uri>

  <function>
    <description>民國年轉換</description>
    <name>chDate</name>
    <function-class>
      com.gjun.domain.AppUtility
    </function-class>
    <function-signature>
      java.lang.String chineseDate()
    </function-signature>
  </function>

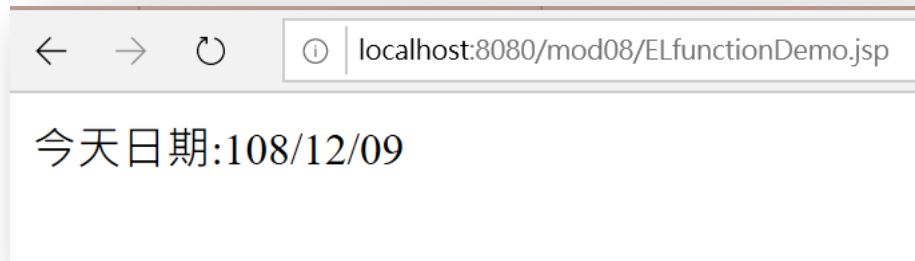
</taglib>
```

EL呼叫函數應用

ELfunctionDemo.jsp

- ◆ JSP使用`<%@taglib ...%>`先行引用tld定義的uri。
- ◆ EL呼叫函數`${xx:functionname(參數,...)}`


```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="fn" uri="http://www.gjun.com.tw/functions" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<div>
    今天日期: ${fn:chDate()}
</div>
</body>
</html>
```






JSTL-Java Standard Tag Library應用

- ◆ JSTL(Java Standard Tag Library)JSP標準標籤庫。
- ◆ JSTL提供了一個有效的途徑，以在JSP頁面中嵌入邏輯，而不是直接嵌入Java代碼。使用標準標籤庫，減少了Java代碼導致的不連續，從而提高JSP網頁的可維護性。
- ◆ 支援有核心庫(core)，格式化(format)，資料庫標籤(sql)與xml文件庫與函數標籤庫等。
- ◆ 下載處：
 - ◇ <https://tomcat.apache.org/taglibs/standard/>

JSP(tm) Standard Tag Library implementations

Apache hosts the Apache Standard Taglib, an implementation of the [JSP Standard Tag Library \(JSTL\)](#)  specification. Various versions are available.

Version	JSTL version	Requirements	Getting the Taglib
Standard 1.2.3	JSTL 1.2	Servlet 2.5, JavaServer Pages 2.1	download  (javadoc)
Standard 1.1	JSTL 1.1	Servlet 2.4, JavaServer Pages 2.0	download 
Standard 1.0	JSTL 1.0	Servlet 2.3, JavaServer Pages 1.2	download 

使用JSTL Core核心標籤

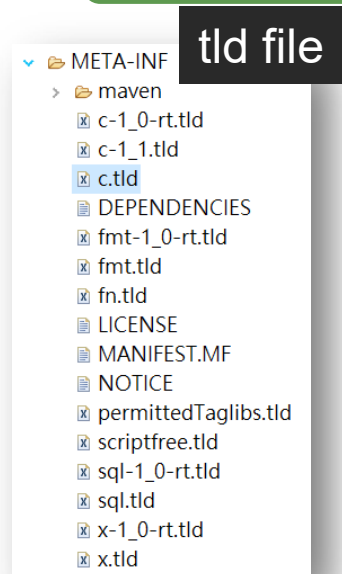
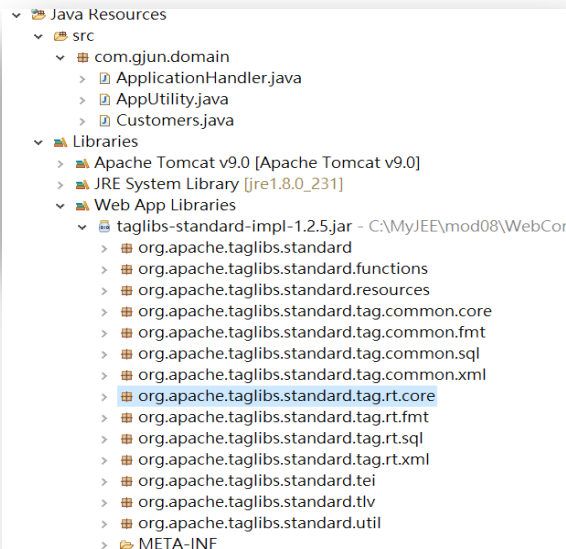
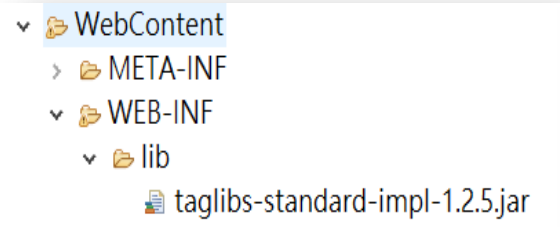
放置jstl jar檔案到網站
WEB-INF/lib資料夾中



JSP網頁使用
<%@taglib...%>進行引用
JSTL標籤庫



使用XML語法使用JSTL標
籤



JSTL Core核心標籤

Tags	Description
<code>c:out</code>	It display the result of an expression, similar to the way <code><%=...%></code> tag work.
<code>c:import</code>	It Retrives relative or an absolute URL and display the contents to either a String in 'var', a Reader in 'varReader' or the page.
<code>c:set</code>	It sets the result of an expression under evaluation in a 'scope' variable.
<code>c:remove</code>	It is used for removing the specified scoped variable from a particular scope.
<code>c:catch</code>	It is used for Catches any Throwable exceptions that occurs in the body.
<code>c:if</code>	It is conditional tag used for testing the condition and display the body content only if the expression evaluates is true.
<code>c:choose, c:when, c:otherwise</code>	It is the simple conditional tag that includes its body content if the evaluated condition is true.
<code>c:forEach</code>	It is the basic iteration tag. It repeats the nested body content for fixed number of times or over collection.
<code>c:forTokens</code>	It iterates over tokens which is separated by the supplied delimiters.
<code>c:param</code>	It adds a parameter in a containing 'import' tag's URL.
<code>c:redirect</code>	It redirects the browser to a new URL and supports the context-relative URLs.
<code>c:url</code>	It creates a URL with optional query parameters.

JSP引用JSTL uri

- ◆ `<%@taglib prefix="c" uri=""%>`
- ◆ 使用`<c:tag>`進行核心標籤應用。

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>Insert title here</title>
</head>
<body>
<!-- 設定attribute內容 -->
<c:set var="com" value="巨匠電腦"></c:set>
<!-- EL輸出 -->
${com}
<br/>
<!-- JSTL輸出 -->
<c:out value="${com}"></c:out>

</body>
</html>
```

格式化標籤庫

- ◆ 用來格式化日期，訊息與區域等資料。

Formatting Tags	Descriptions
<code>fmt:parseNumber</code>	It is used to Parses the string representation of a currency, percentage or number.
<code>fmt:timeZone</code>	It specifies a parsing action nested in its body or the time zone for any time formatting.
<code>fmt:formatNumber</code>	It is used to format the numerical value with specific format or precision.
<code>fmt:parseDate</code>	It parses the string representation of a time and date.
<code>fmt:bundle</code>	It is used for creating the ResourceBundle objects which will be used by their tag body.
<code>fmt:setTimeZone</code>	It stores the time zone inside a time zone configuration variable.
<code>fmt:setBundle</code>	It loads the resource bundle and stores it in a bundle configuration variable or the named scoped variable.
<code>fmt:message</code>	It display an internationalized message.
<code>fmt:formatDate</code>	It formats the time and/or date using the supplied pattern and styles.

格式化一個貨幣數值應用

formatecurrency.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@taglib prefix="fmt" uri="http://java.sun.com/jsp/jstl/fmt" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="UTF-8">
<title>數值貨幣格式化</title>
</head>
<body>
<fmt:setLocale value="en_US"/>
<p>美元en_US: <fmt:formatNumber value="123.456" type="currency"/></p>
<fmt:setLocale value="ja_JP"/>
<p>日幣ja_JP: <fmt:formatNumber value="123.456" type="currency"/></p>
<fmt:setLocale value="de_DE"/>
<p>德國de_DE: <fmt:formatNumber value="123.456" type="currency"/></p>
</body>
</html>
```

← → ↺ ⓘ localhost:8080/mod08/formatcurrency.jsp

美元en_US: \$123.46

日幣ja_JP: ¥ 123

德國de_DE: 123,46 €

SQL標籤庫應用

- ◆ JSTL的SQL標記庫提供標簽為關聯式資料庫（RDBMS），如Oracle，MySQL等，或Microsoft SQL Server進行交互。

SQL Tags	Descriptions
<code>sql:setDataSource</code>	It is used for creating a simple data source suitable only for prototyping.
<code>sql:query</code>	It is used for executing the SQL query defined in its sql attribute or the body.
<code>sql:update</code>	It is used for executing the SQL update defined in its sql attribute or in the tag body.
<code>sql:param</code>	It is used for sets the parameter in an SQL statement to the specified value.
<code>sql:dateParam</code>	It is used for sets the parameter in an SQL statement to a specified java.util.Date value.
<code>sql:transaction</code>	It is used to provide the nested database action with a common connection.

SQL標籤庫連接MySQL進行資料查詢

sqldemo.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8" pageEncoding="UTF-8"%>
<%@ page import="java.io.*,java.util.*,java.sql.*"%>
<%@ page import="javax.servlet.http.*,javax.servlet.*" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/sql" prefix="sql"%>
```

```
<html>
<head>
<title>JSTL sql:query Tag</title>
</head>
<body>
```

```
<sql:setDataSource var="db" driver="com.mysql.jdbc.Driver"
    url="jdbc:mysql://localhost:3306/sakila?useSSL=false&serverTimezone=UTC"
    user="root" password="1111"/>
```

```
<sql:query dataSource="${db}" var="result">
SELECT * from Customer;
</sql:query>
```

```
<table border="1" width="100%">
<tr>
<th>客戶編號</th>
<th>First Name</th>
<th>Last Name</th>
<th>Email</th>
</tr>
<c:forEach var="row" items="${result.rows}">
<tr>
<td><c:out value="${row.customer_id}"/></td>
<td><c:out value="${row.first_name}"/></td>
<td><c:out value="${row.last_name}"/></td>
<td><c:out value="${row.email}"/></td>
</tr>
</c:forEach>
</table>
```

資料查詢結果

客戶編號	First Name	Last Name	Email
1	MARY	SMITH	MARY.SMITH@sakilacustomer.org
2	PATRICIA	JOHNSON	PATRICIA.JOHNSON@sakilacustomer.org
3	LINDA	WILLIAMS	LINDA.WILLIAMS@sakilacustomer.org
4	BARBARA	JONES	BARBARA.JONES@sakilacustomer.org
5	ELIZABETH	BROWN	ELIZABETH.BROWN@sakilacustomer.org
6	JENNIFER	DAVIS	JENNIFER.DAVIS@sakilacustomer.org
7	MARIA	MILLER	MARIA.MILLER@sakilacustomer.org
8	SUSAN	WILSON	SUSAN.WILSON@sakilacustomer.org
9	MARGARET	MOORE	MARGARET.MOORE@sakilacustomer.org
10	DOROTHY	TAYLOR	DOROTHY.TAYLOR@sakilacustomer.org
11	LISA	ANDERSON	LISA.ANDERSON@sakilacustomer.org
12	NANCY	THOMAS	NANCY.THOMAS@sakilacustomer.org
13	KAREN	JACKSON	KAREN.JACKSON@sakilacustomer.org
14	BETTY	WHITE	BETTY.WHITE@sakilacustomer.org
15	HELEN	HARRIS	HELEN.HARRIS@sakilacustomer.org
16	SANDRA	MARTIN	SANDRA.MARTIN@sakilacustomer.org
17	DONNA	THOMPSON	DONNA.THOMPSON@sakilacustomer.org
18	CAROL	GARCIA	CAROL.GARCIA@sakilacustomer.org
19	RUTH	MARTINEZ	RUTH.MARTINEZ@sakilacustomer.org
20	SHARON	ROBINSON	SHARON.ROBINSON@sakilacustomer.org
21	MICHELLE	CLARK	MICHELLE.CLARK@sakilacustomer.org
22	LAURA	RODRIGUEZ	LAURA.RODRIGUEZ@sakilacustomer.org
23	SARAH	LEWIS	SARAH.LEWIS@sakilacustomer.org
24	KIMBERLY	LEE	KIMBERLY.LEE@sakilacustomer.org
25	DEBORAH	WALKER	DEBORAH.WALKER@sakilacustomer.org

XML標籤庫

- ◆ 提供XML剖析(Parser)、流程控制(Controller)、轉換(Convert)等功能之標籤。

XML Tags	Descriptions
x:out	Similar to <%= ... > tag, but for XPath expressions.
x:parse	It is used for parse the XML data specified either in the tag body or an attribute.
x:set	It is used to sets a variable to the value of an XPath expression.
x:choose	It is a conditional tag that establish a context for mutually exclusive conditional operations.
x:when	It is a subtag of that will include its body if the condition evaluated be 'true'.
x:otherwise	It is subtag of that follows tags and runs only if all the prior conditions evaluated be 'false'.
x:if	It is used for evaluating the test XPath expression and if it is true, it will processes its body content.
x:transform	It is used in a XML document for providing the XSL(Extensible Stylesheet Language) transformation.
x:param	It is used along with the transform tag for setting the parameter in the XSLT style sheet.

定義一個段XML文件輸出

parserxml.jsp

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib prefix="c" uri="http://java.sun.com/jsp/jstl/core" %>
<%@ taglib prefix="x" uri="http://java.sun.com/jsp/jstl/xml" %>
<html>
<head>
<title>XML Tag範例</title>
</head>
<body>
<h2>XML Tag範例</h2>
<c:set var="mysite">
    <mysite>
        <name>XML Tag範例</name>
        <example>這是一個JSP網頁</example>
    </mysite>
</c:set>
<x:parse xml="${mysite}" var="output"/>
<b>我的站台 :: </b>
<x:out select="$output/mysite[1]/name" /><br>
<b>範例 ::</b>
<x:out select="$output/mysite[1]/example" />
</body>
</html>
```



localhost:8080/mod08/parserxml.jsp

XML Tag範例

我的站台 :: XML Tag範例
範例 :: 這是一個JSP網頁

函數庫應用

- ◆ 只提供一個針對字串常用的函數，配合EL應用。

JSTL Functions	Description
<u>fn:contains()</u>	It is used to test if an input string containing the specified substring in a program.
<u>fn:containsIgnoreCase()</u>	It is used to test if an input string contains the specified substring as a case insensitive way.
<u>fn:endsWith()</u>	It is used to test if an input string ends with the specified suffix.
<u>fn:escapeXml()</u>	It escapes the characters that would be interpreted as XML markup.
<u>fn:indexOf()</u>	It returns an index within a string of first occurrence of a specified substring.
<u>fn:trim()</u>	It removes the blank spaces from both the ends of a string.
<u>fn:startsWith()</u>	It is used for checking whether the given string is started with a particular string value.
<u>fn:split()</u>	It splits the string into an array of substrings.
<u>fn:toLowerCase()</u>	It converts all the characters of a string to lower case.
<u>fn:toUpperCase()</u>	It converts all the characters of a string to upper case.
<u>fn:substring()</u>	It returns the subset of a string according to the given start and end position.
<u>fn:substringAfter()</u>	It returns the subset of string after a specific substring.
<u>fn:substringBefore()</u>	It returns the subset of string before a specific substring.
<u>fn:length()</u>	It returns the number of characters inside a string, or the number of items in a collection.
<u>fn:replace()</u>	It replaces all the occurrence of a string with another string sequence.

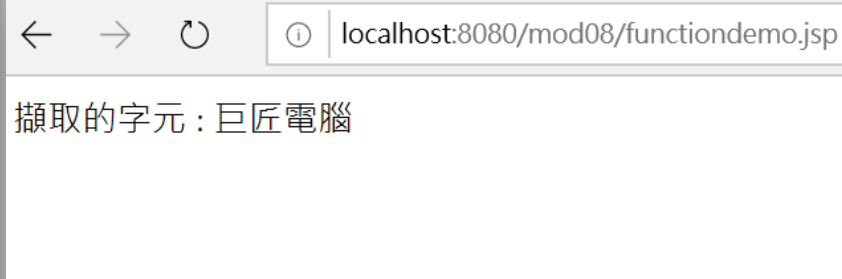
Function字串操作應用範例

```
<%@ page language="java" contentType="text/html; charset=UTF-8"
    pageEncoding="UTF-8"%>
<%@ taglib uri="http://java.sun.com/jsp/jstl/core" prefix="c" %>
<%@ taglib uri="http://java.sun.com/jsp/jstl/functions" prefix="fn" %>
<html>
<head>
<title>使用 JSTL 函数</title>
</head>
<body>

<c:set var="string1" value="巨匠電腦 資訊教育專業."/>
<c:set var="string2" value="{fn:substring(string1, 0, 4)}" />

<p>擷取的字元: ${string2}</p>

</body>
</html>
```



Lab

- ◆ 如何透過EL結合表單頁面輸入處理之後，進行結果輸出。
- ◆ EL內建物件牽涉到狀態管理的有那些，與其應用方式為何
- ◆ 如何撰寫一個字定函數，可以轉換民國年進行輸出
- ◆ JSTL如何設定Tag Library Descriptor
- ◆ 如何在頁面中引用JSTL進行資料查詢