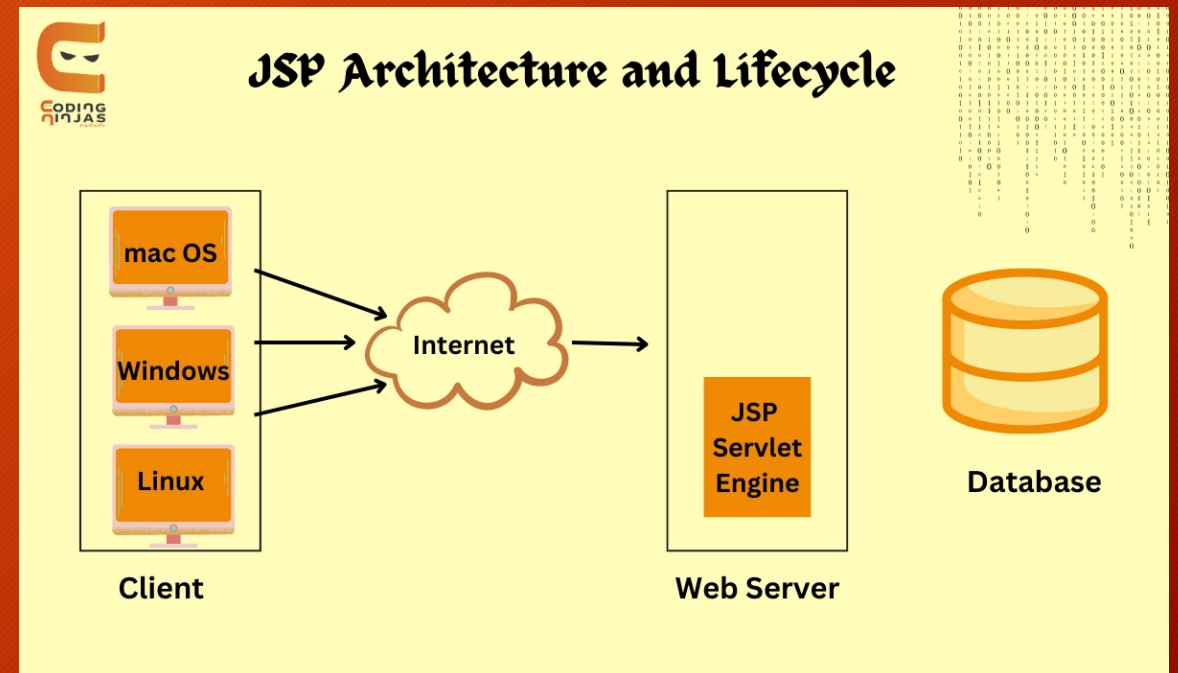


# Java EE Application 1z0-900

講師:江聖榮

# Jsp(全稱Jakarta Server Pages，曾稱為JavaServer Pages)

- 本質為servlet
- 在網頁架構下撰寫java語言
- 檔案類型為\*.jsp
- 基本指令語法
  - Scriptlet語法:撰寫java步驟就是main
    - `<%.....%>`
  - Expression與法:就是指定->=
    - `<%=變數,方法,運算式%>`
  - Declaration:只宣告用->變數，方法
    - `<%!....%>`
- 搭配隱含物件





# Jsp 隱含物件

- Jsp就是 servlet
- 在網頁上寫java
- 事先將 servlet上常用的物件,先new好
- 共9個,以下為常用的物件
- out->PrintWriter物件
- request->HttpServletRequest
- response->HttpServletResponse
- session->HttpSession

物件名稱	衍生類別	功能說明
request	Javax.servlet.ServletException. HttpServletRequest	取得客戶端資料與系統的資訊。
response	Javax.servlet.ServletException. HttpServletResponse	回應客戶端訊息。
application	Javax.servlet.ServletContext	記錄與處理上線者共用的資料。
session	Javax.servlet.http.HttpSession	記錄與處理上線者個別的資料。
out	Javax.servlet.jsp.JspWriter	控制資料輸出的動作。
config	Javax.servlet.ServletConfig	取得JSP編譯後Servlet的資訊。
pageContext	Javax.servlet.jsp.PageContext	存取與處理系統執行時期的各項資訊。
page	Java.lang.Object	代表目前的這個JSP網頁物件。
exception	Java.lang.Throwable	例外處理機制。

# Jsp Page標籤屬性(指令)

- Web server啟動後預先載入的設定值
- 語法:
  - `<%@page 屬性....%>`

```
1 <%@ page
2 language="java"
3 contentType="text/html; charset=BIG5"
4 pageEncoding="BIG5"
5 %>
6 <!DOCTYPE html>
7 <html>
8 <head>
9 <meta charset="BIG5">
10 <title>Insert title here</title>
11 </head>
12 <body>
13
14 </body>
15 </html>
```

## page 指令

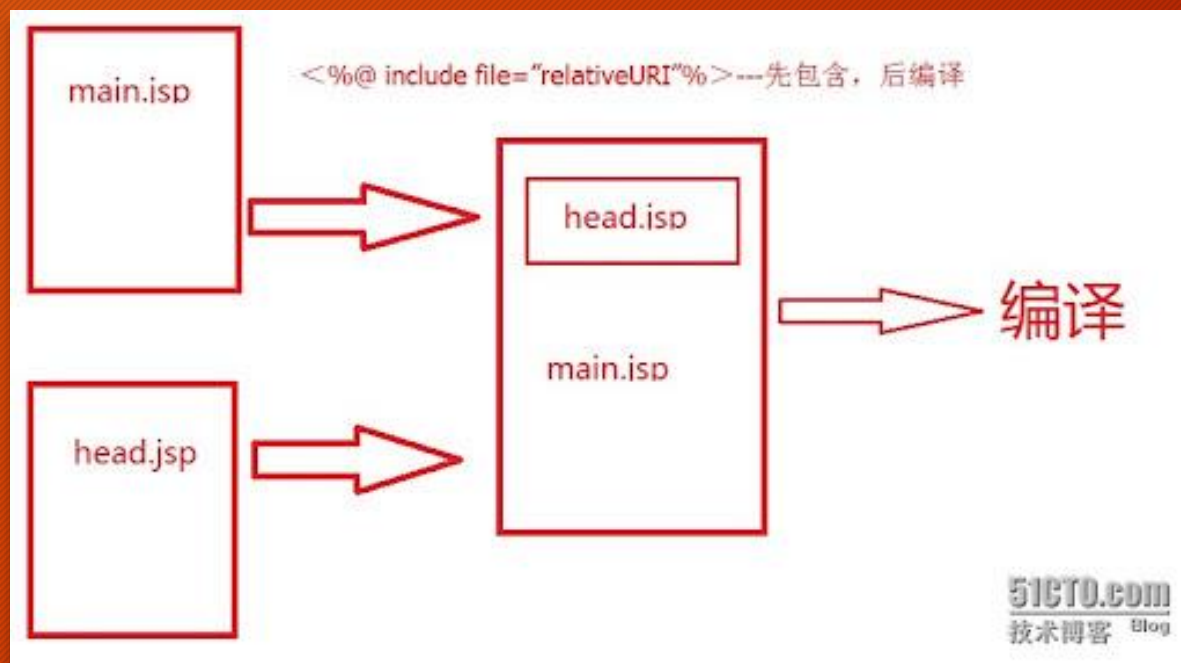
### ◆page 指令标记的属性

language	定义要使用的脚本语言。该属性的值目前只能取“java”，如果在将来 JSP 容器支持多种语言时可使用它。	“java”
import	该属性的作用是为JSP页面引入Java核心包中的类，可以为该属性指定多个值，值以逗号分隔，就像在通常的 Java 代码中所使用的import 语句那样。	无
contentType	定义JSP的字符编码方式和JSP页面响应的 MIME 类型。	CHARSET =iso-8859-1
isErrorPage	表示当前页面能否作为错误页面	true或false
errorPage	定义处理异常的 JSP 页面	errors_url
buffer	设置用来存储客户端请求的缓冲区的大小	“8KB”
session	用于设置是否需要使用内置的session对象	“true”



# Jsp include標籤屬性(指令)

- 將其他頁面內容嵌入在目標頁面上
- 語法:
- `<%@include file=""%>`
- 與來源頁面存在連結關係



# taglib標記指令,主要搭配 JSTL標籤

- 語法:
- `<%@taglib uri=""%>`
- JSTL(java standar tag library)
- 寫在`<body></body>`
- 步驟:
  - 將jar檔放入→lib
  - 撰寫`<%@taglib%>`
  - 撰寫需要的標籤
- Core版本
- `<c:out value="">`
- `<c:forEach var="" items="" />`
- ....
- `</c:forEach>`



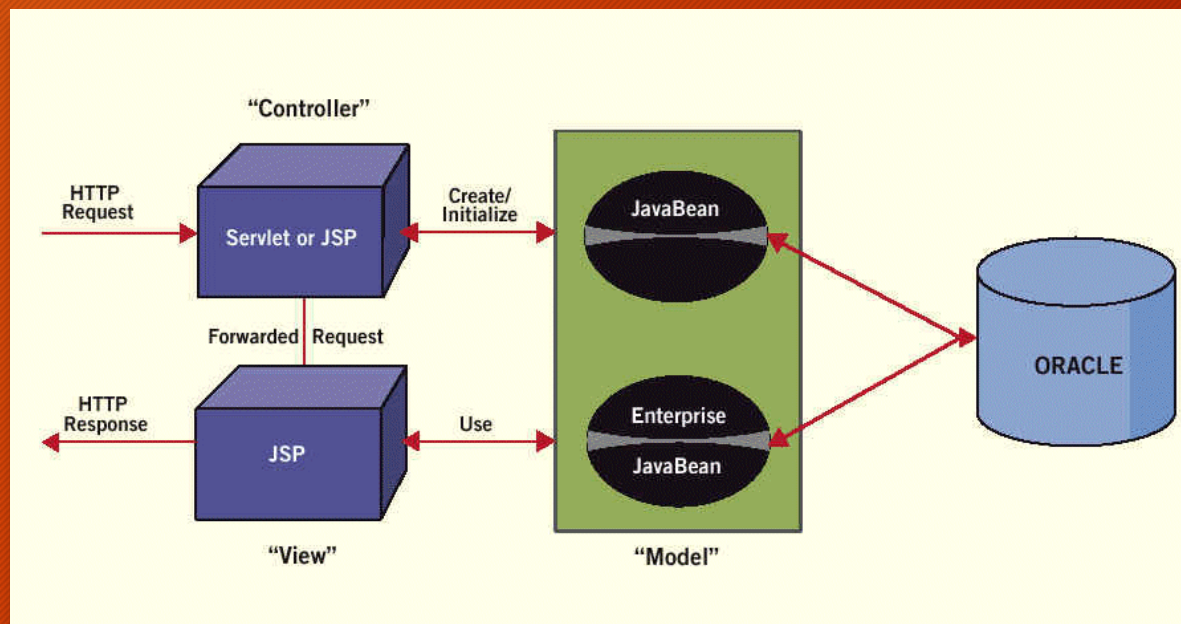
# EL(Expression Language)運算式

- 語法
- `${變數,計算式}`
- 變數需定義成
  - request
  - session
  - javaBean

Operator	Description	Example	EL Result
<code>== (eq)</code>	Equals	<code>\${ 7 == 7 }</code>	True
<code>!= (ne)</code>	Not Equals	<code>\${ 7 ne 7 }</code>	False
<code>&lt; (lt)</code>	Less than	<code>\${ 6 lt 7 }</code>	True
<code>&gt;(gt)</code>	Greater than	<code>\${ 6 &gt; 7 }</code>	False
<code>&lt;= (le)</code>	Less than or equal to	<code>\${ 6 &lt;= 7 }</code>	True
<code>&gt;= (ge)</code>	Greater than or equal to	<code>\${ 6 ge 7 }</code>	False
<code>and (&amp;&amp;)</code>	Logical AND Operation	<code>\${ 7==7 and 6==6 }</code>	True
<code>Not (!)</code>	Logical NOT Operation	<code>\${ not 6==6 }</code>	False
<code>Or (    )</code>	Logical OR Operation	<code>\${ 7==7    6==6 }</code>	True

# Jsp標籤與 javaBean物件,POJO物件

- 語法:
  - <jsp:include..>
- POJO(Plain old Java object)
- javaBean
- 無建構式的物件 class
- <jsp:useBean >--→new的動作
- <jsp:setProperty>→setter
- <jsp:getProperty>--→getter





# Design pattern

- 1990年代從建築設計領域引入到電腦科學的
- 開發專案參考的SOP流程
- 物件導向,MVC架構
- Dao(Data Access Object) pattern
  - interface定義需求功能的抽象方法
  - 子類別將抽象方法實作出來
- model
  - POJO的class
  - 參考資料庫的table
- dao(CRUD)
  - interface
  - Impl
    - 實作的子類別
- controller
  - 使用者操作用的方法