Java EE7 Web
使用Context共用應用程式資源

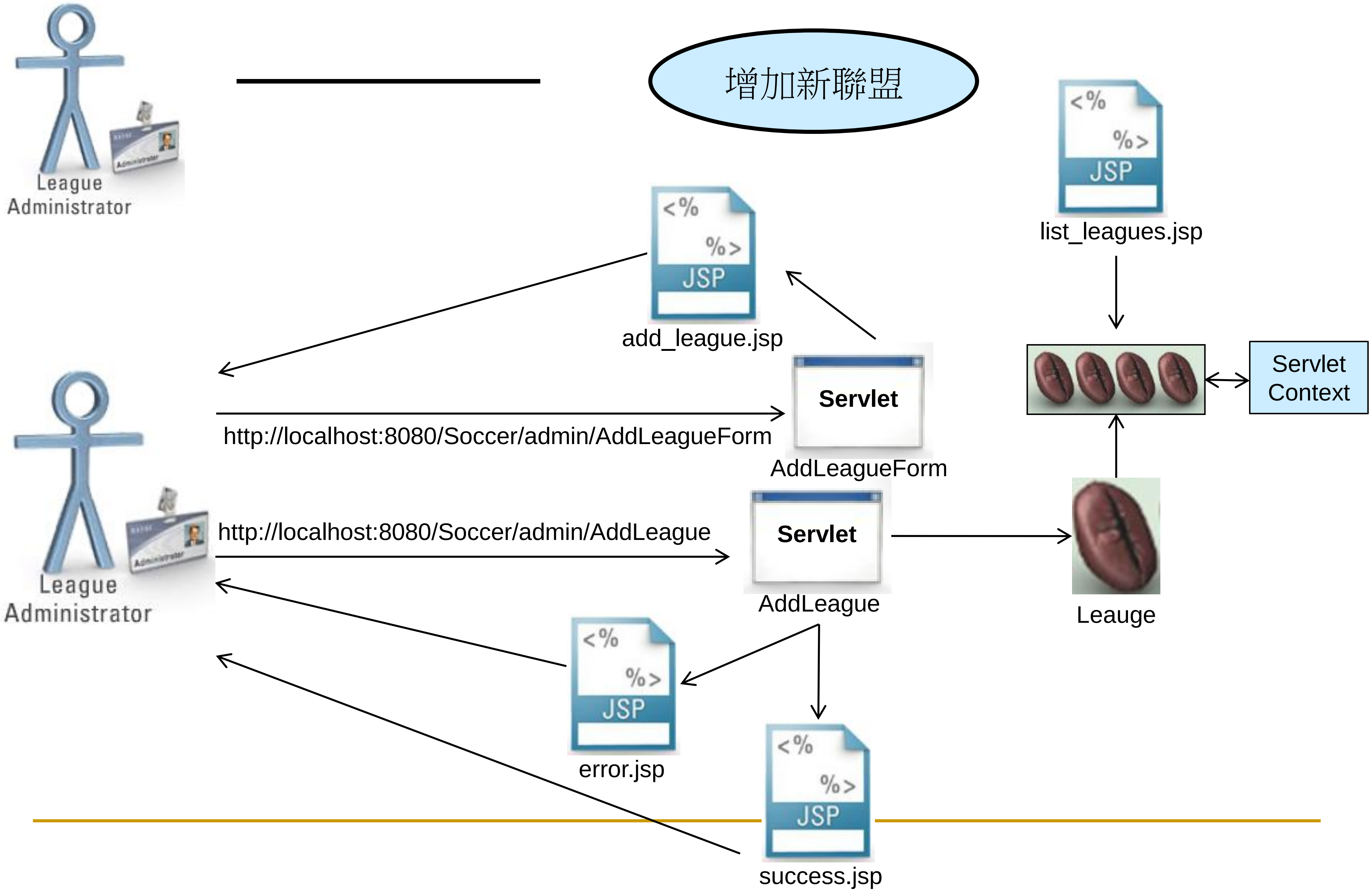
鄭安翔

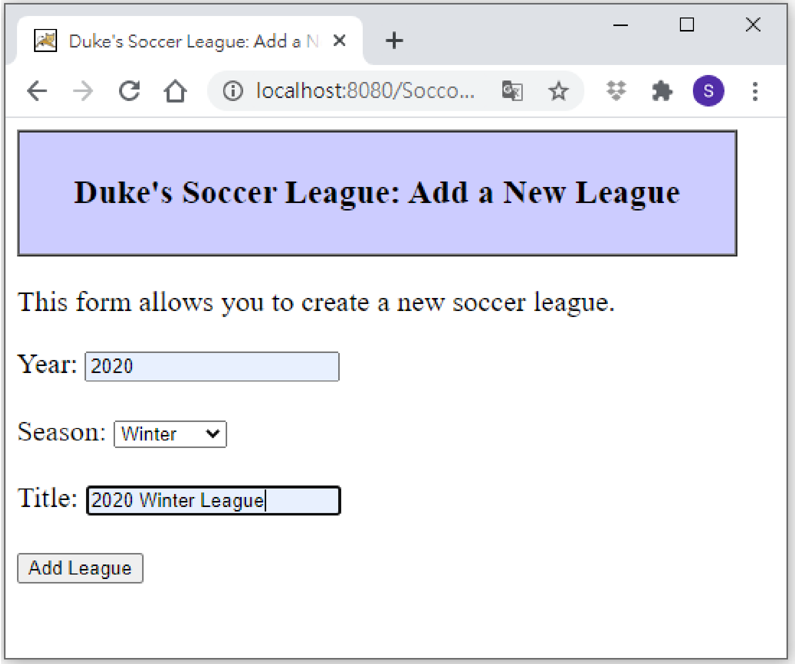
ansel_cheng@hotmail.com

課程大綱

1) **應用程式資源在所有元件中共享**
   - **ServletContext 屬性**
   - **JSP中使用共享資源**
2) 初始化應用程式資源

Soccer 範例


League Administrator
增加新聯盟
list_leagues.jsp
add_league.jsp
Servlet
http://localhost:8080/Soccer/admin/AddLeagueForm
AddLeagueForm
Servlet Context
League Administrator
http://localhost:8080/Soccer/admin/AddLeague
Servlet
AddLeague
Leauge
error.jsp
success.jsp
JSP



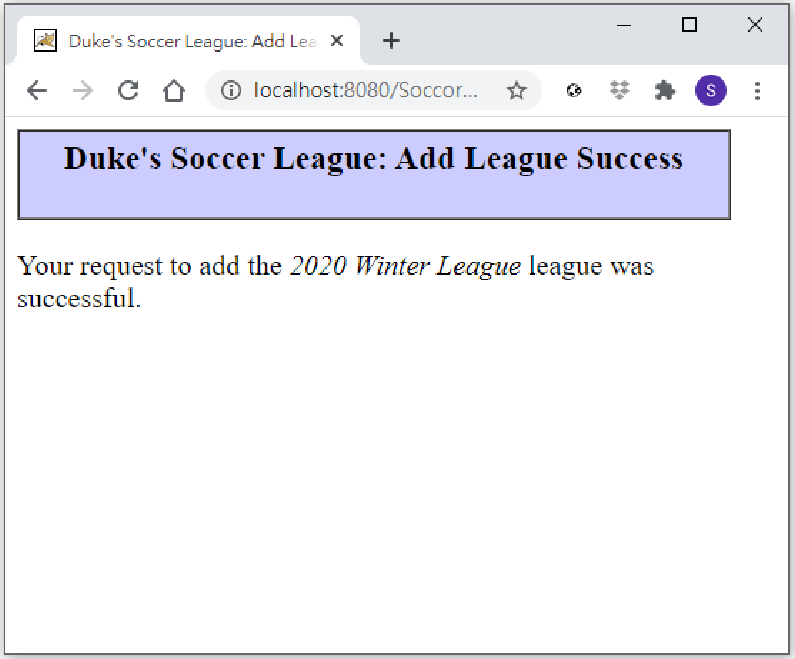
Duke's Soccer League: Add a New League

This form allows you to create a new soccer league.

Year: 2020

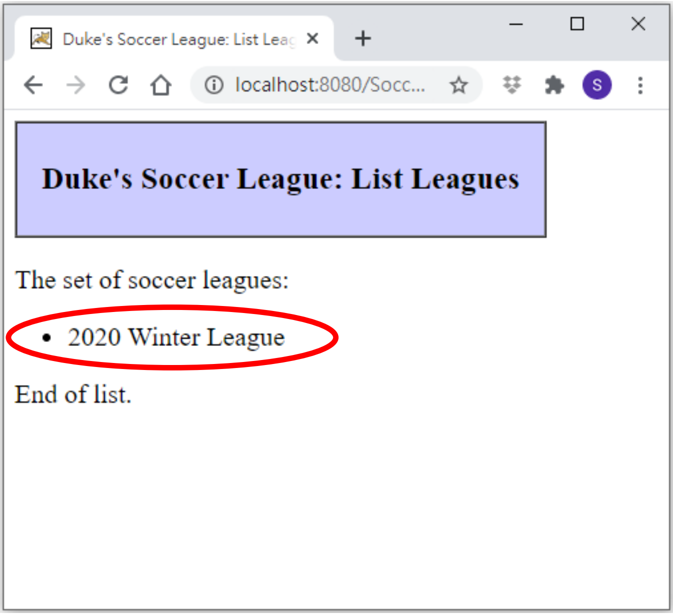
Season: Winter

Title: 2020 Winter League



Duke's Soccer League: Add League Success

Your request to add the *2020 Winter League* league was successful.



Duke's Soccer League: List Leagues

The set of soccer leagues:

- 2020 Winter League

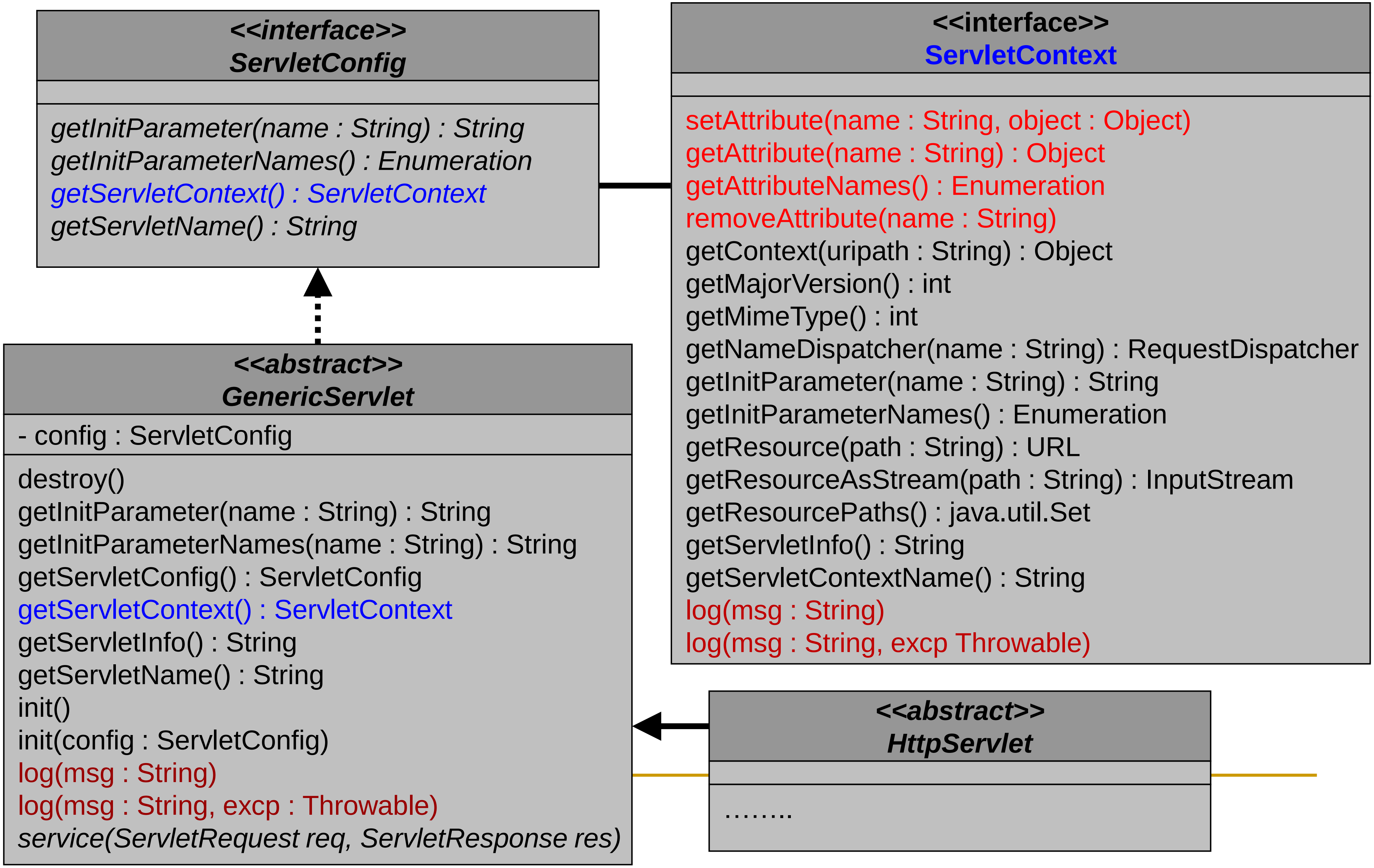
End of list.

使用servlet context 物件來儲存所有servlet共享的聯盟清單

javax.servlet.ServletContext

- 用來代表Web Application網路應用程式執行環境的物件
- 主要功能
  - 存取網路應用程式中所有Servlet共享的資源
    - Context Scope Attributes
    - 屬性資料為物件型態
  - 設定網路應用程式初始參數
  - 讀取網路應用程式中所有Servlet共享的檔案
    - URL
    - InputStream
  - 寫出訊息至日誌檔(Log Files)

javax.servlet.ServletContext



<<interface>>
ServletConfig

getInitParameter(name : String) : String
getInitParameterNames() : Enumeration
getServletContext() : ServletContext
getServletName() : String

<<interface>>
ServletContext

setAttribute(name : String, object : Object)
getAttribute(name : String) : Object
getAttributeNames() : Enumeration
removeAttribute(name : String)
getContext(uripath : String) : Object
getMajorVersion() : int
getMimeType() : int
getNameDispatcher(name : String) : RequestDispatcher
getInitParameter(name : String) : String
getInitParameterNames() : Enumeration
getResource(path : String) : URL
getResourceAsStream(path : String) : InputStream
getResourcePaths() : java.util.Set
getServletInfo() : String
getServletContextName() : String
log(msg : String)
log(msg : String, excp Throwable)

<<abstract>>
GenericServlet

- config : ServletConfig

destroy()
getInitParameter(name : String) : String
getInitParameterNames(name : String) : String
getServletConfig() : ServletConfig
getServletContext() : ServletContext
getServletInfo() : String
getServletName() : String
init()
init(config : ServletConfig)
log(msg : String)
log(msg : String, excp : Throwable)
service(ServletRequest req, ServletResponse res)

<<abstract>>
HttpServlet

……..

JSP 中使用應用程式共享資源

- JSP Expression / Scriptlet Tag 中
  - 使用隱含變數Implicit Variables

變數名稱	型別	用途	Scope
application	javax.servlet. ServletContext	網路應用程式的組態訊息 • 應用程式共享屬性 • 應用程式初始參數	application

```
<%= application.getAttribute("attributeName" ) %>
<%
    Type obj = (Type) application.getAttribute("attributeName");
    …..
 %>
```

JSP 中使用應用程式共享資源

- 使用表示式語言 (EL)
  - 未指定變數範圍時,自動尋找變數
  - 自動轉變資料型態

    ${*attributeName*}

    ${applicationScope.*attributeName*}

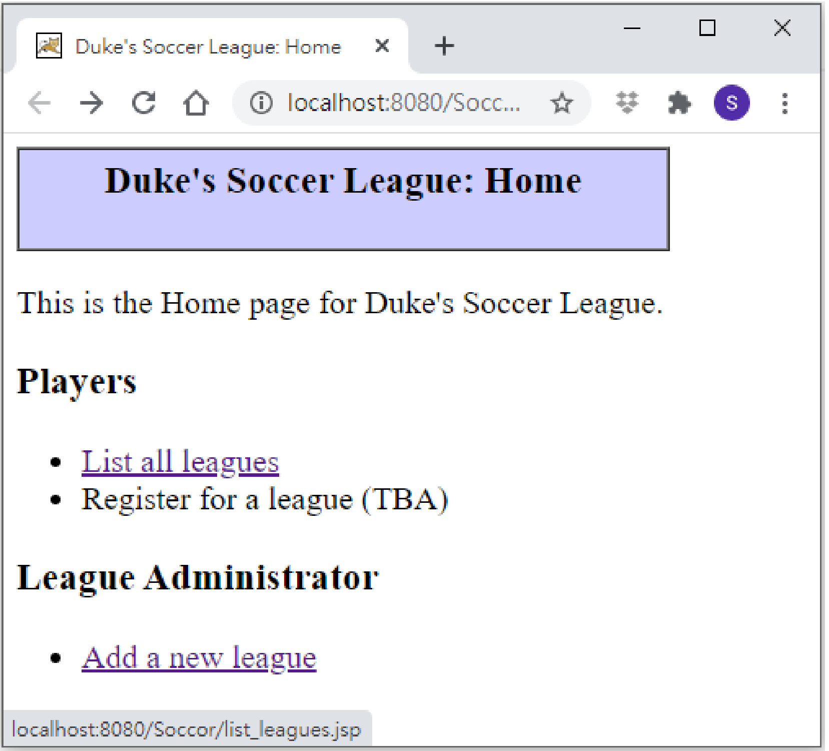
Soccer 範例 - 修改 AddLeague.java

```
package controller;
……
@WebServlet(name = “AddLeague", urlPatterns = {"/admin/AddLeague"})
public class AddLeague extends HttpServlet {
   protected void processRequest(HttpServletRequest request, HttpServletResponse
                                              response) throws ServletException, IOException {
      // Keep a set of strings to record form processing errors.
      // Store this set in the request scope, in case we need to send the ErrorPage.
      try {
          // Retrieve form parameters and Verify form parameters
          // Send the ErrorPage view if there were errors  or Perform business logic
          ……
          League league = new League(year, season, title);
          // Store the new league in the leagueList context-scope attribute
           ServletContext context = getServletContext();
           List leagueList = (List) context.getAttribute("leagueList");
           if(leagueList==null) {
               leagueList = new LinkedList();
               context.setAttribute("leagueList“, leagueList);
           }
           leagueList.add(league);
          // Store the new league in the request-scope
          // Send the Success view
          …….
      } catch (RuntimeException e) {
          …….
      }
   }
}
```

Soccer 範例

- 移除 ListLeagues.java
- 修改index.html

```
<html>
   <head>
      <title>Duke's Soccer League: Home</title>
   </head>
   <body>
      <table border='1' cellpadding='5' cellspacing='0' width='400'>
         <tr bgcolor='#CCCCFF' align='center'
              valign='center' height='20'>
            <td><h3>Duke's Soccer League: Home</h3></td>
         </tr>
      </table>
      <p>This is the Home page for Duke's Soccer League. </p>
      <h3>Players</h3>
      <ul>
         <li><a href='list_leagues.jsp'>List all leagues</a></li>
         <li>Register for a league (TBA)</li>
      </ul>
      <h3>League Administrator</h3>
      <ul>
         <li><a href='admin/AddLeagueForm'>Add a new league</a></li>
      </ul>
   </body>
</html>
```



Duke's Soccer League: Home

This is the Home page for Duke's Soccer League.

Players

- List all leagues
- Register for a league (TBA)

League Administrator

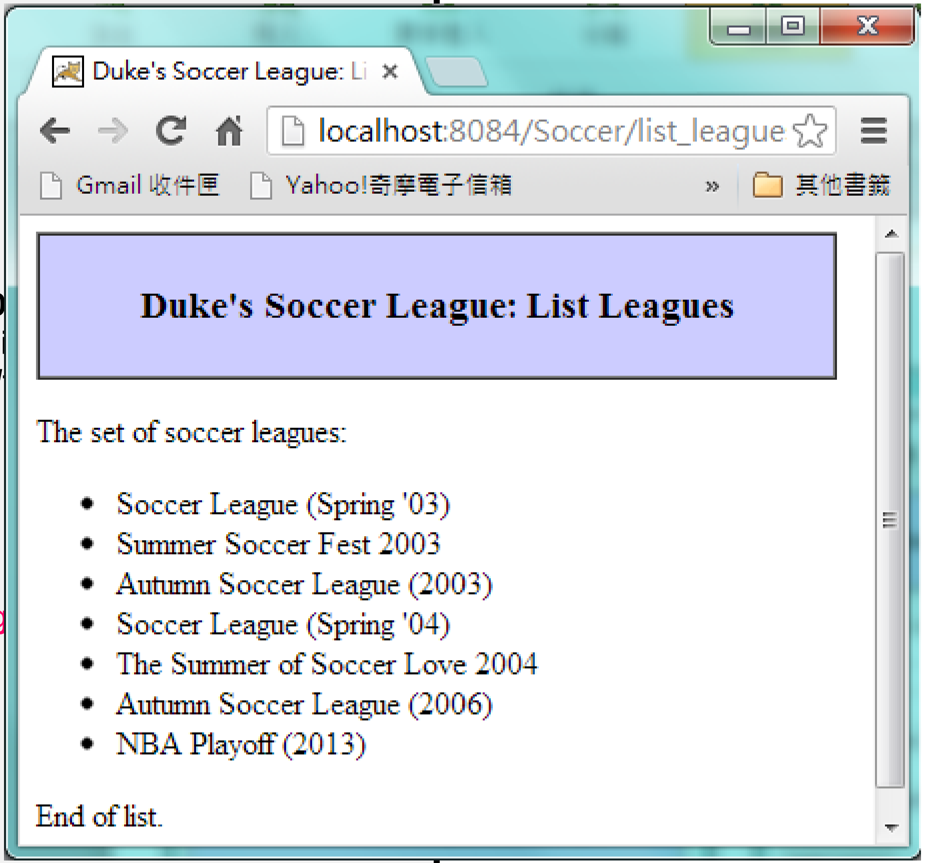
- Add a new league

localhost:8080/Soccor/list_leagues.jsp

Soccer 範例 - 修改list_league.jsp

```
<!DOCTYPE html>
<%@ page contentType="text/html"%>
<%@ page import="model.*" %>
<%@ page import="java.util.*" %>
<%@ page import=“model.*" %>

<html>
   <head>
      <title>Duke's Soccer League: List Leagues</title>
   </head>
   <body bgcolor='white'>
   <!-- Page Heading -->
   <table border='1' cellpadding='5' cellspacing='0' width='40
      <tr bgcolor='#CCCCFF' align='center' valign='center' hei
         <td><h3>Duke's Soccer League: List Leagues</h3></
      </tr>
   </table>
   <p>The set of soccer leagues:</p>
   <ul>
   <%
      List <League> leagueList = (List<League>) application.g
      for(League league : leagueList) {
   %>
   <li><%= league.getTitle() %></li>
   <%
      }
   %>
   </ul>
   <p>End of list.</p>
   </body>
</html>
```



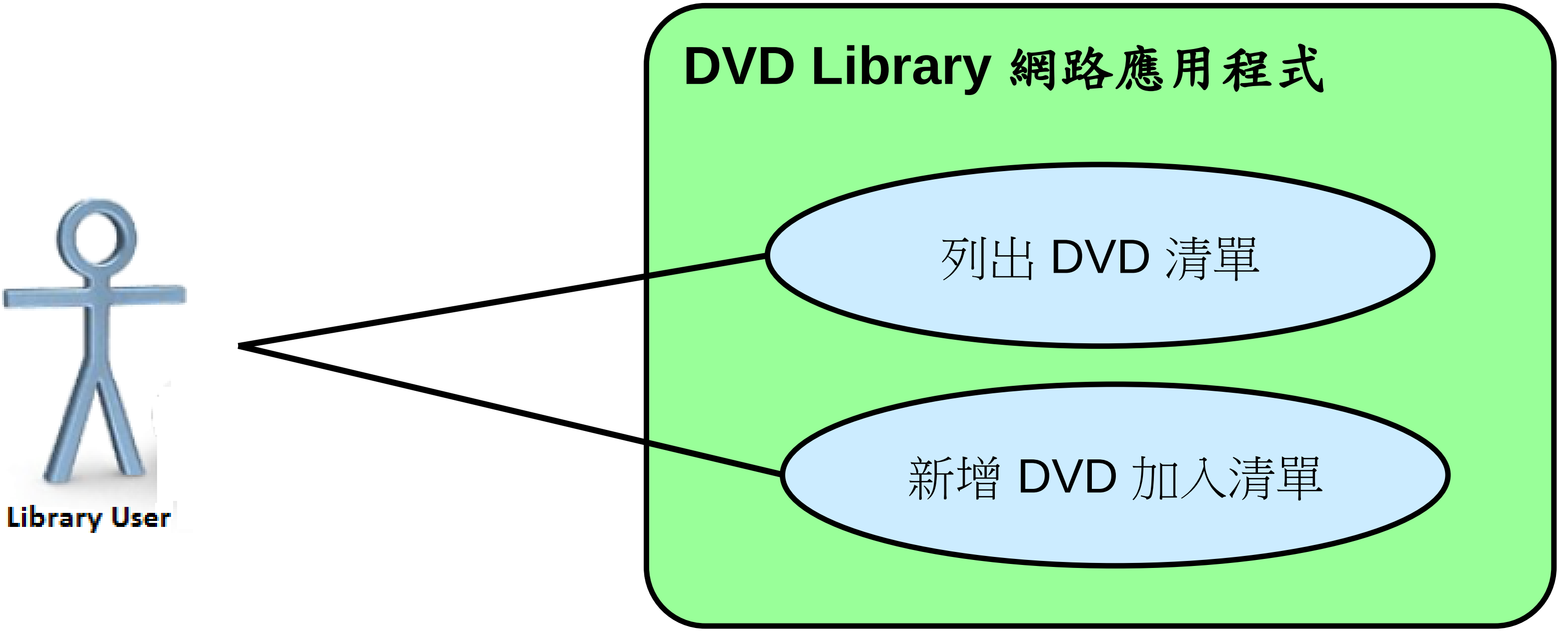
Duke's Soccer League: List Leagues

The set of soccer leagues:

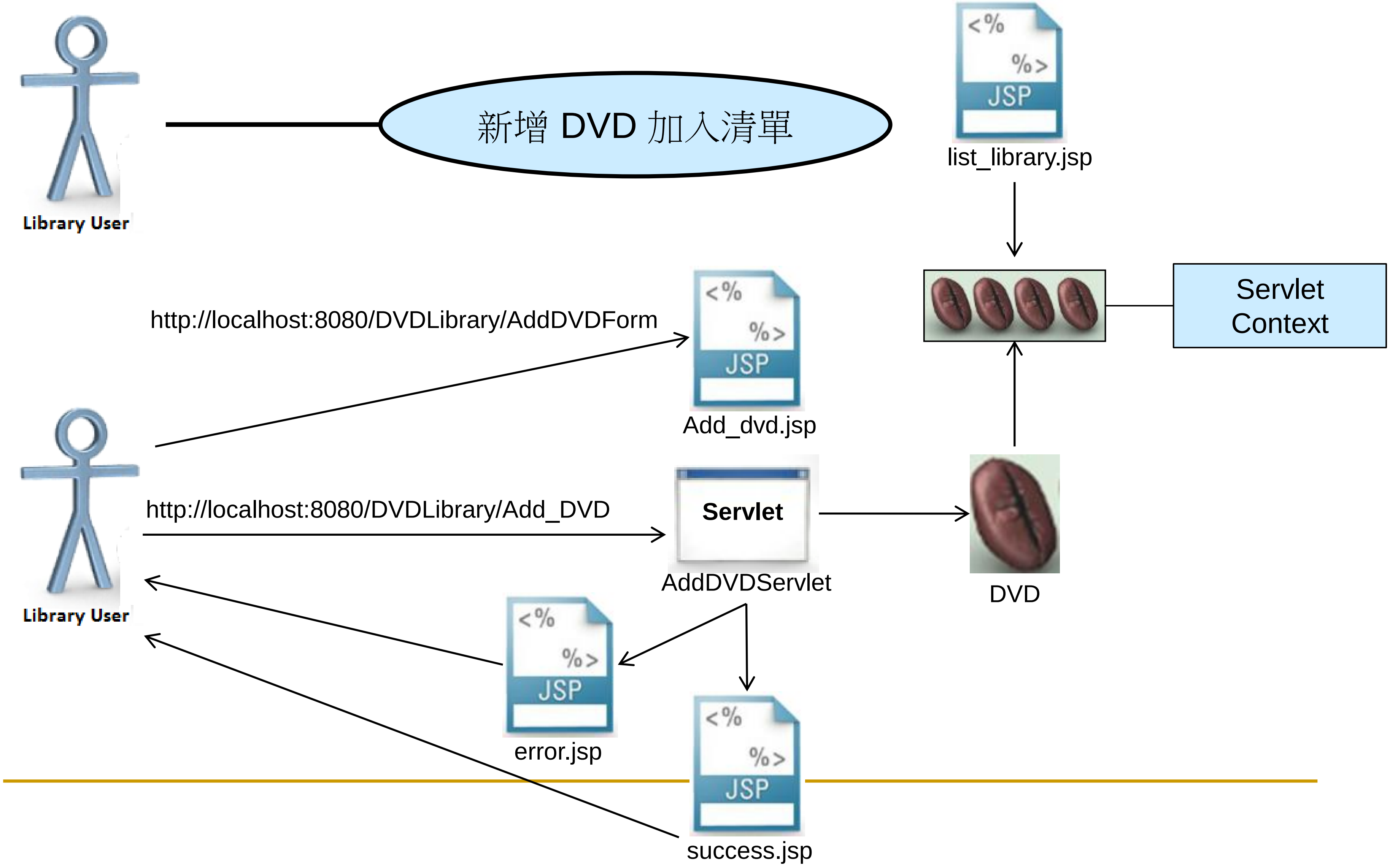
- Soccer League (Spring '03)
- Summer Soccer Fest 2003
- Autumn Soccer League (2003)
- Soccer League (Spring '04)
- The Summer of Soccer Love 2004
- Autumn Soccer League (2006)
- NBA Playoff (2013)

End of list.

Lab - DVDLibrary 應用程式



Lab 1


Library User
新增 DVD 加入清單
list_library.jsp
Servlet Context
http://localhost:8080/DVDLibrary/AddDVDForm
Add_dvd.jsp
Library User
http://localhost:8080/DVDLibrary/Add_DVD
Servlet
AddDVDServlet
DVD
error.jsp
success.jsp

Lab 1

- 修改 controller.AddDVDServlet.java
  - 表單驗證成功
    - 以title, year, genre建立DVDItem物件
    - 將DVDItem物件加入請求屬性dvdItem
    - 取得應用程式共用之DVD清單
      - DVD清單儲存於ServletContext 屬性 “DVDList“
      - 取得之物件轉型為List
      - 如果DVDList屬性不存在,新增一個LinkedList加入ServletContext屬性
    - 將DVDItem物件加入DVD清單
    - 將請求轉送至success.jsp

Lab 1

- 修改index.html
  - 修改List_Library連結至list_library.jsp
- 刪除 controller.ListLibraryServlet.java
- 修改 list_library.jsp
  - 取得應用程式共用之DVD清單
    - 使用隱含變數application取得ServletContext屬性 “DVDList“
  - 將DVD清單資訊顯示於網頁
- 測試、執行

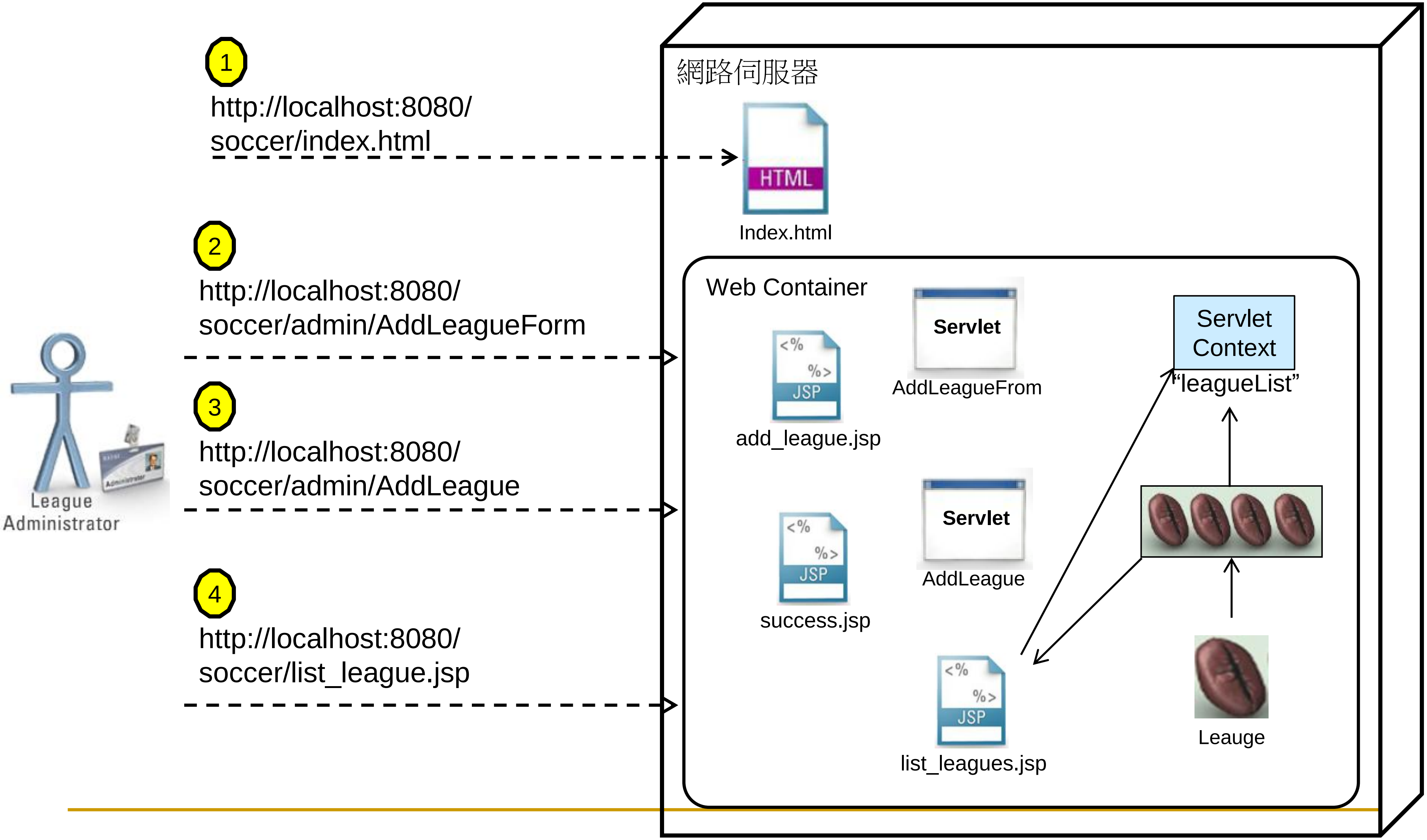
課程大綱

1) 應用程式資源在所有元件中共享

2) **初始化應用程式資源**

- **使用Servlet Context Listener**
- **設定網路應用程式初始屬性**

應用程式資源初始化



1
http://localhost:8080/soccer/index.html

2
http://localhost:8080/soccer/admin/AddLeagueForm

3
http://localhost:8080/soccer/admin/AddLeague

4
http://localhost:8080/soccer/list_league.jsp

League Administrator

網路伺服器

Index.html

Web Container

add_league.jsp

Servlet

AddLeagueFrom

Servlet

AddLeague

success.jsp

list_leagues.jsp

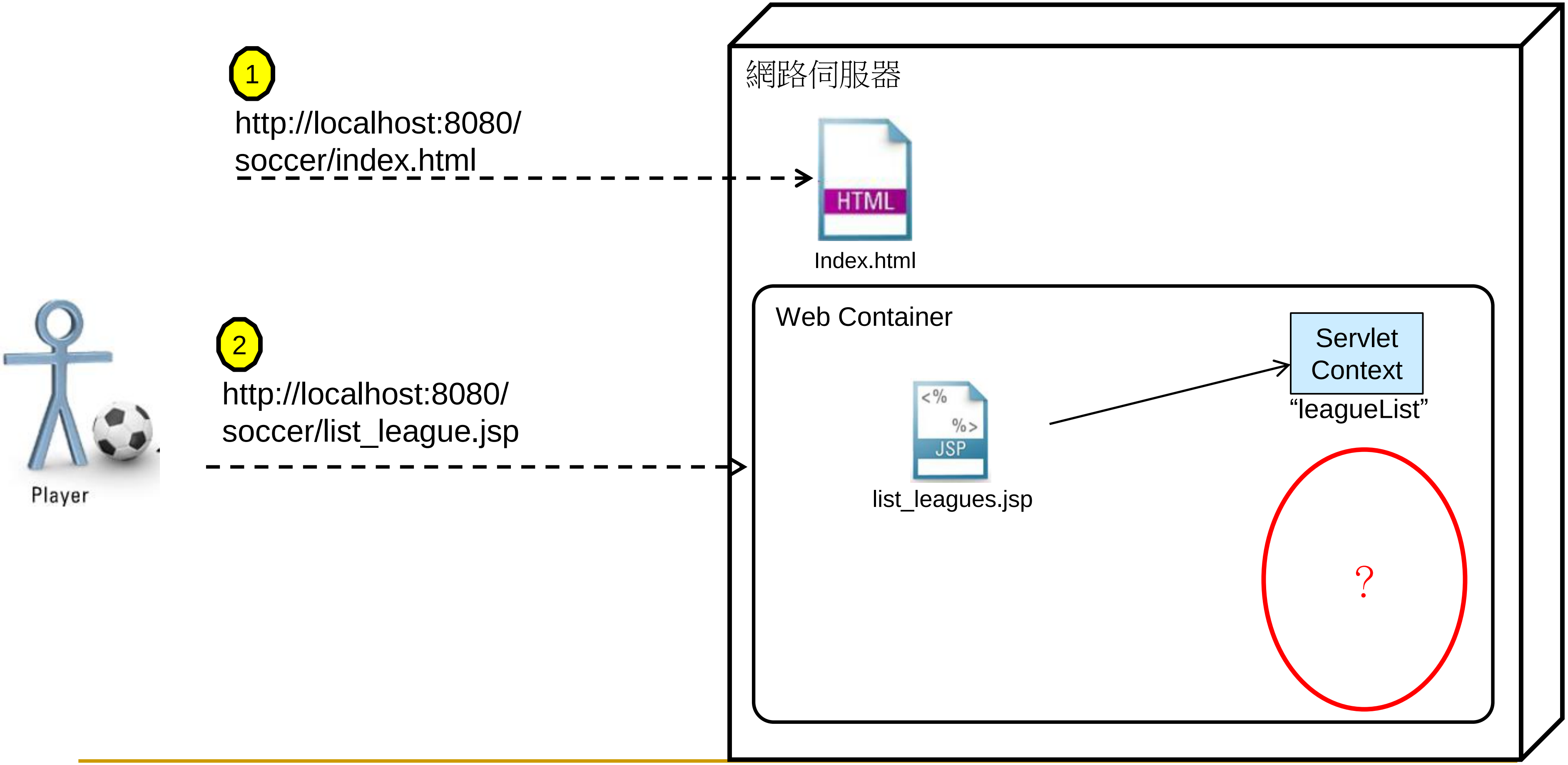
Servlet Context

"leagueList"

Leauge

應用程式資源初始化

網路應用程式在HTTPRequest 執行之前
共享的應用程式資料就必須存在記憶體中

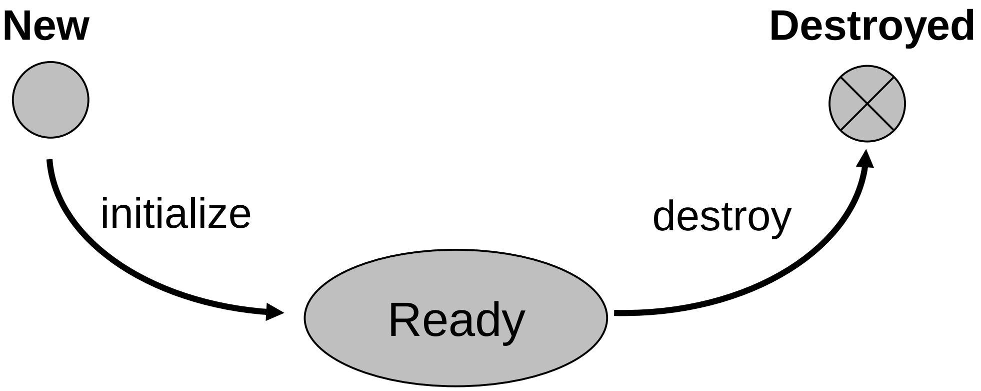


Web Application Life Cycle

- 網路應用程式生命週期

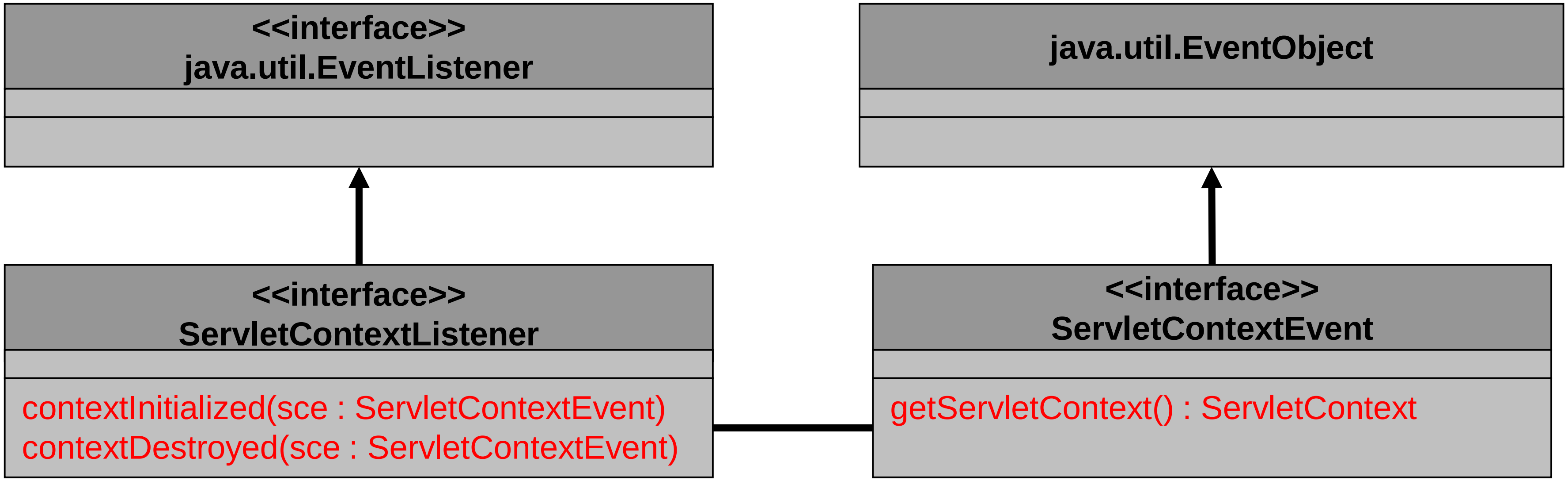
生命週期階段	說明
初始階段	當Web Server啟動時,每個網路應用程式都會被初始化
執行階段	根據客戶戶端請求,呼叫適當Servlet予以處理,並產生回應
結束階段	當Web Server關閉時,釋放網路應用程式佔用之系統資源

- Servlets 操控都在執行階段



javax.servlet.ServletContextEvent

- 初始階段及結束階段操控
  - 採用事件委任模型(Event Delegation Model)
    - 網路應用程式初始及結束時, 容器產生 ServletContextEvent
    - 實作ServletContextListener interface來處裡事件
    - 使用web.xml或annotation進行委派(delegate)



<<interface>>
java.util.EventListener
java.util.EventObject
<<interface>>
ServletContextListener
contextInitialized(sce : ServletContextEvent)
contextDestroyed(sce : ServletContextEvent)
<<interface>>
ServletContextEvent
getServletContext() : ServletContext

Deployment Descriptor – ServletContext ServletContextEvent

```
<web-app>
    <display-name>  Web application 名稱.  </display-name>
    <description>  Web application 說明.  </description>
    <context-param>
        <param-name> Context初始參數名稱 . </param-name>
        <param-value> Context初始參數值 . </param-value>
    </context-param>
    <listener>
        <listener-class> Context事件處理類別 . </listener-class>
    </listener>
    <servlet>
        ……..
        ……..
</web-app>
```

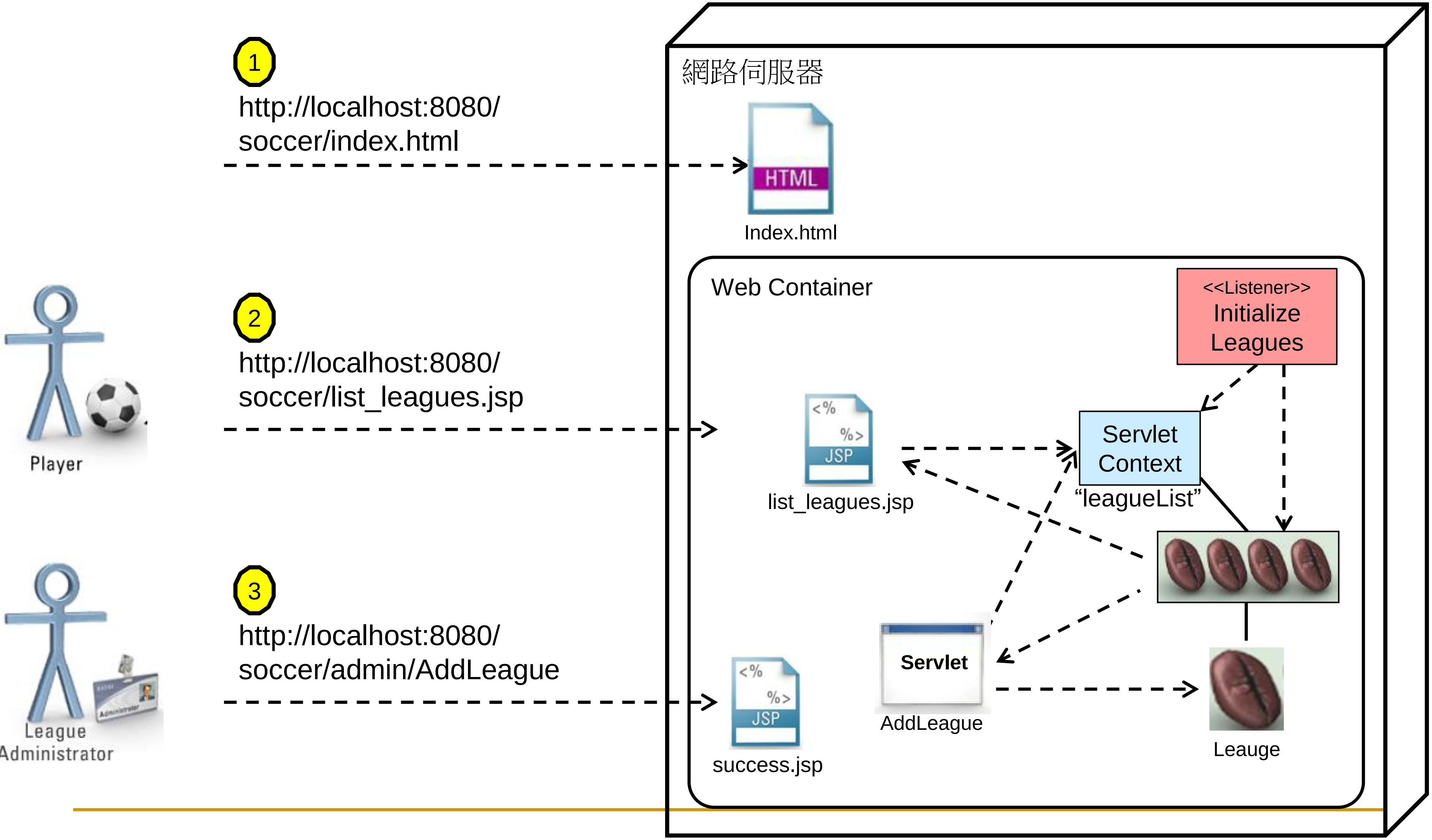
網路應用程式啟動時,呼叫此類別之contextInitialized()
網路應用程式關閉時,呼叫此類別之contextDestroyed()

使用 Annotation 標註

- Annotation 標註描述作部署描述

```
import javax.servlet.annotation.WebListener
……
@WebListener
public class MyListener implements ServletContextListener {
      ……
      ……
}
```

應用程式資源初始化 (再次回顧)



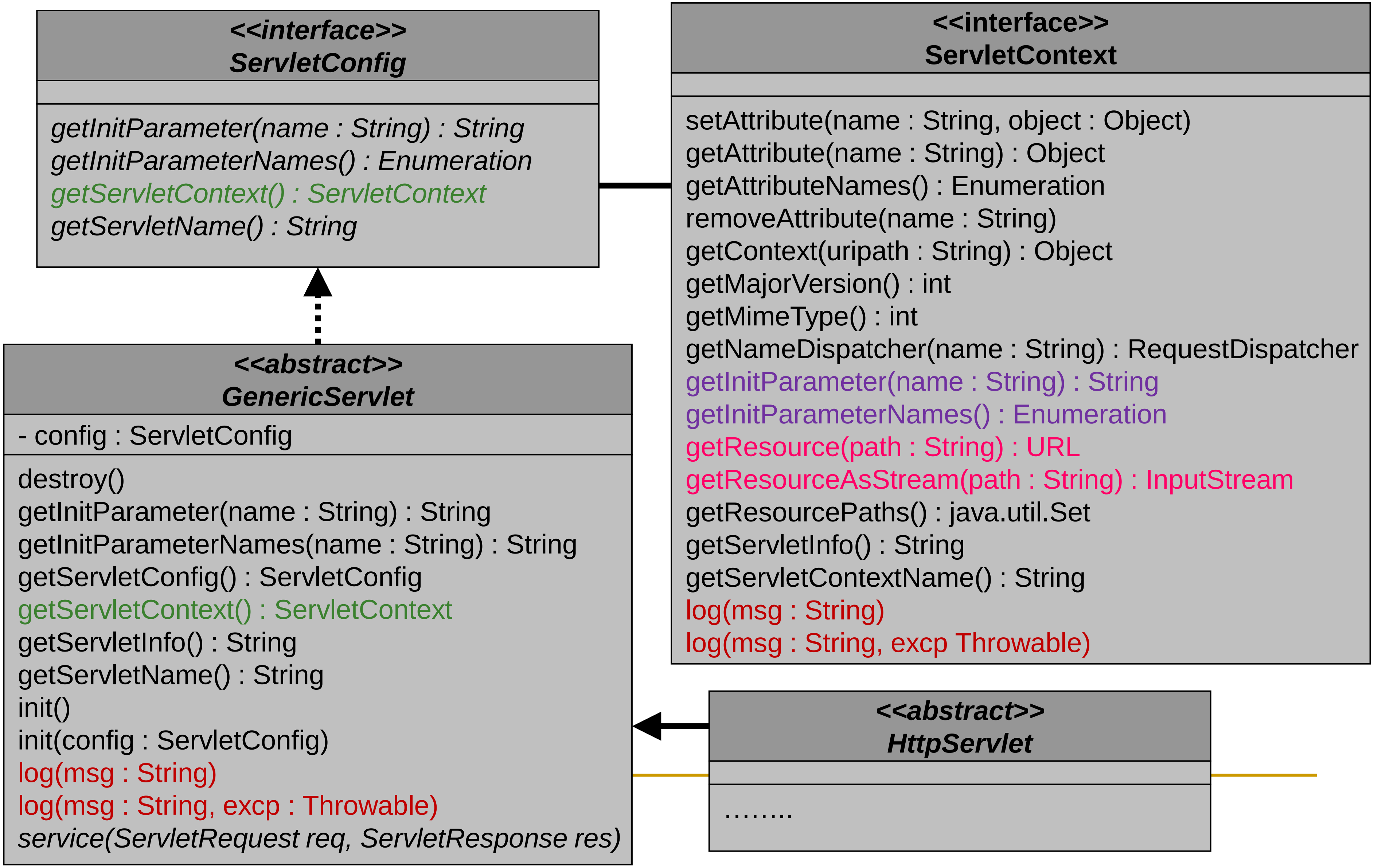
context-scope 屬性初始化

- leagueList context-scope 屬性初始化：
  1. 決定leagues.txt 檔案位置。
  2. 讀取leagues.txt 檔案。
  3. 在leagues.txt 檔案裡每一列建立League 物件，並存到 List 物件中。
  4. 在leagueList 環境屬性中存放聯盟清單。
  5. 記錄清單被初始化或任何被程式碼丟出的例外事件

/WEB-INF/data/leagues.txt

```
2019    Spring    Soccer League (Spring '19)
2019    Summer    Summer Soccer Fest 2019
2019    Autumn    Autumn Soccer League (2019)
2020    Spring    Soccer League (Spring '20)
2020    Summer    The Summer of Soccer Love 2020
2020    Autumn    Autumn Soccer League (2020)
```

javax.servlet.ServletContext



<<interface>>
ServletConfig

getInitParameter(name : String) : String
getInitParameterNames() : Enumeration
getServletContext() : ServletContext
getServletName() : String

<<interface>>
ServletContext

setAttribute(name : String, object : Object)
getAttribute(name : String) : Object
getAttributeNames() : Enumeration
removeAttribute(name : String)
getContext(uripath : String) : Object
getMajorVersion() : int
getMimeType() : int
getNameDispatcher(name : String) : RequestDispatcher
getInitParameter(name : String) : String
getInitParameterNames() : Enumeration
getResource(path : String) : URL
getResourceAsStream(path : String) : InputStream
getResourcePaths() : java.util.Set
getServletInfo() : String
getServletContextName() : String
log(msg : String)
log(msg : String, excp Throwable)

<<abstract>>
GenericServlet

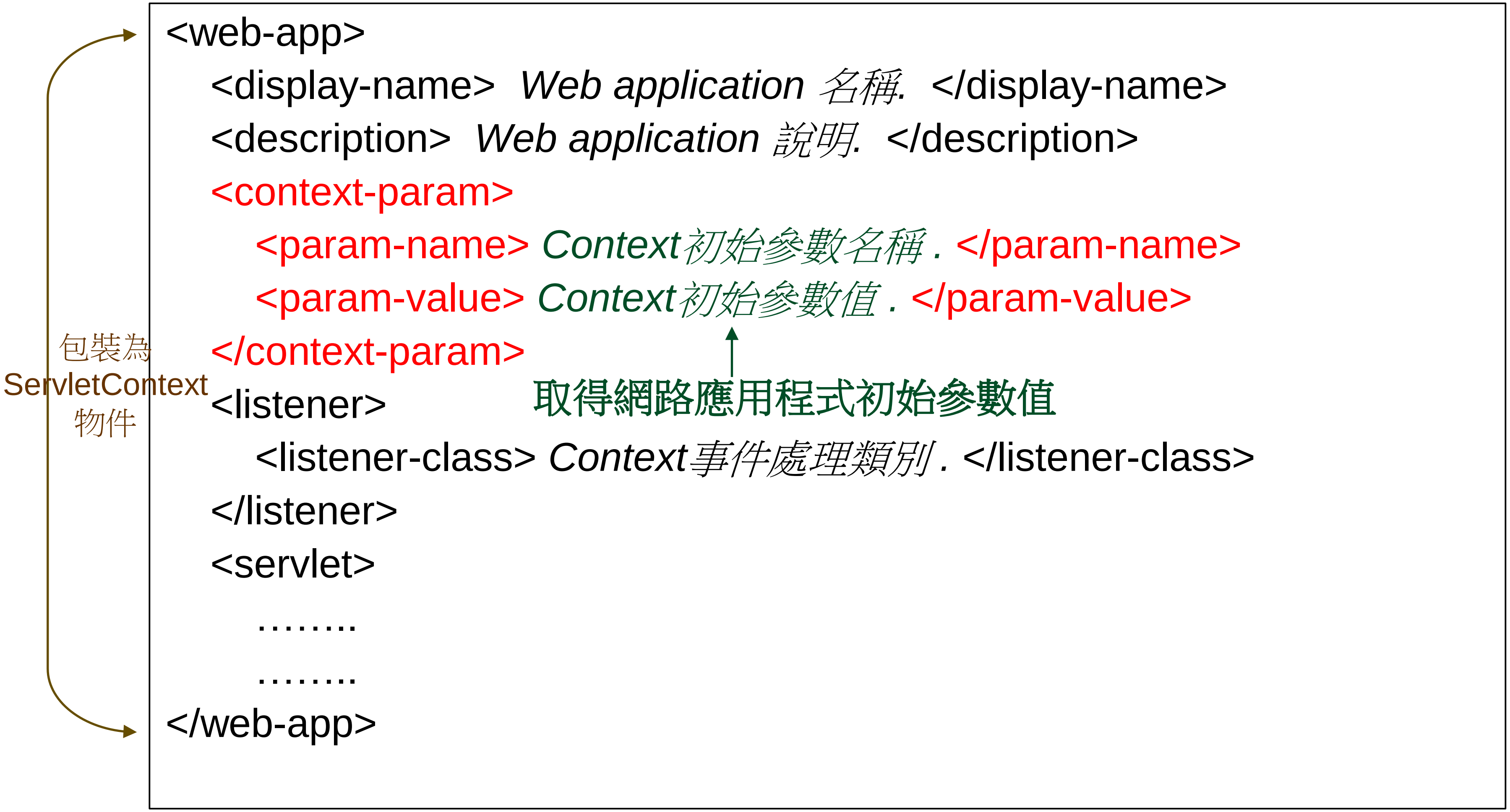
- config : ServletConfig

destroy()
getInitParameter(name : String) : String
getInitParameterNames(name : String) : String
getServletConfig() : ServletConfig
getServletContext() : ServletContext
getServletInfo() : String
getServletName() : String
init()
init(config : ServletConfig)
log(msg : String)
log(msg : String, excp : Throwable)
service(ServletRequest req, ServletResponse res)

<<abstract>>
HttpServlet

……..

Deployment Descriptor – ServletContext ServletContextEvent



```
<web-app>
    <display-name>  Web application 名稱.  </display-name>
    <description>  Web application 說明.  </description>
    <context-param>
        <param-name> Context初始參數名稱 . </param-name>
        <param-value> Context初始參數值 . </param-value>
    </context-param>
    <listener>
        <listener-class> Context事件處理類別 . </listener-class>
    </listener>
    <servlet>
        ……..
        ……..
</web-app>
```

包裝為 ServletContext 物件

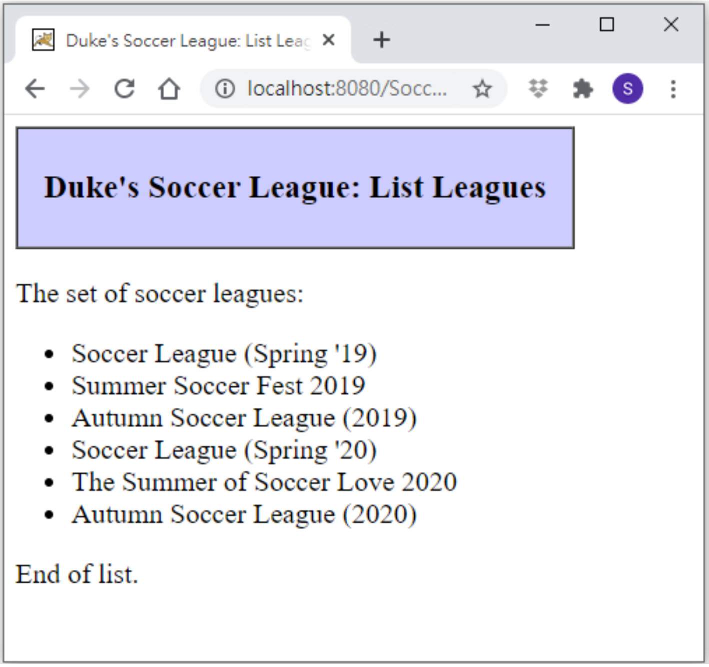
取得網路應用程式初始參數值

Sc

```java
package web;
import javax.servlet.*, java.util.*, java.io.*, javax.servlet.annotation.*, model.League;
@WebListener
public class InitializeLeagues implements ServletContextListener {
    @Override
    public void contextInitialized(ServletContextEvent sce) {
        ServletContext context = sce.getServletContext();
        List<League> leagueList = new LinkedList<League>();
        String leaguesFile = context.getInitParameter("leaguesFile");
        InputStream is = null;        BufferedReader reader = null;
        try {
            is = context.getResourceAsStream(leaguesFile);
            reader = new BufferedReader(new InputStreamReader(is));
            String record;
            while ( (record = reader.readLine()) != null ) {
                String[] fields = record.split("\t");
                int year = Integer.parseInt(fields[0]);
                String season = fields[1];
                String title = fields[2];
                League item = new League(year, season, title);
                leagueList.add(item);
            }
            context.setAttribute("leagueList", leagueList);
            context.log("The league list has been loaded.");
        } catch(Exception e){
            context.log("Exception occured while processing the leagues file.", e);
        } finally {
            if ( is != null ) {    try { is.close(); } catch (Exception e) {}     }
            if ( reader != null ) {    try { reader.close(); } catch (Exception e) {}    }
        }
    }
    @Override
    public void contextDestroyed(ServletContextEvent sce) {    context.log(….);     }
}
```

Soccer 範例

```xml
<?xml version="1.0" encoding="UTF-8"?>
<web-app>
    <context-param>
        <param-name>leaguesFile</param-name>
        <param-value>/WEB-INF/data/leagues.txt</param-value>
    </context-param>
    <listener>
        <description>ServletContextListener</description>
        <listener-class>web.InitializeLeagues</listener-class>
    </listener>
</web-app>
```

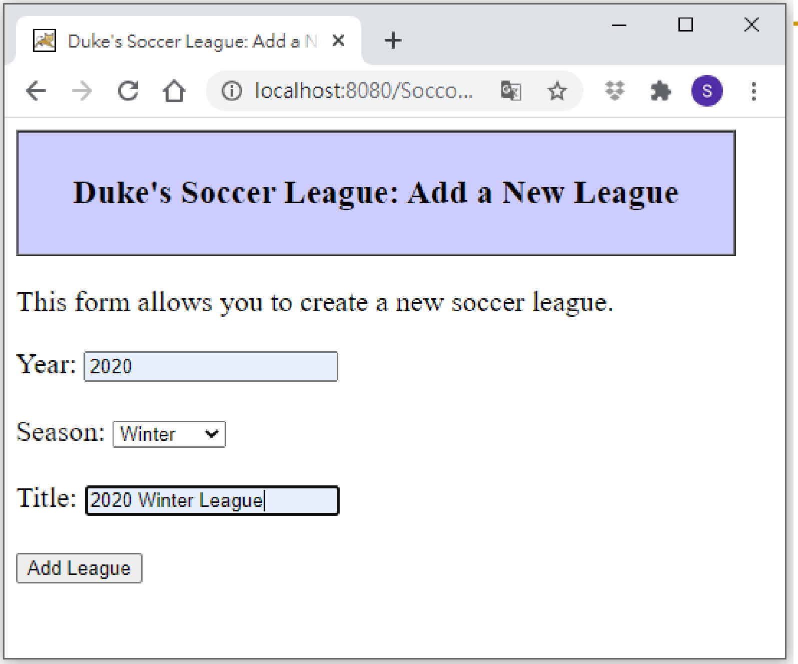


Duke's Soccer League: List Leagues

The set of soccer leagues:

- Soccer League (Spring '19)
- Summer Soccer Fest 2019
- Autumn Soccer League (2019)
- Soccer League (Spring '20)
- The Summer of Soccer Love 2020
- Autumn Soccer League (2020)

End of list.



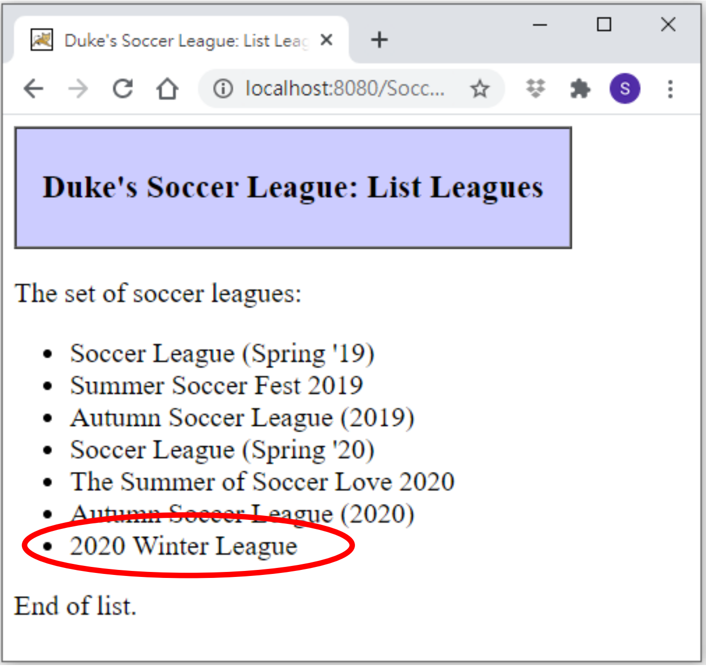
Duke's Soccer League: Add a New League

This form allows you to create a new soccer league.

Year: 2020

Season: Winter

Title: 2020 Winter League

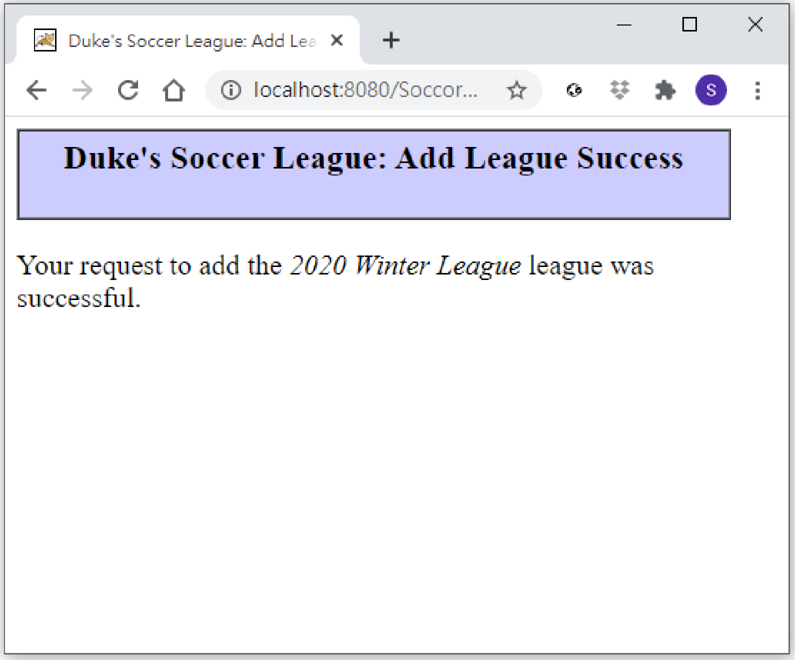


Duke's Soccer League: List Leagues

The set of soccer leagues:

- Soccer League (Spring '19)
- Summer Soccer Fest 2019
- Autumn Soccer League (2019)
- Soccer League (Spring '20)
- The Summer of Soccer Love 2020
- Autumn Soccer League (2020)
- 2020 Winter League

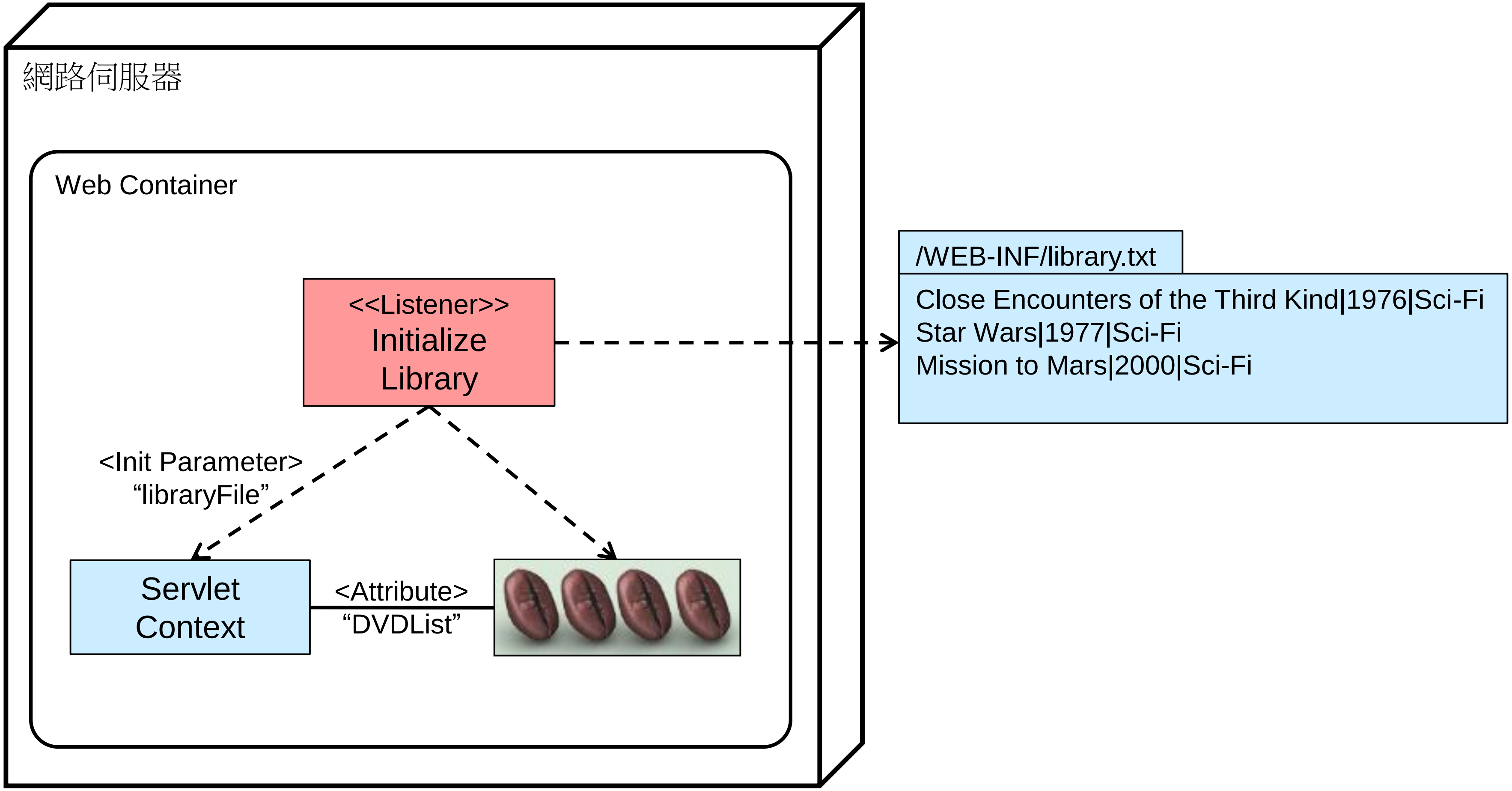
End of list.



Duke's Soccer League: Add League Success

Your request to add the *2020 Winter League* league was successful.

Lab 2 應用程式資源初始化



Close Encounters of the Third Kind|1976|Sci-Fi
Star Wars|1977|Sci-Fi
Mission to Mars|2000|Sci-Fi

Lab 2

- 新增Context Listener
  - 新增一個Context Listener 檔 web.InitializeLibrary.java
  - 修改 contextInitialized(ServletContextEvent sce)
    - 由 ServletContextEvent 取得 ServletContext 物件
    - 由 ServletContext 取得應用程式初始參數 "libraryFile"
      - Library資料檔所在位置
    - 建立一個LinkedList儲存DVD清單
    - 由 ServletContext物件建立讀取資料檔的InputStream物件
      - getResourceAsStream()
      - 再包裝為BufferedReader物件
    - 讀取資料檔資料
      - 讀取每一行資料
      - 使用字串的split()切割
      - 建立對應之DVDItem物件
      - 將DVDItem物件加入DVD清單
    - 將DVD清單加入ServletContext 屬性 "DVDList"

Lab 2

- 建立Library資料檔
  - 複製 library.txt 至 <專案路徑>\web\WEB-INF 資料夾之下
- 網路應用程式設定資料
  - 新增 web.xml
  - 加入初始參數 "libraryFile“
  - 參數值為/WEB-INF/library.txt
- 測試、執行