

---

# Examination: Java EE Presentation Layer

Library

Dawid Loubser

## Table of Contents

General Theory .....	1
Practical .....	1
Scenario .....	1
Setup .....	1
Practical .....	1

## General Theory

1. Explain your understanding of the differences, and why one would use, a “client-side redirect” (returning an HTTP 302 FOUND message) and a server-side redirect (e.g. a Servlet using a `requestDispatcher` to dispatch the request to another servlet or JSP)
2. Explain roughly how you would go about using a Servlet Filter to, in real-time, monitor which search engines (Google, MSN, etc) provide the most traffic to your web site.

## Practical

### Scenario

Consider the existing, very simplistic EJB-based business layer (provided to you by the examiner) which offers simple services around managing books (adding new books, assigning them to categories, etc).

The following is provided in the business layer:

- **library-api.jar**. The JAR containing the “contract” of the library service, which you will need to package with any web application which uses it. The core interface is called `Library`
- **javaDocs/**. JavaDocs explaining the different services offered by the `Library` interface
- **library.jar**. An EJB3 implementation of the `Library`, which uses the Java Persistence API to store books and categories in the database.

### Setup

You need to install an application server (Glassfish is recommended) and deploy the `library.jar` EJB module to it. Every time it is deployed, it should create new empty database structures automatically.

## Practical

### Ant-based project

Create a project for a web application which will use the `Library` business layer. It must use Apache Ant to build a `.war` file, containing all necessary elements to deploy to your app server.

You are most welcome to use an existing project template of this nature, such as those provided to you during the course. It is important, however, that this project is build-able purely via the command-line and ant, although you may use any editors (Eclipse, Oxygen XML, etc) to create your Java classes and JSP pages.

## Requirements

Use a combination of JSP and Servlet components, to create a web application which has the following capabilities:

1. One must be able to add a new book to the library via form-based input
2. One must be able to add categories to the library via form-based input
3. One must be able to see a listing of all books in a given category, by clicking on the category name
4. Every page should (via `jsp:include`) display a menu allowing the user to perform these actions

Use *dependency injection* to get a handle to the EJB implementing the `Library` interface in your Servlets etc. This way, you do not have to know the JNDI name or location of the EJB business object.

```
public class MyServlet extends HttpServlet
{
    ...

    @EJB
    private Library library;
}
```

## Note

Visual styling is not important to this project, so there is no need to make it pretty via CSS (Cascading Style Sheets)

## Handing in your examination

You need to provide the examiner with a ZIP file, which contains your theory questions, as well as your Ant-based project, and instructions how to build / deploy it.