

## Assignment # 1–Developing a Spring MVC application

**Due Date:** Week 5

**Purpose:** By finishing this assignment you will practice to:

- Develop and code a spring app with MVC capabilities.
- Debug and run a spring MVC app.

**References:** Read the lecture slides week 2 – 4 and lab exercises. This material provides the necessary information you need to complete the exercises.

### Instructions and rules:

- This lab should be completed by every student **individually**.
- You will have to demonstrate your solution in a scheduled lab session and submitting the project **through the assignment link on Blackboard**. You must name your Eclipse project according to the following rule:

**Your Name\_COMP303\_AssignmentNumber**

Example: **John\_COMP303\_Assignment1**

### Description

Celebrity cruises are luxury cruises which provides vacation packages for travelers of every age. You are developing an MVC app one of the celebrity cruise business by creating the following programs: Configuration files: POM.xml, dispatcher-servlet.xml and web.xml and controller java file, Model java file, JSP view files.

### Task 1:

Design a JSP page named “**index.jsp**” that handles a POST request and an action on “**CruiseController**” java using a **@RequestMapping**. The page should have “Online Cruise Booking Request” form to get user input as shown in this snap shot given in the next page:

Complete this page design with a “Book Now” submit button and all other html elements (controls) to make this page user friendly with basic validation. Your design should use appropriate layout using div or tables and other formatting features with colors, fonts and styles. You can use a reset button to clear user input if needed.

## Cruise Booking Request Form

---

**Passenger Name**

<input type="text"/>	<input type="text"/>
First Name	Last Name

**Contact Number**

<input type="text"/>	–	<input type="text"/>
Area Code		Phone Number

**Address**  
Street Address  
Street Address Line 2

<input type="text"/>	<input type="text"/>
City	State / Province

<input type="text"/>	<input type="text" value="Please Select"/>
Postal / Zip Code	Country

**E-mail**  
example@example.com**How many Stateroom****Stateroom Type****Price****How many Guests****Cabin Type (If Applicable)**

- ☐ Inside Cabin  
☐ Outside Cabin

**Is Anyone Traveling Over The Age Of 55?**

- ☐ Yes  
☐ No

(Template from jotforms)

**Hint:** Stateroom types are Ocean view, Aqua class, Veranda, Sky Suite and Royal Suite and the prices are different for each rooms.

**Task 2:**

Define a model bean class “**Cruise.java**” with the appropriate properties and methods to handle the user input data.

**Task 3:**

Define a “**CruiseController.java**” class to get the data from “**index.jsp**” form page and publish the output as the following format in the **pages/show-cruise.jsp** page.

You should show the output as given below:

---

**Welcome Max.**

Your Booking details are as follows:

- **Email Address:** [amax@yahoo.com](mailto:amax@yahoo.com)
- **Phone Number:** 600-900-8000
- **Cruise Type:** Ocean View
- **No of Guests:** 2
- **Price :** CAD 1,700

**Cost of Trip**

**Ocean View Stateroom**

First Guest	CAD 1,700.00
Second Guest	CAD 1,700.00
Discount	CAD 400.00
Taxes and Fees	CAD 500.00
Cruise Trip Total	CAD 2,500.00

---

**Deliverables:**

1. Your app must include at least the following code files:
  - Controller file – CruiseController.java
  - Model file – Cruise.java
  - View file – print-cruise.jsp
  - index.jsp and other jsp files if needed
  - Configuration files – web.xml, dispatcher-servlet.xml and pom.xml
  - Formatting CSS files.
2. The assignment source code (Eclipse project with the necessary files) submitted as a .zip file format – “YourName\_StudentNo\_Assign1.zip”.

**Assessment Rubrics:**

<b>Functionalities:</b> Developing an app with MVC capabilities including all the files listed in the deliverables. Provide explanation when asked during the demonstration of the app.	75%
<b>Configuration of XML files</b>	5%
<b>UI friendliness</b> (proper layout, colors, fonts, use of CSS / bootstraps)	10%
<b>HTML</b> form controls validations	5%
<b>Code stands</b> , use of variables, comments, and naming conventions.	5%
<b>Total</b>	<b>100%</b>

**Submission Rules:**

Each file submitted in the solution should have student name, student number and submission date in the top of the code file. (Code standard)

Comments may be necessary inside the functions or predicates but that the names of the functions and predicates and the comments you include to explain each are especially important. (Code standard)

Student must hand in the assignment to e-centennial drop box **before in-class demonstration**.

**Late submission:**

There is a **20% off per day** late penalty which means if a student submits an assignment one day late and does perfect work, the most they can obtain is 80/100.

For example, if a student submits one day late and obtains 8/10 as an assignment then the professor will need to deduct 20% from this grade ( $8 \times .20 = 1.6$ ,  $8 - 1.6 = 6.4/10$  - record 6.4). If a student submits two days late then 40% will be deducted, for example, assume the student obtains 9/10 ( $9 \times .40 = 3.6$ ,  $9 - 3.6 = 5.4/10$ ). By day five the student receives zero.

**Academic honesty (Plagiarism and cheating)**

All students must follow the academic honesty policies regarding Plagiarism and cheating on assignments, Quizzes or Tests. Centennial college's Academic Policy will be strictly enforced. To support academic honesty at Centennial College, all academic work submitted by students may be reviewed for authenticity and originality, with utilizing software tools.

For more details, please visit the Academic Honesty site on

<https://www.centennialcollege.ca/mycentennial/your-support/academic-support/student-academic-advising/academic-honesty/>