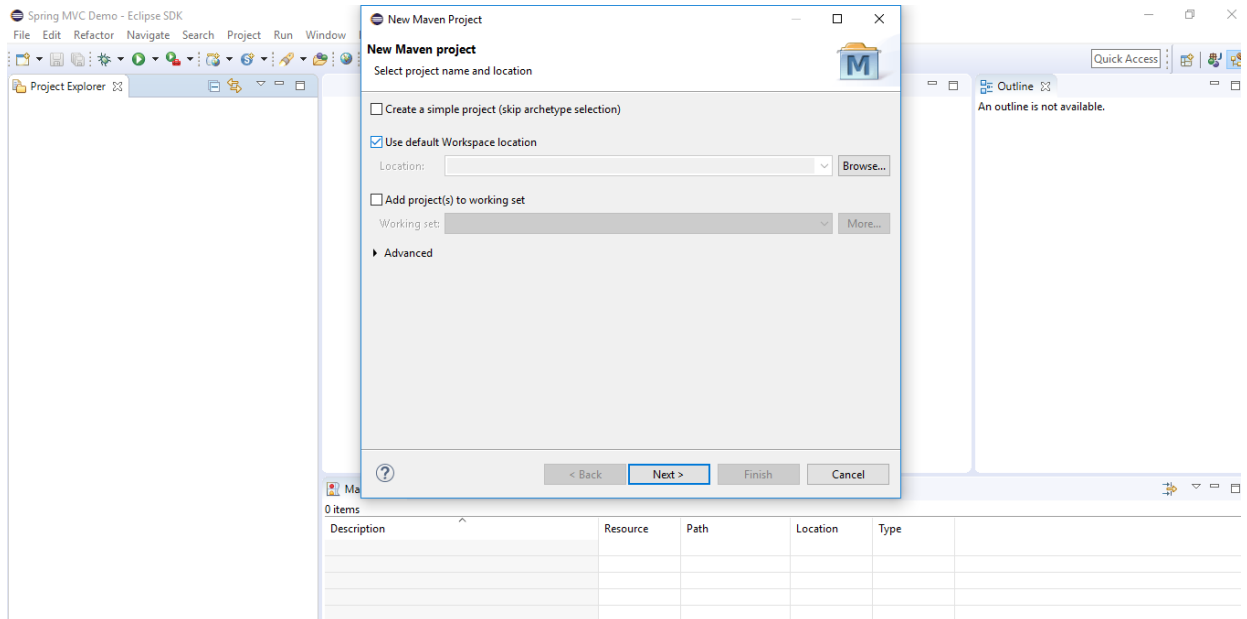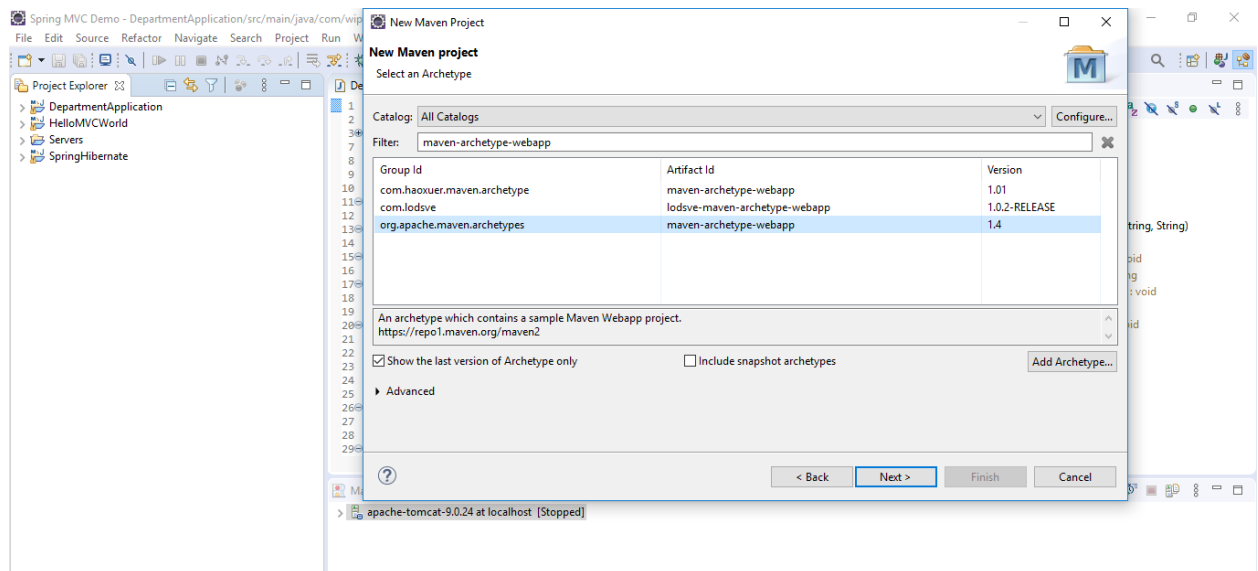Spring MVC with Forms
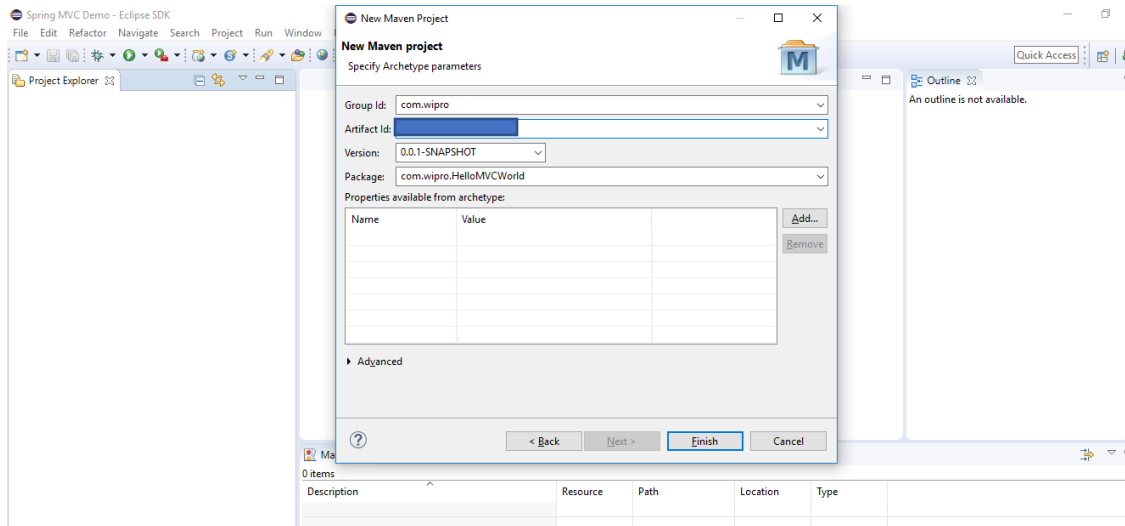
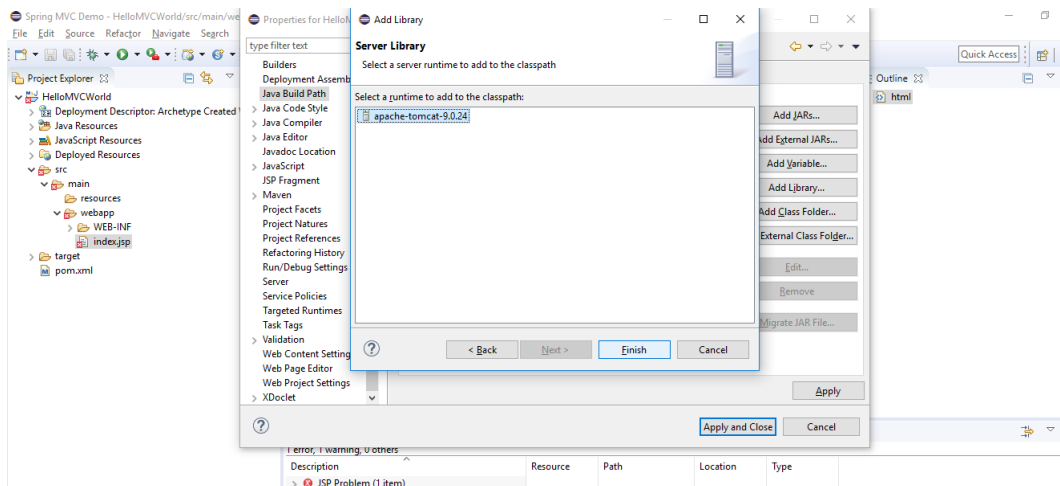1. Create Maven Project and choose use default workspace location
   a. Click Next

2. Under Archetype choose **maven-archetype-webapp** from below given group Id and version
   a. Group Id : **org.apache.maven.archetypes**
   b. Version can be any latest version. Here It is taken 1.4
      i. You can also pick a specific version by checking the ***Include snapshot archetypes***
   c. Click Next

3. Fill the Maven Project details
   a. Group ID :: com.wipro
   b. Artifact ID :: DepartmentApp
   c. Click finish

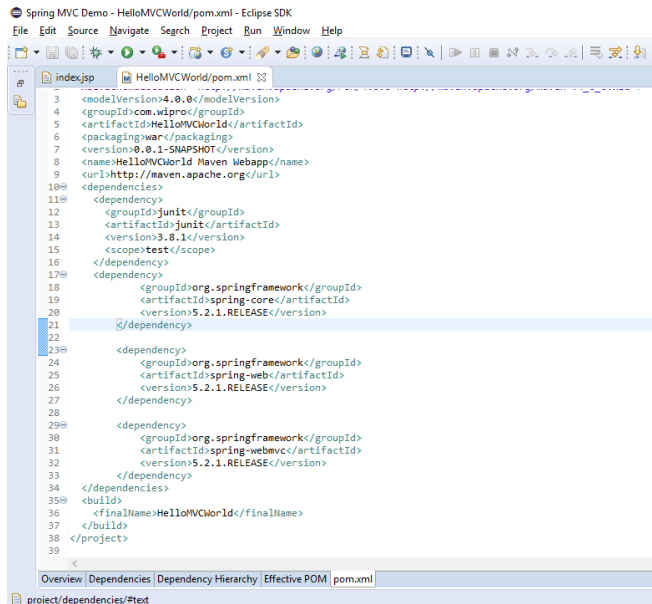4. This generates Spring Web Project. With the initial Error
   a. This is because the build path is not aware of servlets and jsp APIs.
   b. Add Tomcat server
      i. Click on window menu -> preferences -> server -> Runtime Environment
         1. Search the unzipped folder of your Apache Tomcat
   c. right click on project -> Properties -> Java Build Path -> Add Library
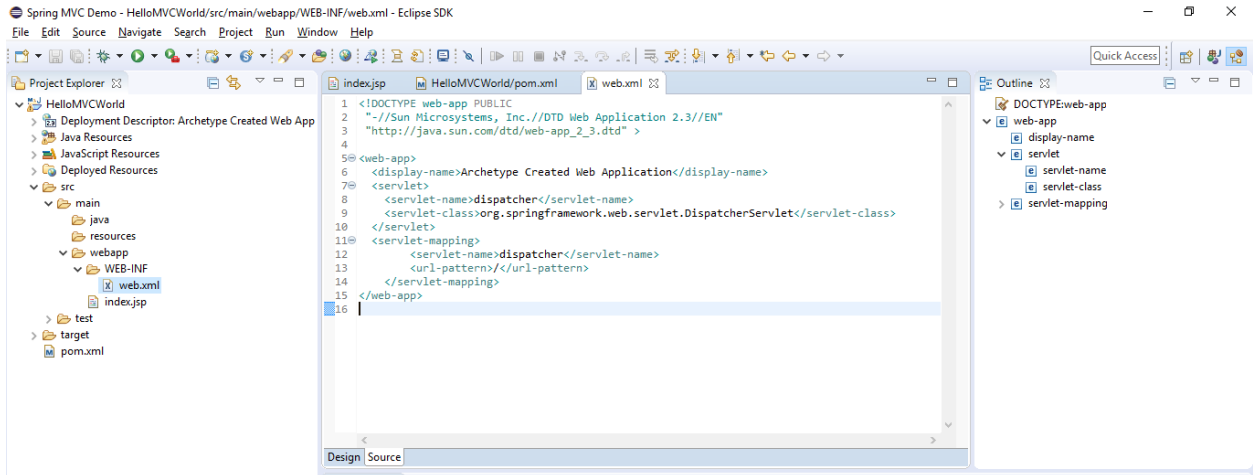   d. Server Runtime -> Apache Tomcat -> Finish

5. Set the spring dependencies.
   a. Open POM file
   b. Add the following spring dependencies under the tag <dependencies>

```xml
<dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-core</artifactId>
        <version>5.2.1.RELEASE</version>
</dependency>
<dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-web</artifactId>
        <version>5.2.1.RELEASE</version>
</dependency>
<dependency>
        <groupId>org.springframework</groupId>
        <artifactId>spring-webmvc</artifactId>
        <version>5.2.1.RELEASE</version>
</dependency>
```


6. Set the dispatcher Servlet.
   a. Edit the **web.xml** file from project path : src->main->webapp->WEB-INF->web.xml
   b. Include <servlet> and <servlet-mapping>

```xml
<servlet>
    <servlet-name>dispatcher</servlet-name>
    <servlet-class>org.springframework.web.servlet.DispatcherServlet
    </servlet-class>
    </servlet>
    <servlet-mapping>
        <servlet-name>dispatcher</servlet-name>
        <url-pattern>/</url-pattern>
    </servlet-mapping>
```
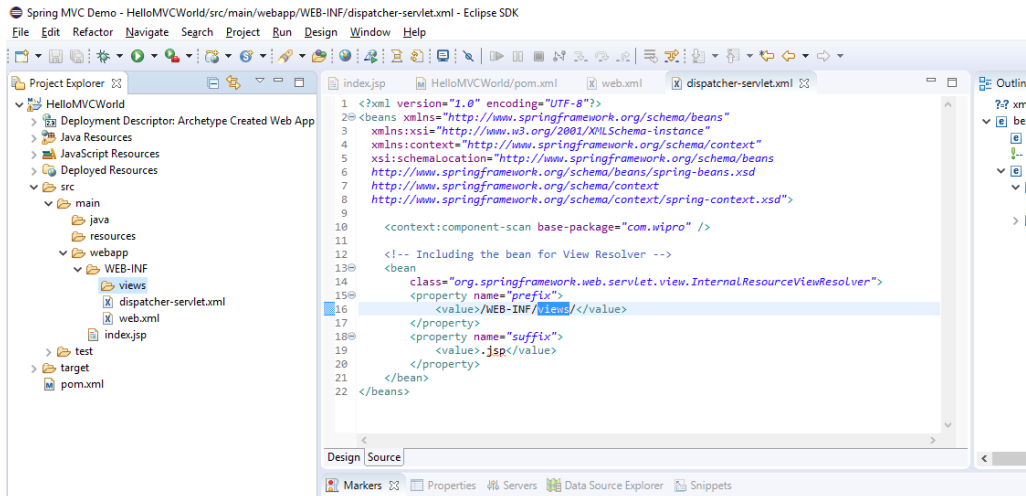
7. Create the Spring configuration file names dispatcher-servlet.xml

```xml
<?xml version="1.0" encoding="UTF-8"?>
<beans xmlns="http://www.springframework.org/schema/beans"
  xmlns:xsi="http://www.w3.org/2001/XMLSchema-instance"
  xmlns:context="http://www.springframework.org/schema/context"
  xsi:schemaLocation="http://www.springframework.org/schema/beans
  http://www.springframework.org/schema/beans/spring-beans.xsd
  http://www.springframework.org/schema/context
  http://www.springframework.org/schema/context/spring-context.xsd">

        <context:component-scan base-package="com.wipro" />

        <!-- Including the bean for View Resolver -->
<bean
class="org.springframework.web.servlet.view.InternalResourceViewResolver">
                <property name="prefix">
                        <value>/WEB-INF/views/</value>
                </property>
                <property name="suffix">
                        <value>.jsp</value>
                </property>
        </bean>
</beans>
```

8. Create the views folder under WEB-INF for defining the views
   a. Right click on WEB-INF folder NEW -> Folder (Enter Name as **views**) - > finish.

9. Create a class called Department under the package com.wipro.bean

```java
package com.wipro.bean;

public class Department {

        private int deptno;
        private String dname;
        private String loc;
        public Department() {

        }
        public Department(int deptno, String dname, String loc) {
                super();
                this.deptno = deptno;
                this.dname = dname;
                this.loc = loc;
        }
        public int getDeptno() {
                return deptno;
        }
        public void setDeptno(int deptno) {
                this.deptno = deptno;
        }
        public String getDname() {
                return dname;
        }
        public void setDname(String dname) {
                this.dname = dname;
        }
        public String getLoc() {
```

```
                    return loc;
          }
          public void setLoc(String loc) {
                    this.loc = loc;
          }
          @Override
          public String toString() {
                    return "Department [deptno=" + deptno + ", dname=" + dname + ", loc=" + loc +
"]";
          }
}
```

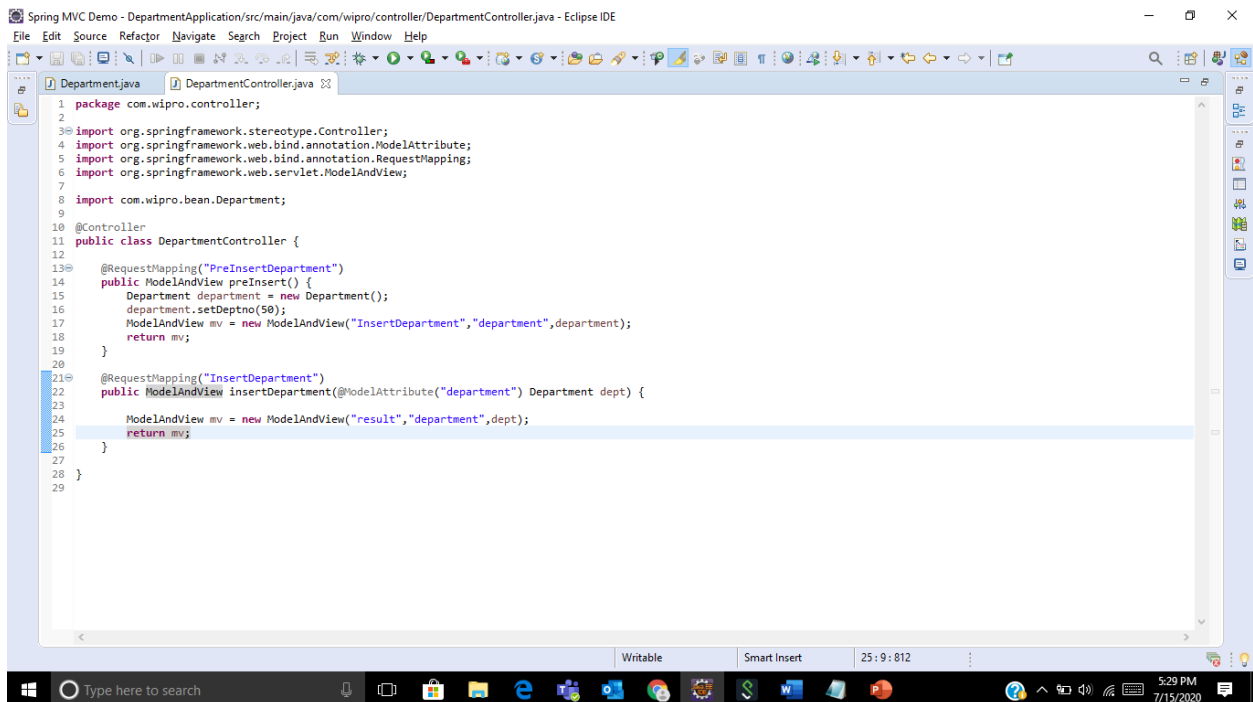10. Edit the Index.jsp page to request for a inserting operation for Department

```html
<html>
<body>
<h2>Hello World!</h2>
<a href="PreInsertDepartment">Insert Department</a>
</body>
</html>
```

11. Create the controller class named **DepartmentController**
    a. Right click on Java Resources and create a package called com.wipro.controller
    b. Under the package Create a class called **DepartmentController**


```java
package com.wipro.controller;

import org.springframework.stereotype.Controller;
```

```java
import org.springframework.web.bind.annotation.ModelAttribute;
import org.springframework.web.bind.annotation.RequestMapping;
import org.springframework.web.servlet.ModelAndView;

import com.wipro.bean.Department;

@Controller
public class DepartmentController {

    @RequestMapping("PreInsertDepartment")
    public ModelAndView preInsert() {
        Department department = new Department();
        department.setDeptno(50);
        ModelAndView mv = new
ModelAndView("InsertDepartment","department",department);
        return mv;
    }

    @RequestMapping("InsertDepartment")
    public ModelAndView insertDepartment(@ModelAttribute("department") Department
dept) {

        ModelAndView mv = new ModelAndView("result","department",dept);
        return mv;
    }

}
```
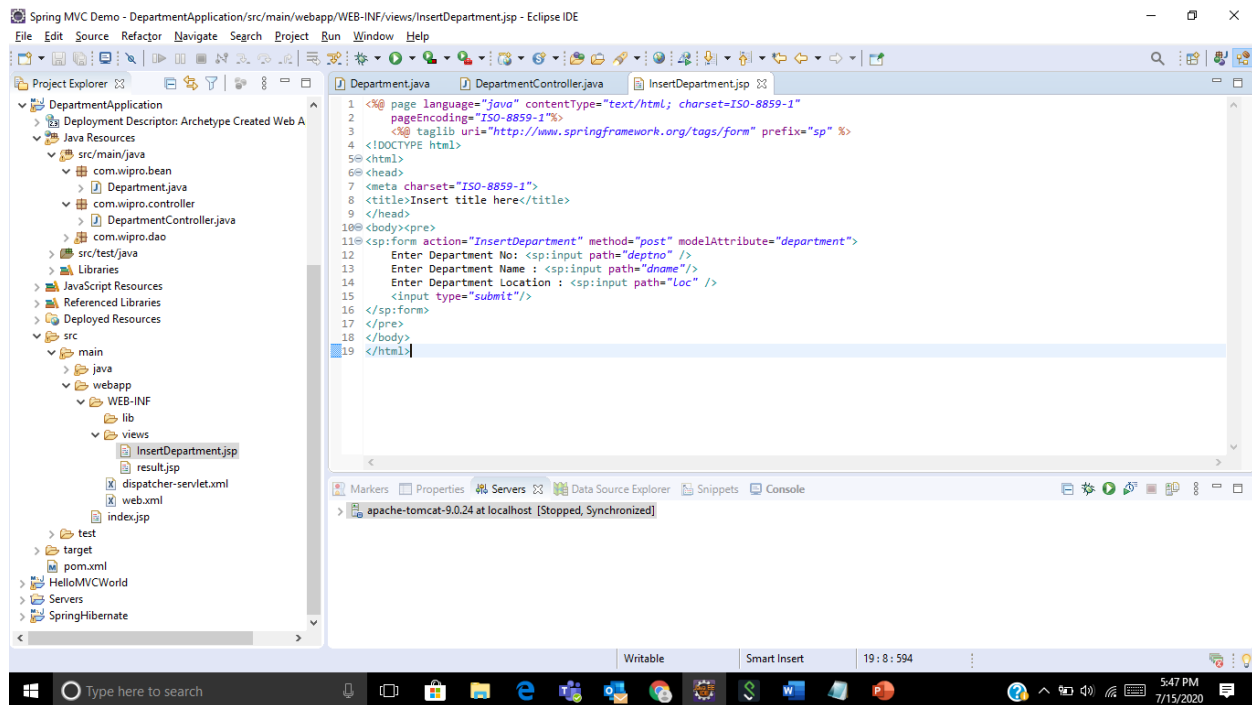
12. Create a new jsp page under WEB-INF -> views called InsertDepartment.jsp

```jsp
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@ taglib uri="http://www.springframework.org/tags/form" prefix="sp" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body><pre>
<sp:form action="InsertDepartment" method="post" modelAttribute="department">
    Enter Department No: <sp:input path="deptno" />
    Enter Department Name : <sp:input path="dname"/>
    Enter Department Location : <sp:input path="loc" />
    <input type="submit"/>
</sp:form>
</pre>
</body>
</html>
```

Create a new jsp page under WEB-INF -> views called result.jsp

```
<%@ page language="java" contentType="text/html; charset=ISO-8859-1"
    pageEncoding="ISO-8859-1"%>
    <%@ page isELIgnored="false" %>
<!DOCTYPE html>
<html>
<head>
<meta charset="ISO-8859-1">
<title>Insert title here</title>
</head>
<body>
<pre>
${msg}
Department [ ${department.deptno } , ${department.dname } , ${department.loc } ]

</pre>
</body>
</html>
```

13. Run the Application from index.jsp