

JSF - USING DATAMODEL IN A DATATABLE

http://www.tutorialspoint.com/jsf/jsf_using_datamodel.htm

Copyright © tutorialspoint.com

In this article, we'll showcase use of datamodel in datatable

Example Application

Let us create a test JSF application to test the above functionality.

Step	Description
1	Create a project with a name <i>helloworld</i> under a package <i>com.tutorialspoint.test</i> as explained in the <i>JSF - Display DataTable</i> sub-chapter of <i>JSF - Data Tables</i> chapter.
2	Modify <i>UserData.java</i> as explained below.
2	Modify <i>home.xhtml</i> as explained below. Keep rest of the files unchanged.
3	Compile and run the application to make sure business logic is working as per the requirements.
4	Finally, build the application in the form of war file and deploy it in Apache Tomcat Webserver.
5	Launch your web application using appropriate URL as explained below in the last step.

UserData.java

```
package com.tutorialspoint.test;

import java.io.Serializable;

import javax.faces.bean.ManagedBean;
import javax.faces.bean.SessionScoped;
import javax.faces.model.ArrayDataModel;
import javax.faces.model.DataModel;

@ManagedBean(name = "userData", eager = true)
@SessionScoped
public class UserData implements Serializable {

    private static final long serialVersionUID = 1L;

    private static final Employee[] employees = new Employee[] {
        new Employee("John", "Marketing", 30,2000.00),
        new Employee("Robert", "Marketing", 35,3000.00),
        new Employee("Mark", "Sales", 25,2500.00),
        new Employee("Chris", "Marketing", 33,2500.00),
        new Employee("Peter", "Customer Care", 20,1500.00)
    };

    private DataModel<Employee> employeeDataModel
    = new ArrayDataModel<Employee>(employees);

    public DataModel<Employee> getEmployees() {
        return employeeDataModel;
    }
}
```

home.xhtml

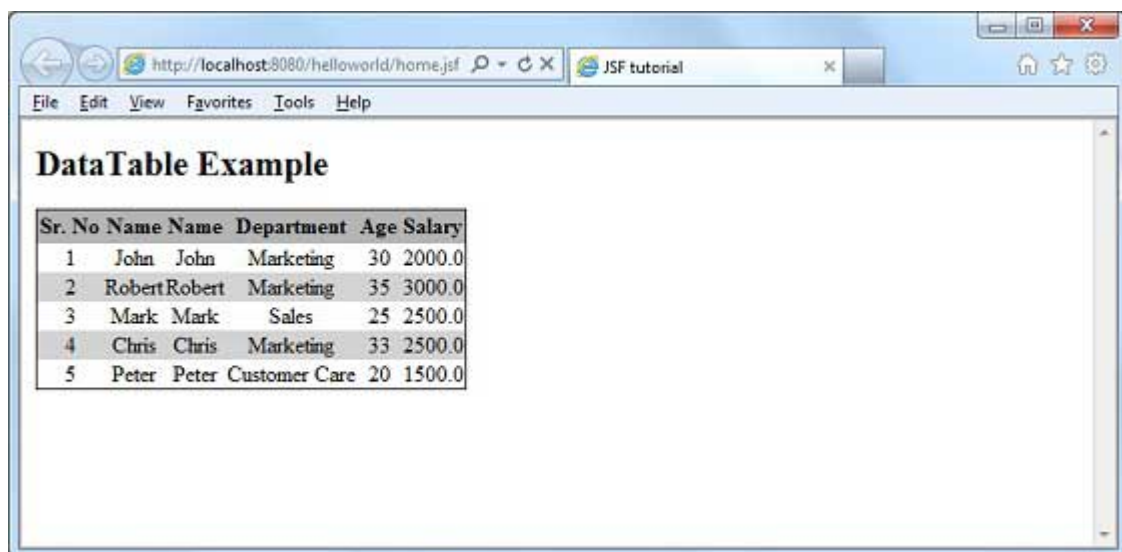
```
<?xml version="1.0" encoding="UTF-8"?>
```

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
xmlns:h="http://java.sun.com/jsf/html"
xmlns:f="http://java.sun.com/jsf/core">
  <h:head>
    <title>JSF tutorial</title>
    <h:outputStylesheet library="css" name="styles.css" />
  </h:head>
  <h:body>
    <h2>DataTable Example</h2>
    <h:form>
      <h:dataTable value="#{userData.employees}" var="employee"
        styleClass="employeeTable"
        headerClass="employeeTableHeader"
        rowClasses="employeeTableOddRow,employeeTableEvenRow">
        <h:column>
          <f:facet name="header">Sr. No</f:facet>
          #{userData.employees.rowIndex + 1}
        </h:column>
        <h:column>
          <f:facet name="header">Name</f:facet>
          #{employee.name}
        </h:column>
        <h:column>
          <f:facet name="header">Department</f:facet>
          #{employee.department}
        </h:column>
        <h:column>
          <f:facet name="header">Age</f:facet>
          #{employee.age}
        </h:column>
        <h:column>
          <f:facet name="header">Salary</f:facet>
          #{employee.salary}
        </h:column>
      </h:dataTable>
    </h:form>
  </h:body>
</html>

```

Once you are ready with all the changes done, let us compile and run the application as we did in JSF - First Application chapter. If everything is fine with your application, this will produce following result:



Sr. No	Name	Name	Department	Age	Salary
1	John	John	Marketing	30	2000.0
2	Robert	Robert	Marketing	35	3000.0
3	Mark	Mark	Sales	25	2500.0
4	Chris	Chris	Marketing	33	2500.0
5	Peter	Peter	Customer Care	20	1500.0