JAVASERVER FACES COURSE

EXERCISE

HANDLING OF COMPONENTS IN JSF
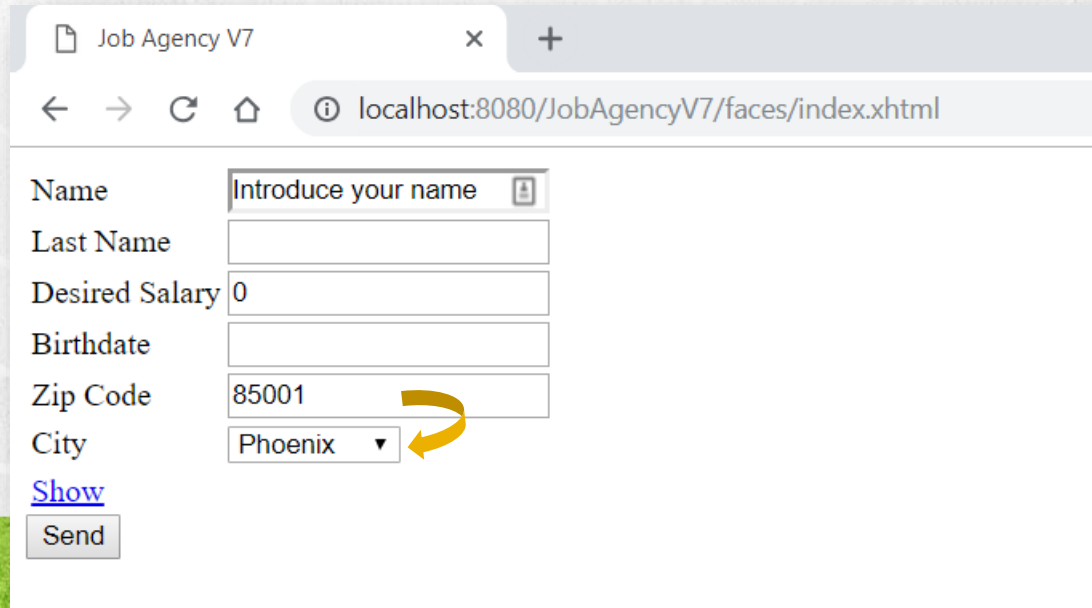
EXERCISE OBJECTIVE

The objective of this exercise is to implement some of the Components of those available in JSF. We will convert the City field from h: inputText to h: selectOneMenu. The result will be similar to the following:
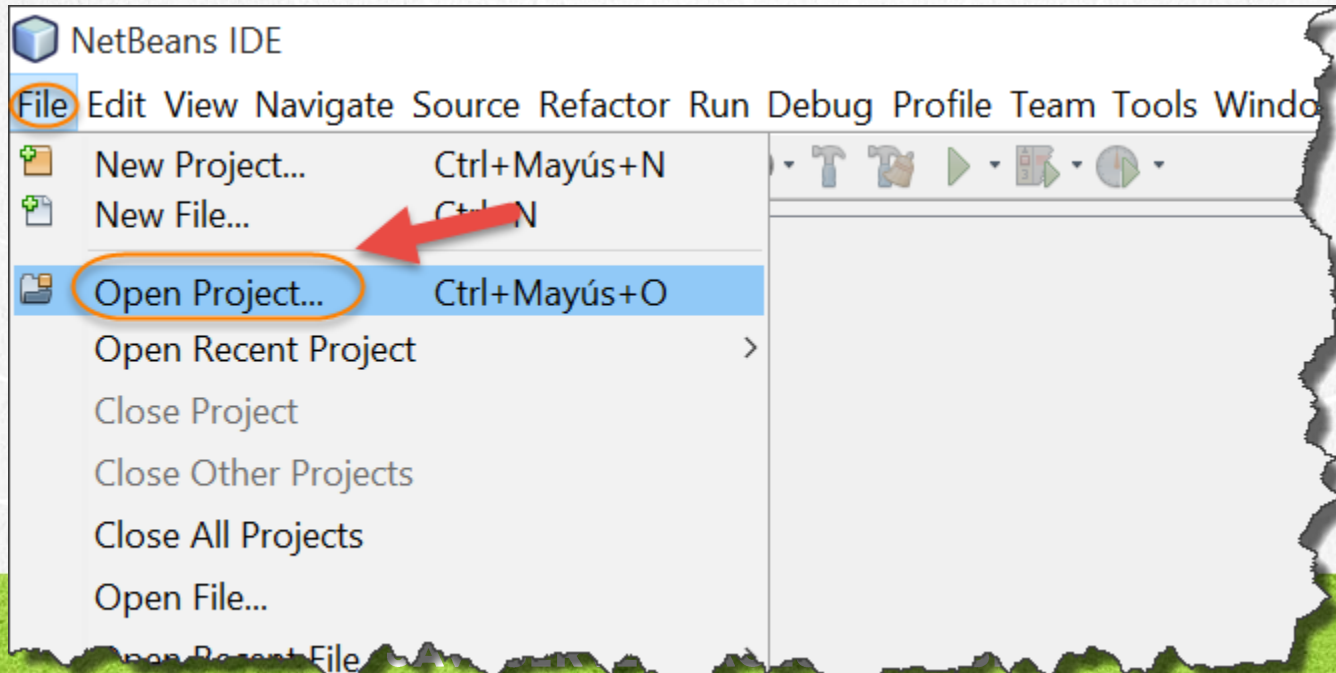
1. OPEN THE PROJECT

We open the JobAgencyV6 project, just in case we do not have it open:

1. OPEN THE PROJECT

We open the JobAgencyV6 project, just in case we do not have it open:

2. COPY THE PROJECT

We copy the JobAgencyV6 project:
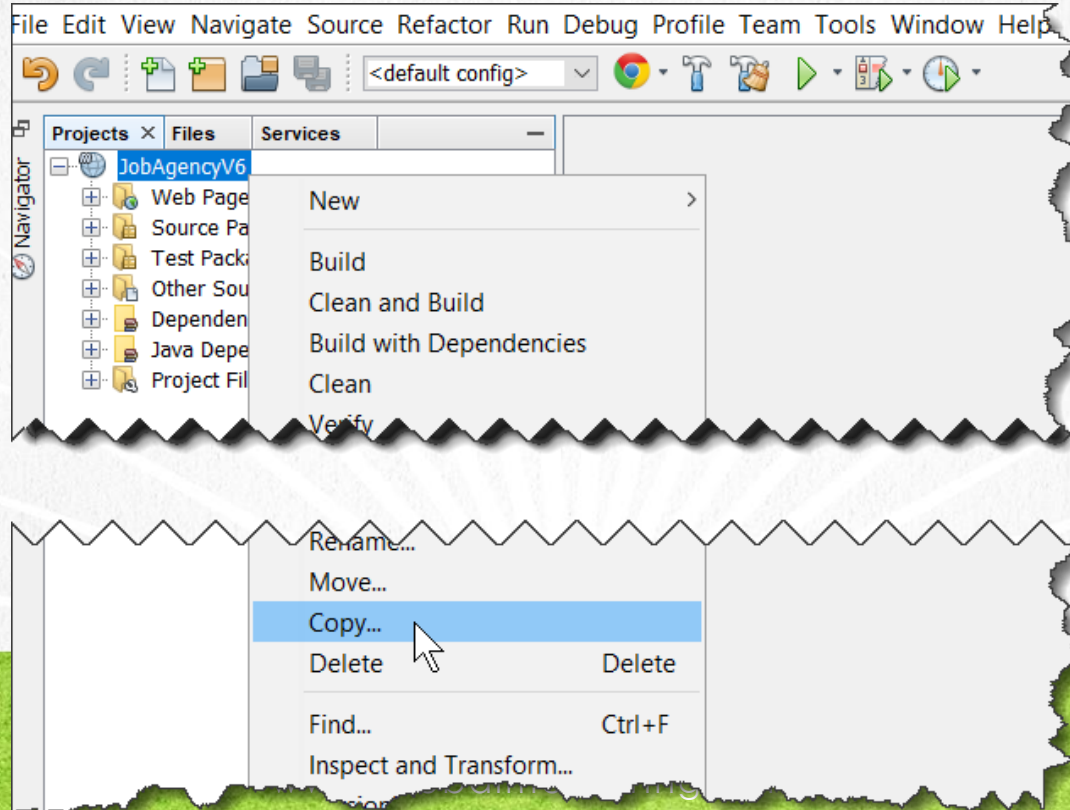
2. COPY THE PROJECT

We copy the JobAgencyV6 project, we rename it to JobAgencyV7 and deposit it in the new path shown:
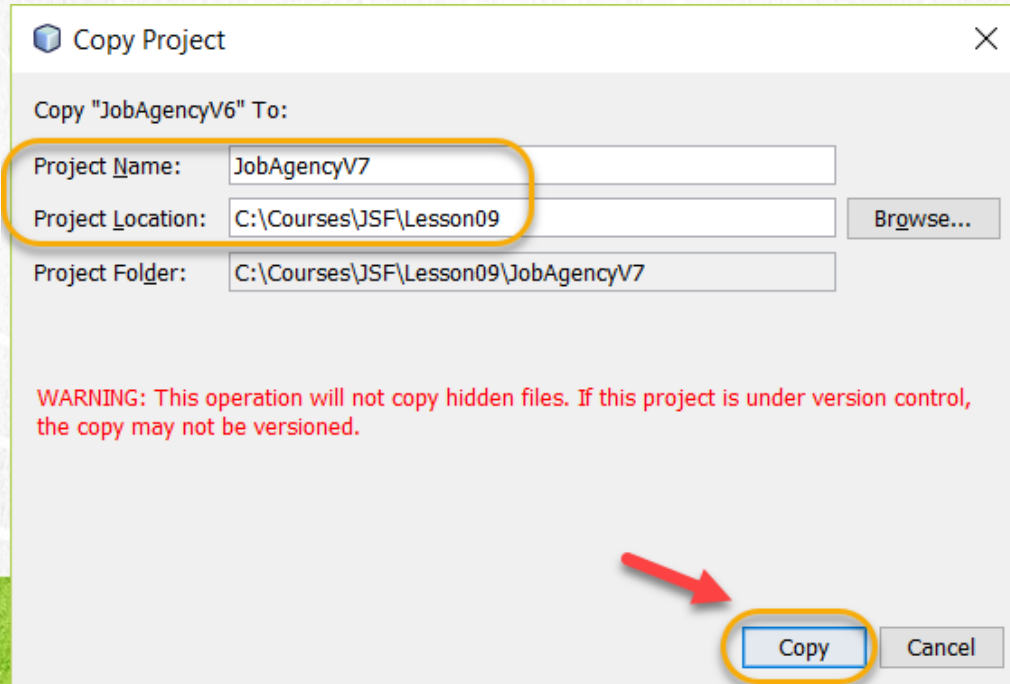
3. CLOSE THE PROJECT

We closed the previous project, and left only the new one open:
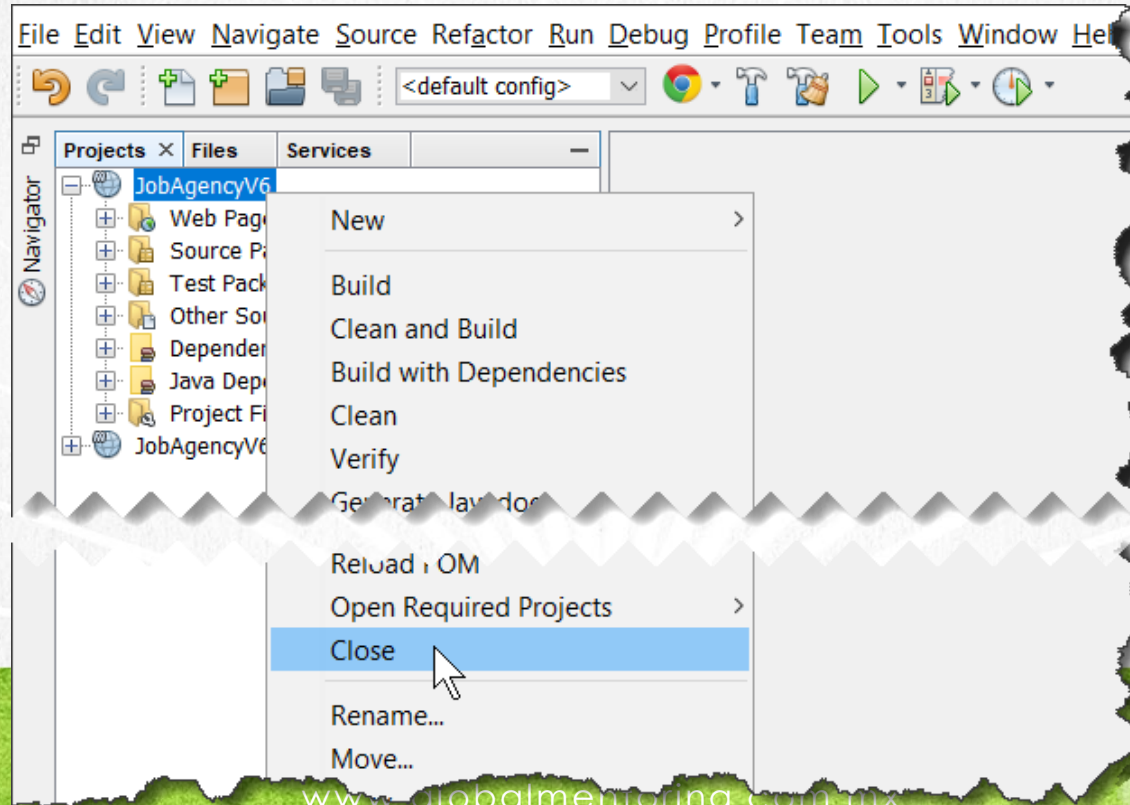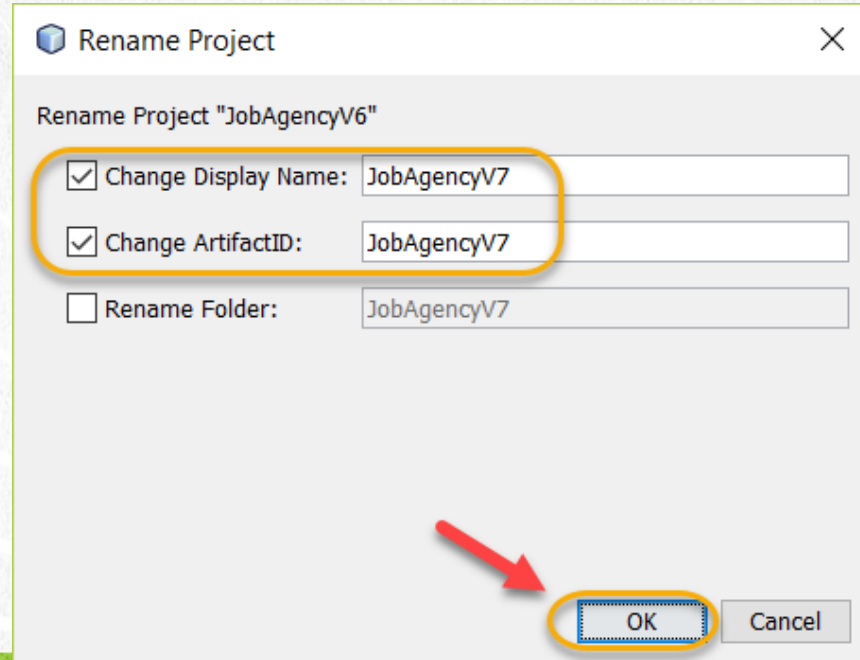
4. RENAME THE PROJECT

Change the name of the project to JobAgencyV7:

4. RENAME THE PROJECT

Change the name of the project to JobAgencyV7:

5. MODIFY THE JAVA CLASS

Open the Candidate.java class and perform the following actions :

```
Remove the city property
Remove the getCity () method
Remove the setCity () method
```

Add the following property:

```
private long cityId;
```

Generate the getter and setter methods for this new property

5. MODIFY THE CODE

Click to download

```java
package beans.model;

import java.util.Date;
import javax.inject.Named;
import javax.enterprise.context.RequestScoped;
import org.apache.logging.log4j.*;

@Named
@RequestScoped
public class Candidate {

    Logger log = LogManager.getRootLogger();

    private String name;
    private String lastName;
    private int desiredSalary;
    private Date birthdate;
    private String zipCode;
    private String cityId;
    private String comment;

    public Candidate() {
        log.info("Creating the Candidate object");
        this.setName("Introduce your name");
    }

    public String getName() {
        return name;
    }
```

5. MODIFY THE CODE

Candidate.java:

```java
public void setName(String name) {
    this.name = name;
}

public String getLastName() {
    return lastName;
}

public void setLastName(String lastName) {
    this.lastName = lastName;
}

public int getDesiredSalary() {
    return desiredSalary;
}

public void setDesiredSalary(int desiredSalary) {
    this.desiredSalary = desiredSalary;
}

public Date getBirthdate() {
    return birthdate;
}

public void setBirthdate(Date birthdate) {
    this.birthdate = birthdate;
}

public String getZipCode() {
    return zipCode;
}
```

5. MODIFY THE CODE

Candidate.java:

```java
    public void setZipCode(String zipCode) {
        this.zipCode = zipCode;
    }

    public String getCityId() {
        return cityId;
    }

    public void setCityId(String cityId) {
        this.cityId = cityId;
    }

    public String getComment() {
        return comment;
    }

    public void setComment(String comment) {
        this.comment = comment;
    }
}
```

6. MODIFY THE PROPERTIES FILE

Open the messages.properties file and make the following changes:

Delete the following field:

```
vacanteForm.city=City
```

Add the new fields

```
vacanteForm.cityId=City
vacanteForm.select=-- Select --
```

Make the same changes in the messages_es.properties file in the corresponding language.

6. MODIFY THE CODE

menssages.properties:

```
vacantForm.name=Name
vacantForm.lastName=Last Name
vacantForm.desiredSalary=Desired Salary
vacantForm.birthdate=Birthdate
vacantForm.zipCode=Zip Code
vacantForm.send=Send
vacantForm.comment=Comment
vacantForm.show=Show
vacantForm.hide=Hide
vacantForm.cityId=City
vacantForm.select=-- Select --
```

7. MODIFY THE CODE

messages_es.properties:

```
vacantForm.name=Nombre
vacantForm.lastName=Apellido
vacantForm.desiredSalary=Sueldo Deseado
vacantForm.birthdate=Fecha de nacimiento
vacantForm.zipCode=Codigo Postal
vacantForm.send=Enviar
vacantForm.comment=Comentario
vacantForm.show=Mostrar
vacantForm.hide=Ocultar
vacantForm.cityId=Ciudad
vacantForm.select=-- Seleccionar --
```

8. MODIFY THE XHTML FILE

Modify the index.xhtml file. We are going to make several changes:

1) Remove the following tag:

```
<td><h:inputText id="city" required="true" value="#{candidate.city}" /></td>
```

2) Insert the following code instead of the one we have removed:

```
<td>
    <h:selectOneMenu id="cityId" required="true" value="#{candidate.cityId}">
        <f:selectItem itemLabel="#{msgs['vacanteForm.select']}" itemValue="" />
        <f:selectItem itemLabel="Phoenix" itemValue="1" />
        <f:selectItem itemLabel="Little Rock" itemValue="2" />
        <f:selectItem itemLabel="Huntsville" itemValue="3" />
    </h:selectOneMenu>
</td>
```

8. MODIFY THE XHTML

Search for h: outputLabel for the cityId field and make the following changes:

Change the h: ouputLabel from:

```
<h:outputLabel for="city" value="#{msgs['vacantForm.city']}"
```

By the following:

```
<h:outputLabel for="cityId" value="#{msgs['vacantForm.cityId']}" />
```

8. MODIFY THE XHTML FILE

Change the h: message from :

```
<h:message for="city" />
```

By the following :

```
<h:message for="cityId" />
```

The index.xhtml file should be as follows:

8. MODIFY THE FILE

Click to download

```html
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:h="http://java.sun.com/jsf/html"
      xmlns:f="http://xmlns.jcp.org/jsf/core">
    <h:head>
        <title>Job Agency V7</title>
    </h:head>
    <h:body>
        <h:form id="vacantForm">

            <h:messages globalOnly="true" ></h:messages>

            <table>
                <tr>
                    <td><h:outputLabel for="name" value="#{msgs['vacantForm.name']}" /></td>
                    <td><h:inputText id="name" required="true" value="#{candidate.name}" /></td>
                    <td><h:message for="name" /></td>
                </tr>
                <tr>
                    <td><h:outputLabel for="lastName" value="#{msgs['vacantForm.lastName']}" /></td>
                    <td><h:inputText id="lastName" required="true" value="#{candidate.lastName}" /></td>
                    <td><h:message for="lastName" /></td>
                </tr>
```

8. MODIFY THE FILE

Click to download

```xml
<tr>
    <td><h:outputLabel for="desiredSalary" value="#{msgs['vacantForm.desiredSalary']}" /></td>
    <td><h:inputText id="desiredSalary" required="true"
                     value="#{candidate.desiredSalary}">
        <f:validateLongRange minimum="500" maximum="5000"/>
    </h:inputText></td>
    <td><h:message for="desiredSalary" /></td>
</tr>
<tr>
    <td><h:outputLabel for="birthdate" value="#{msgs['vacantForm.birthdate']}" /></td>
    <td><h:inputText id="birthdate" required="true"
                     value="#{candidate.birthdate}">
        <f:convertDateTime pattern="MM/dd/yyyy" />
    </h:inputText></td>
    <td><h:message for="birthdate" /></td>
</tr>
<tr>
    <td><h:outputLabel for="zipCode" value="#{msgs['vacantForm.zipCode']}" /></td>
    <td><h:inputText id="zipCode" immediate="true"
                     onchange="this.form.submit()" required="true"
                     value="#{candidate.zipCode}"
                     valueChangeListener="#{vacantForm.zipCodeListener}" /></td>
    <td><h:message for="zipCode" /></td>
</tr>
```

8. MODIFY THE FILE

index.xhtml:

Click to download

```xhtml
        <tr>
            <td><h:outputLabel for="cityId" value="#{msgs['vacantForm.cityId']}" /></td>
            <td>
                <h:selectOneMenu id="cityId" required="true" value="#{candidate.cityId}">
                    <f:selectItem itemLabel="#{msgs['vacantForm.select']}" itemValue="" />
                    <f:selectItem itemLabel="Phoenix" itemValue="1" />
                    <f:selectItem itemLabel="Little Rock" itemValue="2" />
                    <f:selectItem itemLabel="Huntsville" itemValue="3" />
                </h:selectOneMenu>
            </td>

            <td><h:message for="cityId" /></td>
        </tr>
    </table>
    <table>
        <tr>
            <td>
                <h:commandLink actionListener="#{vacantForm.hideComment}"
                              immediate="true" rendered="#{!vacantForm.commentSent}"
                              value="#{msgs['vacantForm.show']}" />
                <h:commandLink actionListener="#{vacantForm.hideComment}"
                              immediate="true" rendered="#{vacantForm.commentSent}"
                              value="#{msgs['vacantForm.hide']}" />
            </td>
        </tr>
    </table>
```

8. MODIFY THE FILE

index.xhtml:

```xml
        <h:panelGroup rendered="#{vacantForm.commentSent}">
            <table>
                <tr>
                    <td><h:outputLabel for="comment" value="#{msgs['vacantForm.comment']}" /></td>
                </tr>
                <tr>
                    <td><h:inputTextarea id="comment" cols="50" rows="10" value="#{candidate.comment}" /></td>
                </tr>
            </table>
        </h:panelGroup>
        <h:commandButton action="#{vacantForm.send}" value="#{msgs['vacantForm.send']}" />
    </h:form>
  </h:body>
</html>
```

9. MODIFY THE JAVA FILE

Modify the VacanteForm.java file.

Replace the code of the zipCodeListener ():

```
UIInput cityInputText = (UIInput) uiViewRoot.findComponent("vacanteForm:city");
String city = "Phoenix";
cityInputText.setValue(city);
cityInputText.setSubmittedValue(city);
```

By the following:

```
UIInput cityInputText = (UIInput) uiViewRoot.findComponent("vacanteForm:cityId");
Long cityId = 1L;
cityInputText.setValue(cityId);
cityInputText.setSubmittedValue(Long.toString(cityId));
```

9. MODIFY THE CODE

VacanteForm.java:

Click to download

```java
package beans.backing;

import beans.model.Candidate;
import javax.inject.*;
import javax.enterprise.context.RequestScoped;
import javax.faces.application.FacesMessage;
import javax.faces.component.*;
import javax.faces.context.FacesContext;
import javax.faces.event.*;
import org.apache.logging.log4j.*;

@Named
@RequestScoped
public class VacantForm {

    Logger log = LogManager.getRootLogger();

    @Inject
    private Candidate candidate;

    private boolean commentSent = false;

    public VacantForm() {
        log.info("Creating VacantForm object");
    }

    public void setCandidate(Candidate candidate) {
        this.candidate = candidate;
    }
```

9. MODIFY THE CODE

VacanteForm.java:

Click to download

```java
public String send() {
    log.info("send() Name=" + this.candidate.getName());
    log.info("send() Desired Salary=" + this.candidate.getDesiredSalary());

    if (this.candidate.getName().equals("John")) {

        if (this.candidate.getLastName().equals("Smith")) {
            String msg = "Thanks, but John Smith is already working with us.";
            FacesMessage facesMessage = new FacesMessage(FacesMessage.SEVERITY_ERROR, msg, msg);
            FacesContext facesContext = FacesContext.getCurrentInstance();
            String componentId = null; //This is a global message
            facesContext.addMessage(componentId, facesMessage);
            return "index";
        }
        return "success";//success.xhtml
    } else {
        return "failure"; //failure.xhtml
    }
}
```

9. MODIFY THE CODE

Click to download

```java
//Method of type Value Change Listener
public void zipCodeListener(ValueChangeEvent valueChangeEvent) {
    FacesContext facesContext = FacesContext.getCurrentInstance();
    UIViewRoot uiViewRoot = facesContext.getViewRoot();
    String newZipCode = (String) valueChangeEvent.getNewValue();
    if ("85001".equals(newZipCode)) {
        log.info("Modify the city value dynamically with ValueChangeListener");
        //We use the name of the form of index.xhtml to find the component
        UIInput cityInputText = (UIInput) uiViewRoot.findComponent("vacantForm:cityId");
        Long cityId = 1L;
        cityInputText.setValue(cityId);
        cityInputText.setSubmittedValue(Long.toString(cityId));
        //Return the response
        facesContext.renderResponse();
    }
}

public void hideComment(ActionEvent actionEvent) {
    this.setCommentSent(!this.commentSent);
    log.info("Executing actionEvent hideComment");

}
```

9. MODIFY THE CODE

VacanteForm.java:
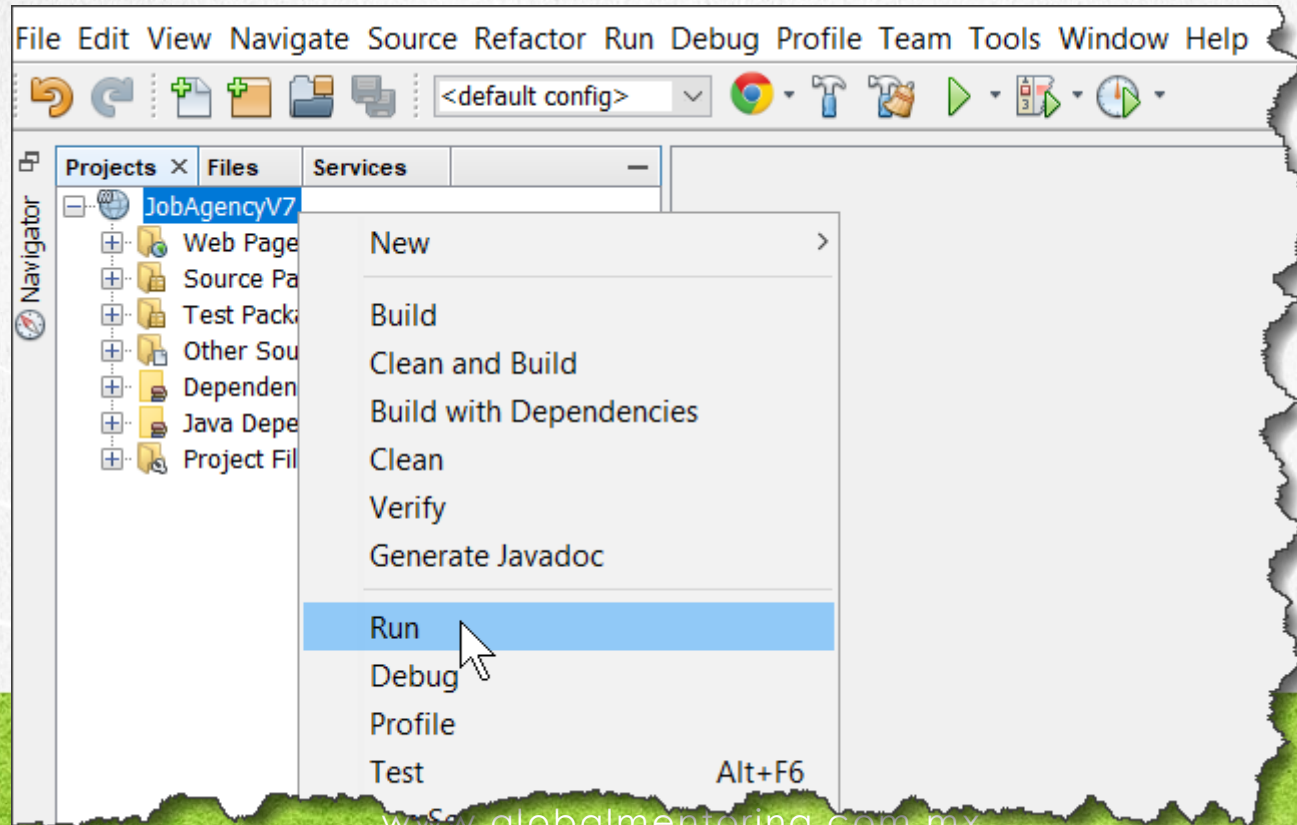
Click to download

```java
    public boolean isCommentSent() {
        return commentSent;
    }

    public void setCommentSent(boolean commentSent) {
        this.commentSent = commentSent;
    }
}
```
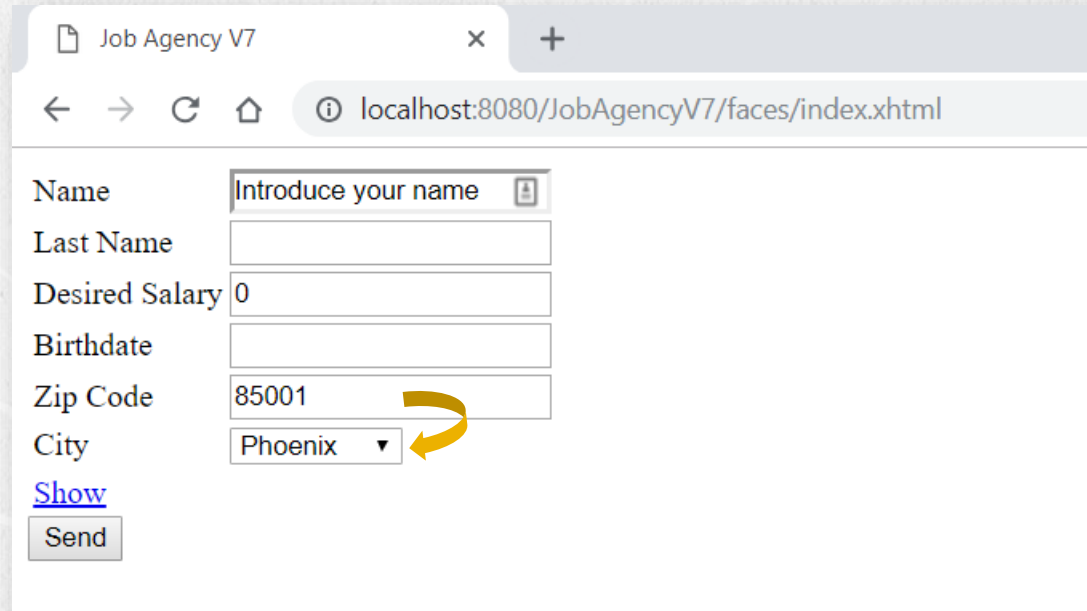
10. EXECUTE THE PROJECT

Execute the project:

10. EXECUTE THE PROJECT

Execute the project:

EXERCISE CONCLUSION

With this exercise we put into practice the concept of components in JSF.

We use the SelectItem component, which will be one of the components that we will use the most in our JSF forms.

One of the great advantages of using JSF components is that many of the components are just as simple to apply or use once having the basics of handling JSF components.

JAVASERVER FACES (JSF)

By: Eng. Ubaldo Acosta

Global Mentoring

JAVA UNIVERSITY