44-563 Web Services Technology Exam 3 Spring 2010

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Part 1 JavaServer Faces \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Here are some tags you might need to use in Part 1 of this exam.

**<h:column>**

**<default-locale>**

**<f:facet>**

**<from-outcome>**

**<from-view-id>**

**<managed-bean>**

**<managed-bean-class>**

**<managed-bean-name>**

**<managed-bean-scope>**

**<h:message>**

**<navigation-case>**

**<navigation-rule>**

**<h:outputText>**

**<resource-bundle>**

**<supported-locale>**

**<to-view-id>**

**<f:validateLongRange>**

1. (10 pts) A navigation rule does the following: If the user is on the page **dog.jsp** and takes an action producing an outcome of **speak**, then the page **woof.jsp** should be displayed. If the user takes an action producing an outcome of **stay**, then the page **sit.jsp** should be displayed. Write the navigation rule that must be included in faces-config.xml. It is not necessary for you to include a description.

**<navigation-rule>**

**<from-view-id>/dog.jsp</from-view-id>**

**<navigation-case>**

**<from-outcome>speak</from-outcome>**

**<to-view-id>/woof.jsp</to-view-id>**

**</navigation-case>**

**<navigation-case>**

**<from-outcome>stay</from-outcome>**

**<to-view-id>/sit.jsp</to-view-id>**

**</navigation-case>**

**</navigation-rule>**

2. (2 pts) In the code below

**<h:outputText**

**value="#{UserBean.studentGPA}"**

**/>**

The expression #{UserBean.studentGPA} invokes what method?

A. studentGPA()

B. getStudentGPA ()

C. setStudentGPA ()

3. (3 pts) Rewrite the code below so that the user is required to enter a value for this field. You do NOT need to write code to display an error message.

**<h:inputText id="payments"**

**value="#{CarBean.numberOfPayments}"**

**required="true">**

**</h:inputText>**

4. (4 pts) Rewrite the code in Problem 3 to validate that the user enters a value between 12 and 48 (inclusive). You do NOT need to write code to display an error message.

**<h:inputText id="payments"**

**value="#{CarBean.numberOfPayments}">**

**<f:validateLongRange**

**minimum="12"**

**maximum="48" />**

**</h:inputText>**

5. (10 pts) Assume we want to use a managed bean in a JSF application. The bean has the name **Pinto**, it is defined in the file **Legume.java**, and it has **request** scope. Write the code that must be included in faces-config.xml.

**<managed-bean>**

**<managed-bean-name>Pinto</managed-bean-name>**

**<managed-bean-class>Legume</managed-bean-class>**

**<managed-bean-scope>request</managed-bean-scope>**

**</managed-bean>**

6. (4 pts) Suppose we want to internationalize a web application so that it can display text in either English or French. We have declared a resource bundle in faces-config.xml:

**<application>**

**<resource-bundle>**

**<base-name>**

studentExam.MyText

**</base-name>**

**<var>**examLabels**</var>**

**</resource-bundle>**

**</application>**

**<application>**

**<locale-config>**

**<default-locale>**

en\_US

**</default-locale>**

**<supported-locale>**

fr\_FR

**</supported-locale>**

**</locale-config>**

**</application>**

(a) We need a label **examScore** with text "Your score must be at least eighty."

( "Votre score doit être d'au moins quatre-vingts." in French).

Write the entry that must be included in the file studentExam.MyText.properties.

examScore=Your score must be at least eighty.

Write the entry that must be included in the file studentExam.MyText\_fr\_FR.properties.

examScore= Votre score doit être d'au moins quatre-vingts.

(b) Complete the JSP code below that displays the label.

**<h:outputText value="#{examLabels.examScore}"/>**

7. (10 pts) Assume

* a Computer object has getNumberOfCores() and getMemorySize() methods
* the expression #{Inventory.computers} returns a list of Computer objects.

Complete the code shown below to produce a table listing the number of cores and memory size of each of the computers. The headings for the columns must be "Cores" and "Memory".

**<h:dataTable value="#{Inventory.computers}"**

**var="computer"**

**border="1"**

**style="border:solid 1px">**

**<h:column>**

**<f:facet name="header">**

**<h:outputText value="Cores"/>**

**</f:facet>**

**<h:outputText value="#{computer.numberOfCores}"/>**

**</h:column>**

**<h:column>**

**<f:facet name="header">**

**<h:outputText value="Memory"/>**

**</f:facet>**

**<h:outputText value="#{computer.memorySize}"/>**

**</h:column>**

**</h:dataTable>**

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Part 2 Persistence \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

8. (6 pts) The following code is from an entity class that was generated by Netbeans.

@Entity

@Table(name = "USERS")

@NamedQueries({@NamedQuery(name = "Users.findAll", query =

"SELECT u FROM Users u"),

@NamedQuery(name = "Users.findByUsername", query =

"SELECT u FROM Users u WHERE u.username = :username"),

@NamedQuery(name = "Users.findByPassword", query =

"SELECT u FROM Users u WHERE u.password = :password"),

@NamedQuery(name = "Users.findByEmailAddress", query =

"SELECT u FROM Users u WHERE u.emailAddress = :emailAddress"),

@NamedQuery(name = "Users.findByUserId", query =

"SELECT u FROM Users u WHERE u.userId = :userId")})

public class Users implements Serializable

{

private static final long serialVersionUID = 1L;

@Column(name = "USERNAME")

private String username;

@Column(name = "PASSWORD")

private String password;

@Column(name = "EMAIL\_ADDRESS")

private String emailAddress;

@Id

@GeneratedValue(strategy = GenerationType.IDENTITY)

@Basic(optional = false)

@Column(name = "USER\_ID")

private Integer userId;

(a) What is the name of the corresponding database table? USERS

(b) How many columns does the database table have? 4

(c) What is the primary key for the table? USER\_ID

9. (8 pts) In the term CRUD,

(a) The C corresponds to the SQL statement

A. DELETE

B. INSERT

C. SELECT

D. UPDATE

(b) R corresponds to the SQL statement

A. DELETE

B. INSERT

C. SELECT

D. UPDATE

(c) U corresponds to the SQL statement

A. DELETE

B. INSERT

C. SELECT

D. UPDATE

(d) D corresponds to the SQL statement

A. DELETE

B. INSERT

C. SELECT

D. UPDATE

10. (8 pts) In a JpaController class, assume em is an instance of an EntityManager. What is the purpose of each of the following expressions?

(a) em.remove(dog);

A. save a new row in the DOG table

B. edit an existing row in the DOG table

C. delete a row in the DOG table

D. retrieve all rows of the DOG table

(b) em.createQuery("select object(o) from Dog as o").getResultList();

A. save a new row in the DOG table

B. edit an existing row in the DOG table

C. delete a row in the DOG table

D. retrieve all rows of the DOG table

(c) em.merge(dog);

A. save a new row in the DOG table

B. edit an existing row in the DOG table

C. delete a row in the DOG table

D. retrieve all rows of the DOG table

(d) em.persist(dog);

A. save a new row in the DOG table

B. edit an existing row in the DOG table

C. delete a row in the DOG table

D. retrieve all rows of the DOG table

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* Part 3 Web Services \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

11. (10 pts) Define a class **Calculator** that implements a web service. The web service has an operation named **inc** that has one int parameter **n**. The operation returns the value n + 1. ***Write the complete class definition***.

**@WebService()**

**public class Calculator**

**{**

**@WebMethod(operationName = "inc")**

**public int inc(@WebParam(name = "n") int n)**

**{**

**return n + 1;**

**}**

**}**