Objective - Business Tier Technologies - Test

Review of attempt 1

Finish review

Started on	Monday, 18 November 2013, 04:38 AM
Completed on	Monday, 18 November 2013, 04:38 AM
Time taken	15 secs
Grade	**1.67** out of a maximum of 24 (**7**%)
Feedback	FAIL

Show **All** / Correct / In-correct

1. A system developed in Java Enterprise Technologies is required for members of a library to facilitate them search and add books which they would like to get issued. Members can search book, add to their kit, edit the kit and remove books if required. Once they are done they can print or email the final list to librarian. What is the best way to implement module to keep track of books for members?

   - a. Stateless session bean ✗
   - b. Stateful session bean ✓
   - c. Message Driven Bean ✗
   - d. Store all data in database and use Core Java ✗

   Choice B is correct. Stateful session beans can maintain client state across multiple calls.

   Choice A is incorrect because stateless session beans cannot maintain client state across method calls.

   Choice C is incorrect because client needs synchronous response.

   Choice D is incorrect because stateful session beans already provide same feature out of the box, custom code is not required.

   Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

   Correct
   Marks for this submission: 1/1.

   Feedback to Author

2. Which of the following statements are true about JMS?

   - a. JMS tightly couples message sender and receiver components ✗
   - b. A Servlet can act as JMS client ✓
   - c. JMS message queue retains the message until it is consumed or until the message expires ✓
   - d. Under point to point messaging, each message has only one consumer ✓

   Choices B, C and D are correct.

   Choice A is incorrect because JMS is a loosely coupled technology in the sense that it does not require sender and receiver of the message to know about each other's API

   Choice B is correct, any Java Enterprise edition component can act as JMS client

   Choices C and D are correct as these are the basic characteristics of point to point messaging

   Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

Partially correct
Marks for this submission: 0.67/1.

Feedback to Author

3. Which of the following are disadvantages of ORM Frameworks?

- a. Usage of ORM Frameworks requires complex coding. ✗
- b. ORM Frameworks reduce development times ✗
- c. Manageability of application decreases ✗
- d. Requires specialized resources. ✓

Choice D is correct.

Option A is incorrect as the statement is wrong.

Option B is incorrect as it is in fact an advantage but not a disadvantage.

Option C is incorrect as the inverse is true.

Option D is correct as you require developers who are skilled in particular framework or accommodate for learning curve.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

4. Which of the following is NOT true about ORM Frameworks?

- a. Application can be isolated from the database and the only connection to it is using the ORM framework and a JDBC driver. ✗
- b. ORM Frameworks reduce development times ✗
- c. Manageability of application increases ✗
- d. It is faster than the direct JDBC access. ✓

Options A, B and C are incorrect as they are true statements about ORM frameworks.

Option D is correct as it is an incorrect statement. ORM induces additional layer in application which implies some performance loss over direct JDBC access.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

5. You are an architect of a web based project for a company that publishes Magazines. The company needs a system for its members to let them select and email articles, they like, to their friends. Sometimes the number of such requests can be large and hence performance is a concern. You decide to implement this feature using:

- a. Plain Java Servlets to keep the code light and use a faster processor to email at better rate ✗
- b. JMS because emails can be queued and processed asynchronously while users can continue using the site ✓
- c. Custom multithreaded program to send multiple emails in parallel ✗
- d. Session beans to synchronously send the emails ✗

Choice B is correct because JMS is best to process jobs that can be handled asynchronously while letting the users continue using the system

Choice A, C and D are incorrect because number of emails to send varies and implementing a synchronous system will always keep performance a concern

Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

Incorrect
Marks for this submission: 0/1.

Feedback to Author

6. Which of the following is NOT an advantage of Java EE Web Services Programming Model?

- a. Supports resource injection removing the burden of creating and initializing common resources in a Java runtime environment. ✗
- b. Simplifies development/deployment of web services through annotations. ✗
- c. Supports both SOAP 1.1 and SOAP 1.2 ✗
- d. A JAX-WS client can access a web service that is not running on the Java platform, and vice versa. ✗
- e. None of the above. ✓

Option E is correct.

All the above are advantages of Java EE WebServices Programming Model.

Features of Java EE WebServices Programming Model :-

- JAX-WS introduces support for annotating Java classes with metadata to indicate that the Java class is a Web service. Using annotations within the Java class simplifies development of Web services.
- JAX-WS supports resource injection of Java EE 5 to shift the burden of creating and initializing common resources in a Java runtime environment.
- Supports both SOAP 1.1 and SOAP 1.2
- JAX-WS 2.0 supports the Web Services Interoperability (WS-I) Basic Profile Version 1.1.

The WS-I Basic Profile is a document that clarifies the SOAP 1.1 andWSDL 1.1 specifications to promote SOAP interoperability.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

7. What are two benefits of the new EJB 3.0 model? Select two choices.

- a. Session bean is a POJO managed by the EJB container. ✓
- b. Stateless session beans require a home interface only. ✗
- c. Callback methods can be defined either in the bean class itself or in a bean listener class. ✓
- d. All session beans and message driven beans need to have a business interface ✗

Options A and C are correct.

EJB 3.0 beans do not need to implement SessionBean or EntityBean interfaces and implement all the callback methods. Also, they do not need to provide home and component interfaces.

Option B is incorrect because stateless session beans do not require a home interface.

Option D is incorrect because message-driven beans do not have any client interface.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

8. Which of the following is true about EJB3.0? Select two choices.

- a. Elimination of lifecycle callback methods. ✗
- b. An interceptor facility for session beans and message-driven beans. ✓
- c. Increased requirements for usage of checked exceptions. ✗
- d. Now you can use standard SQL queries with Java Persistence API. ✓

Option A is incorrect as it is an incorrect statement.

EJB3.0 still provides callback methods but removes the requirement of the same.

Option C is incorrect. It infact reduces requirements for checked exceptions like CreateException and FinderException.

Option D is correct. EJB3.0 provides as query language for Java Persistence that is an extension to EJB QL, with addition of projection, explicit inner and outer join operations, bulk update and delete, subqueries, and group-by. Also supports dynamic query capability and native SQL queries.

Option B is correct. An interceptor is a method that intercepts a business method invocation or a lifecycle callback event.An interceptor method may be defined on the bean class or on an interceptor class associated with the bean. An interceptor class is a class (distinct from the bean class itself) whose methods are invoked in response to business method invocations and/or lifecycle events on the bean class.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

9. Current website of your company is using Java enterprise technology. The website has used CMP beans heavily. Now the company needs another small non-web application for in-house team accessing number of model objects from the company website system. They want to reuse the existing code as as to control cost and improve performance. You should suggest them to move all CMPs to JPA. Why?

- a. JPA entities can be used outside container environment hence promoting re-usability and coding effort ✓
- b. JPA entities can be used on both presentation layer and business logic layer, thus reducing coding effort, size of code and maintenance cost associated ✓
- c. JPA supports lazy loading of objects hence improving performance ✓
- d. Clients currently accessing CMP beans will remain the same to work with JPA ✗

Choice A, B and C are correct

Choice A is correct because JPA can be used outside container environment for non-web project also. Therefore common JPA based API to manage entities will be reusable in multiple projects hence saving duplicate coding effort.

Choice B is correct because JPA entities can hold data and be passed to presentation layer again promoting re-usability and reducing coding effort.

Choice C is correct because JPA does support lazy loading. Objects can be loaded when required instead of always keeping them in memory.

Choice D is incorrect because client code to call a JPA entity is different from that to access a CMP bean. Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

Incorrect
Marks for this submission: 0/1.

Feedback to Author

10. What is the biggest difference between Message-Driven Beans and Entity and Session Beans? Select two choices.

- a. There is no difference in structure between Message-Driven Beans and Entity and Session Beans. They all have home, remote interfaces and a bean class. ✗
- b. Message-Driven beans don't have a home interface. ✓
- c. A Message Driven bean is different from Session Bean because its state is persisted. ✗
- d. Clients do not access Message-Driven beans through interfaces. ✓

Choices B and D are correct.

Note: This type of question will not be in the EJB 1.1 version of the SCEA but is likely to be in the EJB 2.0 version of the SCEA.

Choice A is incorrect, Message-Driven beans do not have Home or Remote interfaces, they just have a bean class.

The state of Message-Driven beans is not persisted like Entity beans. So choice C is incorrect.

The following is taken from: http://java.sun.com/j2ee/tutorial/1_3-fcs/doc/EJBConcepts5.html

The most visible difference between message-driven beans and session and entity beans is that clients do not access message-driven beans through interfaces. Unlike a session or entity bean, a message-driven bean has only a bean class.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

11. What are two reasons why you would use an ORM approach? Select two choices.

- a. Application can be isolated from the database and the only connection to it is using the ORM framework and a JDBC driver. ✓
- b. Manageability of a Java application increases. ✓
- c. They perform faster than direct JDBC connections. ✗
- d. ORM supports same level of data access as SQL. ✗
- e. ORM does little to improve developer productivity. ✗

Options A and B are correct.

ORM not only isolates the application from database but also improves manageability by developers as it generates code. Option C is incorrect.

ORM induces additional layer in application which implies some performance loss over direct JDBC access. Option D is an incorrect statement.

ORM provides good data access features but do not provide capabilities same as SQL.

Option E is wrong because they improve developer productivity.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

12. You are building a new Customer Relationship Management system. You have developed the design model and planning to choose persistence mechanism. The model has complex relationships between tables. You do not want any persistence mechanism code to be present in business objects.

Which of the following technologies would you choose?

- a. Java Persistence API ✓
- b. CMP entity bean ✗
- c. BMP entity bean ✗
- d. DAO ✗

Option A is correct.

JPA supports the requirement, particularly does not require any persistence code in Business Objects.

Option B is incorrect. CMP supports requirements but has persistence mechanism related code in the objects (ejbCreate methods, must implement EntityBean interface).

Options C and D are incorrect as it requires coding.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

13. You are building a simple application which will read XML repositories and display the Web Service details on the browser. In addition, application tries to validate the XML file received from a third party vendor.

Which of the following APIs would you use for each scenario described?

- A. JAXR, JAXP ✓

Option A is correct.

The Java API for XML Registries (JAXR) provides a uniform and standard Java API for accessing different kinds of XML registries. An XML registry is an enabling infrastructure for building, deploying, and discovering web services. The JAXP validation API decouples the validation of an instance document from the parsing of an XML document. This is advantageous for several reasons, some of which are:

Support for additional schema languages. As of JDK 1.5, the two most popular JAXP parser implementations, Crimson and Xerces, only support a subset of the available XML schema languages. The Validation API provides a standard mechanism through which applications may take of advantage of specialization validation libraries which support additional schema languages.
Easy runtime coupling of an XML instance and schema.

Options B, C and D are incorrect. The requirement clearly specifies that the xml repositories and validation of XML is needed from a third party vendor. The supported APIs are JAXR and JAXP only.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

14. You are developing an online ticketing system that requires standard data access technology to interact with Oracle Base. Part of another requirement, Application wants to store the state of saved searches for that user during his conversation between the screens?
What is your design strategy to implement these requirements in Java EE platform?

- A. Entity Bean, Stateless Session Bean ✗
- B. DAO and Stateless Session Bean ✗
- C. JPA and Stateful Session Bean ✓
- D. DAO and Stateful Session Bean ✗

Option C is correct.

Java Persistence API is now the standard API for persistence and object/relational mapping for the Java EE platform. Other requirement is that, state of saved searches between the conversations (state represents the interaction between the bean and a specific client) are done by Stateful Session Bean.

Option A is incorrect. The Entity Beans are replaced with JPA in EJB3.0 and is not the standard API for data access in Java EE platform. The state of saved searched between the conversations can not be maintained using Stateless Session Bean. Stateless Session Bean can not keep track of each client invocation.

Option B is incorrect. DAOs are Data Access Object separates the client interface from its client access mechanism and is not a standard part of Java EE. The state of saved searches between the conversations can not be maintained using Stateless Session Bean. Stateless Session Bean can not keep track of each client invocation.

Option D is incorrect. DAOs are Data Access Object separates the client interface from its client access mechanism and is not a standard part of Java EE. The state of saved searches between the conversations can be maintained using Stateful Session Bean.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

15. Which of the following is FALSE about EJB3.0?

- a. Supports annotations ✗
- b. Supports resource injection ✗
- c. Simplifies enterprise bean types ✗
- d. Gives interceptor facility for session, message-driven and entity Beans. ✓

Option D is correct as interceptor facility is provided only for session and message-driven beans but not entity Beans.

Some of the EJB3.0 changes are:

- Definition of the Java language metadata annotations that can be used to annotate EJB applications. These metadata annotations are targeted at simplifying the developer's task, at reducing the number of program

classes and interfaces the developer is required to implement, and at eliminating the need for the developer to provide an EJB deployment descriptor.
- Encapsulation of environmental dependencies and JNDI access through the use of annotations, dependency injection mechanisms, and simple lookup mechanisms.
- Simplification of the enterprise bean types.
- An interceptor facility for session beans and message-driven beans. An interceptor is a method that intercepts a business method invocation or a lifecycle callback event. An interceptor method may be defined on the bean class or on an interceptor class associated with the bean. An interceptor class is a class (distinct from the bean class itself) whose methods are invoked in response to business method invocations and/or lifecycle events on the bean class.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

16. You are designing a business component as a Stateless Session Bean and decided to configure pool of beans in application server to server multiple user requests. What is true about bean pooling?

- A. Improves memory ✗
- B. Improves Scalability ✓
- C. Improves Security ✗
- D. Bean pooling can not be done using Stateless Session Bean ✗

Option B is correct.
Bean pooling is the technique used when there is a need for more users to use system with fewer resources and improves application scalability.

Option A, C and D are incorrect. With the usage of bean pooling there is nothing you can improve on security and memory usage of an application. Bean pooling is possible for Stateless Session Bean to improve Scalability.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

17. Your project supports dedicated server side resources per client. Here "Client" can be a web application or a desktop application. On business tier, state of client's interaction with server is maintained. What is the best way to implement it?

- a. Keep data in HTTPSession object ✗
- b. Persist data on some storage media although performance will be a downside ✗
- c. Use stateless session beans ✗
- d. Use stateful session beans ✓

Choice D is correct because stateful session bean is a business tier components that is dedicated per client and maintains conversational state.

Choice A is incorrect because business tier data should not be kept on web tier which will overload web server. Secondly HTTPSession object may not be available as it can be a non-web application.

Choice B is incorrect as custom code to persist and retrieve data per client is time consuming, increases maintenance effort and is error prone.

Choice C is incorrect as stateless session beans cannot maintain client state

Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

Incorrect
Marks for this submission: 0/1.

Feedback to Author

18. A store, with several departments, needs a system to generate business related messages. These messages will be accepted by one or more departments based on relevancy of the message subject for the respective department. You, as the architect of this project, will suggest which combination of technologies?

- a. JMS topics with message driven beans ✓
- b. JMS queues with message driven beans ✗
- c. RMI and stateless session beans ✗
- d. Servlets ✗

Choice A is correct as JMS topics ensure that a single message can be consumed by multiple MDBs

Choice B is incorrect as JMS queues allow only single consumer per message

Choice C and D are incorrect as this will be a custom solution and JMS is already a dedicated solution for this standard problem

Further Reference: The Java EE 5 Tutorial – Third Edition by Eric Jendrock (Publisher – Pearson Education)

Incorrect
Marks for this submission: 0/1.

Feedback to Author

19. In EJB3.0, the entity beans are replaced with Java Persistence API. What are the advantageous of EJB3.0 over EJB2.1 Entity programming model?
Which of the following statements are true about JPA model?

- A. In EJB 3.0, persistent fields must be identified through deployment descriptor. ✗
- B. The persistent state of an entity is represented either by its persistent fields or persistent properties. ✓
- C. For an EJB 3.0 entity, you no longer need to code interfaces such as LocalAddressHome and LocalAddress. ✓
- D. In the Java Persistence API, you do not not need to provide an XML descriptor to specify an entity's primary key. ✓
- E. Java Persistence API is simplified by removing support for complex relationships between Entities. ✗

Options B, C and D are correct.

Java Entity is a POJO class in EJB3.0 but not an regular EJB to implement any Local/Home interfaces. Entities may either use persistent fields or persistent properties.

If the mapping annotations are applied to the entity's instance variables, the entity uses persistent fields.

If the mapping annotations are applied to the entity's getter methods for JavaBeans-style properties, the entity uses persistent properties. You cannot apply mapping annotations to both fields and properties in a single entity.

Simple primary keys use the javax.persistence.Id annotation to denote the primary key property or field. Composite primary keys are denoted using the javax.persistence.EmbeddedId and javax.persistence.Id Class annotations.

Option A is incorrect because this was for EJB 2.1.

In the Java Persistence API, you no longer need to provide a deployment descriptor.

Option E is incorrect as it is incorrect statement. JPA supports complex relationships between Entities.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

20. There is an excellent open-source ORM framework available in the market. Your team suggests using this new framework. Which of the following is the main disadvantage of using this new framework?

- a. Application can be isolated from the database and the only connection to it is using the ORM framework and a JDBC driver. ✗
- b. Developer productivity increases. ✗
- c. Performance will be slower than direct JDBC connections. ✗
- d. Application maintainance may be a problem. ✓

Option D is correct.

Since it is open-source code, the maintainance team may find it hard in case of problems with the ORM Framework code.

Options A, B and C are true for any ORM Frameworks and are not sufficient reasons to discard this new framework.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

21. Which two statements are FALSE about implementing web services utilizing the Java EE programming model? Select two choices.

- a. Supports the processing of XML content, data binding, and the development SOAP based and RESTful Web Services. ✗
- b. Web Services exposed by JAX-WS can only be of type document/literal. ✓
- c. Stateless session beans cannot be exposed as web services using Java EE technology. ✓
- d. JAX-WS greatly simplifies the web service implementation model. ✗

Options A and D are incorrect as they are true statements with respect to Java EE Programming model.

Option B is correct because JAX-WS web services can be either a remote procedure call (RPC) style binding or a document style binding.

Option C is correct because a stateless session bean can be exposed as a web service.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

22. You are planning to design a Stateful Session Bean to maintain the state of the shopping cart items between method invocations.
What is true about Stateful Session Bean?
Choose three options.

- A. The bean's state does not represent the interaction between the bean and a specific client ✗
- B. The bean needs to hold information about the client across method invocations ✓
- C. The bean mediates between the client and the other components of the application, presenting a simplified view to the client. ✓
- D. The bean is short lived business component. ✓

Options B, C and D are correct.

Stateful Session Bean - The state of an object consists of the values of its instance variables. In a stateful session bean, the instance variables represent the state of a unique client-bean session. Because the client interacts ("talks") with its bean, this state is often called the conversational state.

The state is retained for the duration of the client-bean session. If the client removes the bean or terminates, the session ends and the state disappears. This transient nature of the state is not a problem, however, because when the conversation between the client and the bean ends there is no need to retain the state.

Option A is false. The bean's state does represent the interaction between the bean a specific client.

For more information, please refer:
http//docs.oracle.com/javaee/5/tutorial/doc/bnbly.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

23. The development lead instructed the team to handle the timeout notifications in Java EE environment.
What type of enterprise beans support this feature?

- A. Only Stateless Session Bean ✗
- B. Stateless, Stateful and Message Driven Beans ✗
- C. Stateless and Stateful Session Beans ✗
- D. Stateless and Message Driven Beans ✓

Option D is correct.
In Java EE, The TimerService interface provides enterprise bean components with access to the container-provided Timer Service. The EJB Timer Service allows stateless session beans and message-driven beans to be registered for timer callback events at a specified time, after a specified elapsed time, or after a specified interval.

Options A, B and D are incorrect.

For more information, please refer:
http://docs.oracle.com/javaee/5/tutorial/doc/bnboy.html
http://docs.oracle.com/javaee/5/api/javax/ejb/TimerService.html

Incorrect
Marks for this submission: 0/1.

Feedback to Author

24. You are using an EJB3.0 Stateful Session Bean for storing shopping cart information. You would like to invoke a particular method to store state data whenever bean passivates. What is the best way to achieve it?

- a. You must write ejbPassivate() method. Shift the logic to this method. ✗
- b. Mention the method name in the deployment descriptor ✗
- c. Use annotations ✓
- d. There is no such facility in EJB3.0 ✗

Option C is correct.

You can annotate the method @PrePassivate. There is no requirement that method has to be ejbPassivate(). So, option A is incorrect.

Option B is incorrect as there is no such facility.

Option D is an incorrect statement.

Incorrect
Marks for this submission: 0/1.

Feedback to Author

Finish review