

TIBCO BusinessWorks Certified Associates Exam

Tibco TCA-Tibco-BusinessWorks

Version Demo

Total Demo Questions: 7

Total Premium Questions: 77

Buy Premium PDF

<https://passqueen.com>

support@passqueen.com

QUESTION NO: 1

An application is deployed to an AppSpace containing several AppNodes. Which two statements correctly describe application activation settings? (Choose two.)

- A. Single AppNode activation permits only one active application instance in an AppSpace at a time.
- B. Failover AppNodes activation can activate the application on another AppNode after the active AppNode fails.
- C. Multiple AppNodes activation prevents the application from running on more than one AppNode.
- D. Single AppNode activation requires that an AppSpace contain exactly one AppNode.
- E. Failover AppNodes activation starts all configured AppNodes as active instances during normal operation.

ANSWER: A B

Explanation:

With **Single AppNode** activation, only one AppNode runs the active application instance at a time. If the active AppNode becomes unavailable, another eligible AppNode in the AppSpace can activate the application.

Failover AppNodes activation is appropriate when a secondary AppNode must activate the application only after the existing active AppNode fails. Multiple AppNodes activation is used when application instances are intended to run on more than one AppNode during normal operation, not only after a failure.

QUESTION NO: 2

Select the 2 types of modules that are available in TIBCO BusinessWorks. (Choose two.)

- A. Application Module
- B. BusinessWorks Module
- C. Shared Module
- D. Package module

ANSWER: A C

Explanation:

TIBCO BusinessWorks projects are organized using **Application Modules** and **Shared Modules**. An Application Module is the deployable portion of a BusinessWorks application. It contains the resources that implement an application's business logic, such as process definitions, service configurations, and related application-level resources. When an application is built and deployed, its Application Module is packaged as part of the application archive.

A Shared Module is intended for reusable resources that can be referenced by one or more Application Modules. This supports modular design and reduces duplication by allowing common process logic, schemas, connections, and other shared assets to be maintained centrally. Shared Modules are included as dependencies of the Application Modules that use them, rather than being deployed independently as complete applications. This module model enables developers to separate application-specific implementation from reusable functionality while maintaining clear project dependencies. TIBCO's BusinessWorks documentation describes creating both application modules and shared modules as the available module types in a BusinessWorks application project. See the [TIBCO ActiveMatrix BusinessWorks documentation](#) and the [TIBCO Business Studio documentation](#).

QUESTION NO: 3

You need to cancel an online order and ensure the money is returned to the customer. Which method in TIBCO ActiveMatrix BusinessWorks allows you to most effectively manage, execute, or revert a Unit Of Work so that you can achieve this goal?

- A. Define a conversation.
- B. Configure Catch and Rethrow Activities
- C. Define a Repeat on Error activity.
- D. Define an error/fault handler in a scope.

ANSWER: A

Explanation:

Define a conversation. is correct because a BusinessWorks conversation represents a logical unit of work whose lifecycle can be coordinated across related process activities. For an order-cancellation scenario, the conversation provides the context needed to treat cancellation and refund processing as related work: the flow can complete the work when processing succeeds, or invoke the appropriate rollback/compensation behavior when the unit of work cannot be completed. This is particularly useful when the work involves multiple service interactions, such as cancelling an order in an order-management system and issuing a refund through a payment service. A conversation maintains the state and correlation necessary to manage that work as one business-level interaction rather than as unrelated activity executions. The rollback behavior is implemented according to the resources and compensation logic involved; it is not merely an exception notification. In this way, defining a conversation directly addresses the requirement to manage, execute, or revert the order-cancellation unit of work and helps preserve a consistent customer outcome. See the [TIBCO ActiveMatrix BusinessWorks documentation](#) and the [TIBCO BusinessWorks product documentation resources](#) for BusinessWorks process, transaction, and error-handling concepts.

QUESTION NO: 4

What is the default setting for the log level of an AppNode log file?

- A. INFO
- B. |WARN
- C. DEBUG
- D. ERROR

ANSWER: A

Explanation:

INFO is the default log level for an AppNode log file in TIBCO BusinessWorks. This level records normal operational and lifecycle information, such as AppNode startup and shutdown activity, deployment-related events, configuration processing, and other significant runtime messages. It provides useful visibility into normal platform operation without producing the high volume of diagnostic detail that would be generated by a debugging-level configuration. Consequently, an administrator can use the AppNode log as a practical first source when monitoring a node or investigating an operational event.

The logging level can be changed when additional diagnostic detail is required, typically for a limited troubleshooting period. Keeping the default at INFO is a sensible balance between retaining actionable operational records and controlling log-file growth and I/O overhead. AppNode administration, logging configuration, and runtime management are covered in the TIBCO BusinessWorks documentation available through the [TIBCO ActiveMatrix BusinessWorks documentation portal](#). TIBCO product documentation and supported-version resources can also be accessed from the [TIBCO BusinessWorks product page](#).

QUESTION NO: 5

Which two statements are true about dependencies between TIBCO ActiveMatrix BusinessWorks modules? (Choose two.)

- A. An application module can depend on a shared module.
- B. A shared module can depend on another shared module.
- C. A shared module can depend on an application module.
- D. An AppNode can be added as a dependency of an application module.
- E. An AppSpace can be added as a dependency of a shared module.

ANSWER: A B

Explanation:

An application module can depend on a shared module so that it can use reusable resources and public processes exported by that shared module. A shared module can also depend on another shared module, which supports layered reuse of common artifacts.

A shared module cannot depend on an application module because an application module represents application-specific implementation. Application modules and shared modules are assembled into an application for packaging and deployment rather than being deployed as independent dependencies of one another at runtime.

QUESTION NO: 6

A process can receive either an approval event or a rejection event. The process must continue as soon as the first event arrives and must not wait for the other event.

Which group should be used?

- A. Iterate
- B. Repeat on Error
- C. Pick First
- D. Critical Section

ANSWER: C

Explanation:

Pick First is correct because it waits for multiple alternative paths and continues when the first eligible path completes. In this scenario, the approval and rejection events are mutually exclusive outcomes, so the process should proceed with whichever event is received first.

An Iterate group is used to process a collection repeatedly, a Repeat on Error group is used to retry processing after an error, and a Critical Section controls concurrent access to a protected section of a process.

QUESTION NO: 7

A REST service must require HTTP Basic authentication. The team has already created an Identity Provider that validates the users.

Which additional configuration is required?

- A. Enable Authentication on the HTTP Connector Resource and enable Authentication in the Component Configurations.
- B. Enable Confidentiality on the HTTP Connector Resource and remove the Identity Provider.
- C. Create a service reference for the REST service and configure an Invoke activity.

D. Configure only an SSL Server Configuration on the HTTP Connector Resource.

ANSWER: A

Explanation:

The developer must **enable Authentication on the HTTP Connector Resource and enable Authentication in the REST service Component Configurations**. The HTTP Connector enforces authentication for inbound HTTP requests, while the component configuration applies the requirement to the REST service implementation. The Identity Provider supplies the mechanism used to validate the presented credentials.

An SSL Server Configuration is recommended to protect Basic authentication credentials in transit, but SSL alone does not enable authentication. A service reference is used to consume an external service and does not secure an inbound REST service.