



ORDER-MANAGEMENT- ADMINISTRATOR^{Q&As}

Salesforce Order Management Administrator Accredited Professional

**Pass Salesforce ORDER-MANAGEMENT-
ADMINISTRATOR Exam with 100% Guarantee**

Free Download Real Questions & Answers **PDF** and **VCE** file from:

<https://www.pass4itsure.com/order-management-administrator.html>

100% Passing Guarantee
100% Money Back Assurance

Following Questions and Answers are all new published by Salesforce
Official Exam Center



- ⚙️ **Instant Download** After Purchase
- ⚙️ **100% Money Back** Guarantee
- ⚙️ **365 Days** Free Update
- ⚙️ **800,000+** Satisfied Customers



**QUESTION 1**

A company's service team should not see Process Exceptions on any Order Summaries. How can an administrator meet these requirements?

- A. Modify the service team profiles to remove access to the lightning component and all associated Apex classes
- B. Add a filter to the Process Exception component on the Order Summary lightning record page to hide the component for the service team's Profile
- C. Create a copy of the Order Summary lightning record page for the service team and remove the Process Exceptions component. Activate the lightning record page by profile for all Service team profiles.
- D. Modify the Order Summary page layout to remove visibility to the Process Exception component for Service team profiles

Correct Answer: C

The best way for the administrator to meet these requirements is to create a copy of the Order Summary lightning record page for the service team and remove the Process Exceptions component. Activate the lightning record page by profile for all Service team profiles. A Process Exception is a record that represents an error or exception that occurred during the order lifecycle, such as payment authorization failure, inventory allocation failure, fulfillment location assignment failure, etc. A Process Exception has a lookup relationship to the Order object, and it contains information such as the exception type, status, message, etc. The Process Exceptions component is a standard component that displays process exceptions related to an order on its record page. The administrator can use the Lightning App Builder to create a copy of the Order Summary lightning record page for the service team and remove the Process Exceptions component from the page layout. The administrator can then activate the lightning record page by profile for all Service team profiles, so that they do not see Process Exceptions on any Order Summaries.

https://help.salesforce.com/s/articleView?id=sf.order_management_process_exception.htm&type=5

https://help.salesforce.com/s/articleView?id=sf.om_process_exceptions_component.htm&type=5

https://help.salesforce.com/s/articleView?id=sf.lightning_app_builder_assign_lex_pages.htm&type=5

QUESTION 2

Universal Containers (UC) is evaluating Salesforce Order Management for managing its overarching process because their current system is written mostly in Apex code and has proved difficult to modify, deploy and debug. What are three advantages of using Flow Builder vs writing Apex code to manage the main flow of Order data?

- A. It allows for non-coding members of staff to contribute suggestions for optimizations and better overall customer experience
- B. The admin can attach a debugger to live customer sessions
- C. It will notify the admin before a third party integration's data interface has changed
- D. The admin can easily debug specific business cases visually.
- E. Because it is visual it is also self-documenting as changes are made

Correct Answer: BDE

Three advantages of using Flow Builder vs writing Apex code to manage the main flow of Order data are:



The admin can attach a debugger to live customer sessions. This allows the admin to monitor and troubleshoot the flow execution in real time, without affecting the customer experience or data integrity.

The admin can easily debug specific business cases visually. This allows the admin to test the flow with different input values and see how the flow behaves in a graphical interface, without writing any code.

Because it is visual it is also self-documenting as changes are made. This allows the admin to easily understand and maintain the flow logic, as well as track the changes and versions of the flow.

https://help.salesforce.com/s/articleView?id=sf.flow_builder_debug.htm&type=5

https://help.salesforce.com/s/articleView?id=sf.flow_builder.htm&type=5

QUESTION 3

An administrator is running into performance issues due to a high number of records being created in a flow. How can the administrator modify the flow to improve scalability?

- A. Ask a developer to create a Flow apex action to offload the creation of records
- B. Ask a developer to offload all the Flow functionality to Apex code via Triggers and Apex classes
- C. Use the Bulk Create Records node in the Flow to improve performance
- D. Offload the creation of records to a Subflow that will be called in the Reference Flow

Correct Answer: C

The best way for an administrator to modify the flow to improve scalability when running into performance issues due to a high number of records being created in a flow is to use the Bulk Create Records node in the Flow. This node allows the administrator to create multiple records of the same object type in one transaction, which reduces the number of database operations and improves the flow performance.

https://help.salesforce.com/s/articleView?id=sf.flow_ref_elements_data_bulk_create.htm&type=5

QUESTION 4

What are two reasons for using Flows instead of Apex code?

- A. Flows can be modified and created without a developer
- B. Flows have better performance options for large batches of records
- C. Flows can be triggered by undelete events
- D. Flows provide a visual debug process

Correct Answer: AD

Two reasons for using Flows instead of Apex code are:

Flows can be modified and created without a developer. Flows are declarative tools that allow administrators to build complex business logic using clicks instead of code. Flows do not require any programming skills or knowledge, and they



can be easily modified and maintained by administrators. Flows provide a visual debug process. Flows have a built-in debugger that allows administrators to test and troubleshoot their flows in a graphical interface. The debugger shows the

flow execution path, the values of variables and sObject fields, and any errors or warnings that occur in the flow.

https://help.salesforce.com/s/articleView?id=sf.flow_builder.htmandtype=5

https://help.salesforce.com/s/articleView?id=sf.flow_builder_debug.htmandtype=5

QUESTION 5

An admin is working with Order Management licensed as standalone (B2C Commerce is not in the scenario). The admin submits a JSON payload using Workbench and Workbench indicates the operation was successful. The second time the same payload is submitted the operation fails. What are two likely causes?

- A. Stock Keeping Unit values must be updated
- B. The payment data must be updated
- C. The stock levels in the org are depleted from the first order
- D. The Order Reference Number must be updated

Correct Answer: BD

Two likely causes that could make the second submission of the same JSON payload fail are:

The payment data must be updated. The payment data in the JSON payload contains information such as the payment method, amount, and authorization code. If the same payment data is used for the second order, it may cause a duplicate

payment error or an invalid authorization error. The Order Reference Number must be updated. The Order Reference Number is a unique identifier for each order that is generated by the external system and sent in the JSON payload. If the

same Order Reference Number is used for the second order, it may cause a duplicate order error or an invalid order error.

https://help.salesforce.com/s/articleView?id=sf.order_management_order_api.htmandtype=5

https://help.salesforce.com/s/articleView?id=sf.order_management_payment_api.htmandtype=5

[ORDER-MANAGEMENT-ADMINISTRATOR VCE Dumps](#)

[ORDER-MANAGEMENT-ADMINISTRATOR Practice Test](#)

[ORDER-MANAGEMENT-ADMINISTRATOR Study Guide](#)