

Plat-Con-201 exam dumps and Salesforce Plat-Con-201 exam Simulator



Plat-Con-201 exam dumps are so comprehensive that you do not need any other study material. The Plat-Con-201 study material is all-inclusive and contains straightforward questions and answers comprising all the important topics in the actual Plat-Con-201 exam. Plat-Con-201 latest download demo is available for all of you. You can know the exam format and part questions of our Complete Plat-Con-201 Exam Dumps. Besides, we can ensure 100% passing and offer the Money back guarantee when you choose our Plat-Con-201 pdf dumps.

Our Plat-Con-201 exam prep can bring you high quality learning platform to pass the variety of exams. Plat-Con-201 guide dumps are elaborately composed with major questions and answers. Plat-Con-201 test question only needs 20 hours to 30 hours to practice. There is important to get the Plat-Con-201 Certification as you can. There is a fabulous product to prompt the efficiency-- the Plat-Con-201 exam prep, as far as concerned, it can bring you high quality learning platform to pass the variety of exams.

>> Valid Plat-Con-201 Exam Answers <<

Salesforce Plat-Con-201 Examcollection Free Dumps | New Plat-Con-201 Exam Name

Exam Plat-Con-201 tests your professional talent and expertise. This is the reason that passing this Salesforce Consultant certification exam has been a tough challenge for professionals. But it is made easy now to ace it! The recently developed FreeCram's Plat-Con-201 Exam Questions dumps aim at to deliver you the shortest possible route to obtaining Plat-Con-201 without any chance of losing the exam.

Salesforce Certified Omnistudio Consultant Sample Questions (Q58-Q63):

NEW QUESTION # 58

A company is designing a new console for contact center agents to display case data for customers. All cases will be fetched using a single DataRaptor. This page needs to display cases as follows:

* "Open" cases with case description, case open date, case type, assigned to, and priority fields. Open cases should be highlighted with a red border.

* "Awaiting Closure" cases with case description, last action taken date, resolution, approval reason for closure, and assigned to fields. These cases should be highlighted with a grey border.

* "Closed" cases with case description, resolution, case closed date fields with a link to duplicate cases.

How should the consultant design the FlexCard solution to meet these requirements?

- A. Using a single FlexCard with multiple card filters
- **B. Using a single FlexCard with three states**
- C. Using a single FlexCard with three flyouts
- D. Using three FlexCards

Answer: B

Explanation:

Comprehensive and Detailed In-Depth Explanation: The requirement involves designing a FlexCard solution to display case data fetched by a single DataRaptor, with three distinct categories of cases ("Open," "Awaiting Closure," and "Closed"), each with specific fields and unique styling (e.g., red border for Open, grey border for Awaiting Closure). Based on Salesforce OmniStudio capabilities, the most efficient and appropriate design is a single FlexCard with three states.

Here's why B. Using a single FlexCard with three states is the correct answer:

* FlexCard States Overview: In OmniStudio, a FlexCard can have multiple "states," which are conditional views of the card's content and styling based on data conditions. Each state can display different fields, apply unique CSS styles (e.g., borders), and include interactive elements (e.g., links).

States are driven by conditions evaluated against the data retrieved by the DataRaptor, making them ideal for scenarios where the same dataset (in this case, cases from a single DataRaptor) needs to be presented differently based on a field value (e.g., Status).

* Meeting the Requirements:

* Data Source: All cases are fetched via a single DataRaptor, which aligns with using one FlexCard since FlexCards are designed to work with a single data source (e.g., a DataRaptor Extract) and then manipulate or display that data in various ways.

* "Open" Cases: A state can be configured with a condition like Status = 'Open', displaying fields such as CaseDescription, CaseOpenDate, CaseType, AssignedTo, and Priority. Custom CSS can be applied in the state's style settings to add a red border (e.g., border: 2px solid red).

* "Awaiting Closure" Cases: Another state with a condition like Status = 'Awaiting Closure' can show CaseDescription, LastActionTakenDate, Resolution, ApprovalReasonForClosure, and AssignedTo, styled with a grey border (e.g., border: 2px solid grey).

* "Closed" Cases: A third state with Status = 'Closed' can display CaseDescription, Resolution, and CaseClosedDate, with a hyperlink element added for duplicate cases (e.g., linking to a related DuplicateCaseId field). No specific border color is mentioned, so default styling or a custom style could be applied.

* Display Logic: By default, FlexCards in a list format (e.g., a datatable or repeating block) can iterate over the DataRaptor's JSON output, applying the appropriate state to each case record based on its Status. This ensures all cases are displayed dynamically with their respective fields and styles.

* Why One FlexCard?: Using a single FlexCard with states keeps the solution efficient, maintainable, and aligned with OmniStudio best practices. It leverages one DataRaptor call, reduces redundancy, and centralizes the logic and styling in a single component.

Now, let's examine why the other options are incorrect:

* A. Using three FlexCards: While it's technically possible to create three separate FlexCards (one for Open, one for Awaiting Closure, and one for Closed), this approach is inefficient and impractical. Each FlexCard would need its own DataRaptor or a filtered subset of the same DataRaptor output, leading to redundant configurations and multiple queries or data manipulations.

Additionally, displaying all three categories together in a cohesive console view would require embedding them in an OmniScript or custom layout, complicating the design. The requirement implies a unified display of all cases, which a single FlexCard with states handles more naturally.

* C. Using a single FlexCard with multiple card filters: Card filters in FlexCards allow users to dynamically filter the displayed data (e.g., a dropdown to show only Open cases). However, filters don't inherently change the fields displayed or apply specific styling like red or grey borders per category- they only subset the data. To meet the requirement of showing different fields (e.g., Priority for Open vs. LastActionTakenDate for Awaiting Closure) and unique styling, you'd still need states or custom coding beyond what filters provide. Filters are user-driven, not automatic, and don't fully address the requirement.

* D. Using a single FlexCard with three flyouts: Flyouts are pop-up panels in FlexCards that display additional details or actions when a user interacts with an element (e.g., clicking a case row). While flyouts could show extra case details, they don't suit the requirement of displaying all cases with their respective fields and borders directly on the main card. Flyouts are supplementary, not primary displays, and using three flyouts would require user interaction to view each category, which contradicts the need to "display cases as follows" in a visible, highlighted manner.

Best Practice Alignment:

The OmniStudio documentation emphasizes FlexCard states for scenarios where data from a single source needs conditional rendering and styling. This approach minimizes maintenance overhead, optimizes performance (one DataRaptor call), and provides a seamless user experience for contact center agents viewing all case types in one console.

References:

- * Salesforce OmniStudio Documentation: FlexCards Designer - Details how states can conditionally display data and apply custom styles.
- * Salesforce OmniStudio Developer Guide: Working with FlexCard States - Explains configuring states based on field values like Status.
- * Salesforce Help: FlexCard Styling - Covers applying CSS (e.g., borders) to states for visual differentiation.

NEW QUESTION # 59

A consultant designs a FlexCard with five card states following best practices. Four of the card states have a condition. At runtime, if two of the states' condition are true, which state will be displayed?

- A. The state with a true condition that has the highest priority
- B. The first state with the highest priority closest to the top of the canvas
- C. The state closest to the top of the canvas
- D. The first state with a true condition closest to the top of the canvas

Answer: A

Explanation:

The state that will be displayed at runtime if two of the states' condition are true is the state with a true condition that has the highest priority. Priority is a field on the state object that determines the order in which states are evaluated when there are multiple states with true conditions. The lower the priority number, the higher the precedence. The position of the state on the canvas or the order of evaluation does not affect the state selection logic.

NEW QUESTION # 60

Which element retrieves data from a Salesforce picklist in an org?

- A. Select
- B. Mark this item for later review.
- C. Lookup
- D. Calculation Action
- E. DataRaptor Extract Action

Answer: A

Explanation:

In Salesforce OmniStudio, the Select element within an OmniScript is specifically designed to retrieve and display picklist values from a Salesforce object field, allowing users to choose from a predefined list of options. This element is used to create a dropdown or picklist interface in an OmniScript, enabling users to interact with Salesforce data by selecting a single value from the available options.

Here's why Select is the correct answer:

- * The Select element in OmniScript supports multiple methods to populate its options, including retrieving values directly from a Salesforce picklist field. According to the official Salesforce OmniStudio documentation, you can configure the Select element's "Option Source" to "SObject," which allows it to fetch picklist values from a specified Salesforce object and field. For example, if you have a picklist field like Industry on the Account object, the Select element can retrieve all active picklist values (e.g., "Technology," "Healthcare," etc.) and present them as a dropdown to the user.
- * The Select element is highly flexible and supports three option source types:
- * Manual: Manually defined label-value pairs.
- * SObject: Retrieves options from a Salesforce object field (such as a picklist).
- * Custom: Uses an Apex controller for more complex logic. When set to "SObject," it directly queries the Salesforce schema to pull the picklist metadata, ensuring the values reflect what's defined in the org.

Now, let's examine why the other options are incorrect:

- * A. DataRaptor Extract Action: A DataRaptor Extract is a powerful tool in OmniStudio used to retrieve data from Salesforce objects, transform it, and pass it to an OmniScript or FlexCard. While it can retrieve picklist data as part of a broader dataset (e.g., extracting a record with a picklist field value), it is not an "element" within an OmniScript, nor is it specifically designed to display or interact with picklist values in a user interface. Instead, it operates behind the scenes as a data retrieval mechanism.
- The Select element, however, is the UI component that presents those values to the user.
- * B. Lookup: The Lookup element in OmniScript is used to search for and select a record from a Salesforce object based on user input (e.g., finding an Account by name). It does not retrieve or display picklist values from a field; it retrieves entire records. While it

can indirectly involve picklist fields as part of the record data, its primary purpose is record selection, not picklist value retrieval for display.

* D. Calculation Action: A Calculation Action in OmniScript performs mathematical or logical operations based on user input or data (e.g., adding two numbers or concatenating strings). It does not retrieve data from Salesforce picklist fields or interact with them directly. Its role is computation, not data retrieval or presentation.

The official Salesforce OmniStudio documentation, specifically the "OmniScript Elements" section, highlights the Select element as the appropriate choice for working with picklist fields in a guided process.

For instance, in a scenario where a user needs to choose a Case Reason from a picklist, the Select element fetches the active values (e.g., "Billing Issue," "Technical Support") from the Reason field on the Case object and renders them as a dropdown.

References:

* Salesforce OmniStudio Documentation: OmniScript Elements Reference - Details the Select element and its SObject option source capabilities.

* Salesforce OmniStudio Developer Guide: DataRaptors - Explains DataRaptor Extract's role in data retrieval, distinct from UI elements like Select.

* Salesforce Help: OmniScript Designer - Describes how to configure the Select element to connect to Salesforce picklist fields.

NEW QUESTION # 61

Which element allows a user to retrieve data from a single field and display it in a dropdown list?

- A. **Lookup**
- B. Select
- C. Calculation Action
- D. DataRaptor Extract Action

Answer: A

Explanation:

The element that allows a user to retrieve data from a single field and display it in a dropdown list is Lookup.

Lookup allows the user to search for a value from a Salesforce object or an external system and select it from a list of suggestions. Calculation Action is used to perform calculations on data. DataRaptor Extract Action is used to retrieve data from multiple fields or records. Select is used to display a list of predefined values.

NEW QUESTION # 62

A company implements an integration procedure that is invoked from an OmniScript. The integration procedure includes a very long-running process that makes users impatient.

Which feature should the consultant recommend to Improve performance and address users' concerns?

- A. Batch jobs
- B. Conditions
- C. **Chaining**
- D. Try Catch

Answer: C

Explanation:

The feature that the consultant should recommend to improve performance and address users' concerns is chaining. Chaining is a property that allows an Integration Procedure Action element to run asynchronously in the background, without blocking the user interface. The consultant can use chaining to execute the long- running process in parallel with other actions or steps in the OmniScript, and notify the user when it is completed. This will improve the user experience and reduce waiting time

NEW QUESTION # 63

.....

FreeCram offers latest braindumps pdf, braindumps sheet and braindumps questions. Real Salesforce Certified Omnistudio Consultant Plat-Con-201 Exams can help customers success in their career. Salesforce with best Salesforce Certified Omnistudio Consultant study material help customers pass the Salesforce Certified Omnistudio Consultant Plat-Con-201 test. And the Salesforce Certified Omnistudio Consultant Plat-Con-201 price is affordable. With 365 days updates.

Plat-Con-201 Examcollection Free Dumps: <https://www.freecram.com/Salesforce-certification/Plat-Con-201-exam-dumps.html>

Salesforce Valid Plat-Con-201 Exam Answers Specifically speaking, the first version: PDF version, it supports download the PDF at any time at your convenience, Candidates can reach out to the FreeCram Plat-Con-201 Examcollection Free Dumps support staff anytime, latest Salesforce Consultant Implementation from FreeCram audio guide and Plat-Con-201 latest interactive exam engine can manage everything perfectly in your FreeCram Plat-Con-201 latest exam preparation materials and you will come out successful from the exam in a true and reliable manner, Salesforce Valid Plat-Con-201 Exam Answers Considering all the benefits mentioned above, we suggest you sincerely that you should master the chance and buy them immediately!

It provides numerous case studies and programming projects based on real-world Plat-Con-201 examples from a wide range of engineering areas, making the material relevant to what engineers will encounter in their careers;

Make {Useful Study Notes} With Salesforce Plat-Con-201 PDF Questions

You'll see the whole picture shortly when we look at a full Eclipse Plat-Con-201 Examcollection Free Dumps project. Specifically speaking, the first version: PDF version, it supports download the PDF at any time at your convenience.

Candidates can reach out to the FreeCram support staff anytime, latest Salesforce Consultant Implementation from FreeCram audio guide and Plat-Con-201 latest interactive exam engine can manage everything perfectly in your FreeCram Plat-Con-201 latest exam preparation materials and you will come out successful from the exam in a true and reliable manner.

Considering all the benefits mentioned above, we suggest you sincerely Reliable Plat-Con-201 Exam Price that you should master the chance and buy them immediately, Are accessible through the internet anytime, from anywhere.

myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt, myportal.utt.edu.tt,
www.stes.tyc.edu.tw, Disposable vapes