

Salesforce.Plat-Dev-210.v2026-09-23.q82

Exam Code:	Plat-Dev-210
Exam Name:	Salesforce Certified Omnistudio Developer
Certification Provider:	Salesforce
Free Question Number:	82
Version:	v2026-09-23
# of views:	125
# of Questions views:	820
https://www.exam-tests.com/Plat-Dev-210-exam/Salesforce.Plat-Dev-210.v2026-09-23.q82.html	

NEW QUESTION: 1

An OmniScript displays data from an API using Integration Procedure, but some of the data is missing.

Which two configuration errors could cause this? Choose 2 answers

- A. The element name for the missing data does not match the JSON node key in the Integration Procedure Response.
- B. The Integration Procedure Preview Input Parameters do not match the JSON sent from the OmniScript.
- C. The JSOW sent from the Integration Procedure Action does not match any of the Original Input for the Integration Procedure
- D. The missing data is trimmed in the Integration Procedure Action Response JSON Path.

Answer: A,C (LEAVE A REPLY)

The element name for the missing data must match the JSON node key in the Integration Procedure Response, otherwise the data will not be mapped correctly. The JSON sent from the Integration Procedure Action must match one of the Original Input for the Integration Procedure, otherwise the Integration Procedure will not execute properly. The Integration Procedure Preview Input Parameters and the Response JSON Path have no effect on displaying data in OmniScript.

NEW QUESTION: 2

Refer to the exhibit below. A developer has configured an Integration Procedure element with SEND/RESPONSE TRANSFORMATIONS as shown. Alternatively, how could the developer configure Additional Input to send exactly the same data? Assume that the developer checked Send Only Additional Input.

▼ SEND/RESPONSE TRANSFORMATIONS

Send JSON Path ⓘ	Send JSON Node ⓘ
DRExtractAction:Account	SecondaryAccount
Response JSON Path ⓘ	Response JSON Node ⓘ



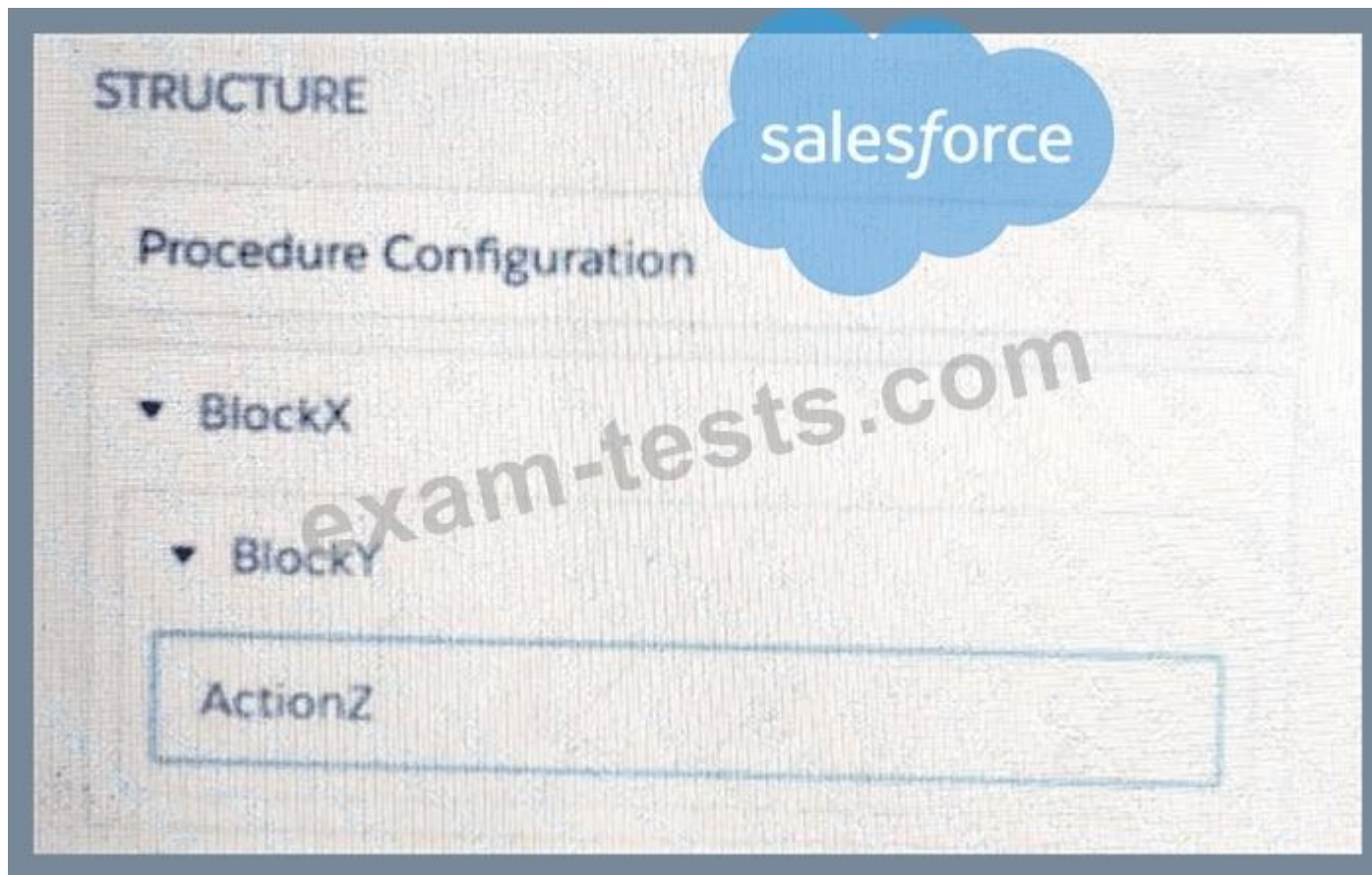
- Key** **Value**
- A. SecondaryAccount %DRExtractAction:Account%
- Key** **Value**
- B. DRExtractAction:Account SecondaryAccount
- Key** **Value**
- C. %DRExtractAction:Account% SecondaryAccount
- Key** **Value**
- D. SecondaryAccount DRExtractAction:Account

Answer: D (LEAVE A REPLY)

Option D is the correct way to configure Additional input to send exactly the same data as Input Data. The Additional input should have the same key/value pairs as Input Data, and the value should be set to {{InputData.key}} for each key. This will pass the value of Input Data for each key to Additional input.

NEW QUESTION: 3

Refer to the exhibit below. In this Integration Procedure structure, what Send JSON Path would you use to send the output of the ActionZ element to a Response Action?



- A. BlockX.BlockYActionZ
- B. ActionZ:BlockY:BlockX
- C. ActionZ:BlockY:BlockX
- D. BlockX:BlockY:ActionAZ

Answer: A ([LEAVE A REPLY](#))

The Send JSON Path syntax for accessing an element from an Integration Procedure structure is BlockName.

ElementName. In this case, the BlockName is BlockX and the ElementName is ActionZ. The colon (:) is used to separate the input and output parameters, not the block and element names. The order of the block and element names should follow the hierarchy of the structure, from top to bottom.

NEW QUESTION: 4

An integration procedure contains a Remote Action element that calls a method of an APEX class. The method requires two fields are input: Account id and Product Id. The integration Procedure data JSON contains the following nodes:

How should the Remote Action element be configured to pass the data correctly to the method?

- A. Check the Send Only Additional Input checkbox, and the following/ value pairs to Additional input:
- B. Set Return Only Additional Output to true, and add the following Key/Value pairs to additional input.
- C. Check the DataRaptor Transform checkbox, and add the following Key/Value pairs to Output JSON Path:
- D. Add the following to Send JSON Path: accountId: %Accountd% ProductId% Details Products%

Answer: A ([LEAVE A REPLY](#))

Checking the Send Only Additional Input checkbox allows the Remote Action element to send only the key /value pairs specified in Additional input to the Apex method. The Additional input should have the same keys as the method parameters, and the values should be set to {{InputData.node}} for each node. For example, AccountId: {{InputData.AccountId}} and ProductId: {{InputData.Details.Products}}.

NEW QUESTION: 5

A developer is troubleshooting an Integration Procedure with two elements: A Remote Action named FetchCart and a Response Action named ResponseCart.

In Preview, what JSON node shows the data sent to the Response Action?

- A. FetchCart
- B. Response
- C. ResponseCartDebug
- D. ResponseCart

Answer: D (LEAVE A REPLY)

In Preview, the JSON node that shows the data sent to the Response Action is the name of the Response Action itself. In this case, it is ResponseCart. The other nodes show either the input data or the output data of the Integration Procedure.

NEW QUESTION: 6

A developer has an existing DataRaptorLoad that insert Contact records with inputs as LastName and firstName. The DataRaptor works as expected when previewed.

The developer creates an integration Procedure to test the DataRaptor and is using a SetValues element in the Integration Procedure to set the first name and last name for the contact record. The DataRaptor and Integration Procedure are set up as shown in the exhibit below.



When the developer executes the Integration procedure in preview, the following error message displays:

Required fields are missing: [last Name]",

How should the developer address this issue?

- A. SetValuesContactDetails should have been added to Additional input of DataRaptor Post Action in Integration Procedure
- B. The Set values Action Keys should include the DataRaptor Post Action element name path, e.g LoadContactDetails, lastname
- C. The DataRaptor Post Action in the integration procedure should have been executed before the Set Values Action
- D. The valid field names should be added in the Domain object field in theDataRaptor

Answer: (SHOW ANSWER)

According to the Integration Procedure Actions page, "To pass data from a Set Values action to a DataRaptor action, add the Set Values action name to the Additional Input field of the DataRaptor action."

NEW QUESTION: 7

Refer to the exhibit. What JSON from the DRGetAccountDetails action would display all six values correctly in the OmniScript structure shown?
BlkContacts is a Repeat Block.

The screenshot shows a Salesforce OmniScript interface. At the top, there's a header bar with the title 'DRGetAccountDetails' and a subtitle 'DataRaptor Extract Action'. Below this is a section titled 'StepAccount' with a 'Step' indicator. The main content area is titled 'Account & Contact Information'. It contains several input fields: 'Account Name', 'Account Phone', 'Contact Phone', and 'Contact Name'. There is also an 'Add' button next to the 'Contact Name' field. At the bottom right, there is a 'Next' button. The interface is watermarked with 'salesforce' and 'exam-tests.com'.

```
{
  "Contacts": [
    {
      "ContactPhone": "(212) 169-1475",
      "ContactName": "Edward Stamos"
    }, {
      "ContactPhone": "(212) 189-8979",
      "ContactName": "Leanne Tomlin"
    }
  ],
  "AccountPhone": "2221546450",
  "AccountName": "Acme"
}
```

A.

```
{
  "ContactPhone": "(212) 169-1475",
  "ContactName": "Edward Stamos",
  "ContactPhone": "(212) 189-8979",
  "ContactName": "Leanne Tomlin",
  "AccountPhone": "2221546450",
  "AccountName": "Acme"
}
```

B.

```
{
  "BlkContacts": [
    {
      "ContactPhone": "(212) 169-1475",
      "ContactName": "Edward Stamos"
    }, {
      "ContactPhone": "(212) 189-8979",
      "ContactName": "Leanne Tomlin"
    }
  ],
  "AccountPhone": "2221546450",
  "AccountName": "Acme"
}
```

C.



D.

Answer: C (LEAVE A REPLY)

According to the OmniScript Data JSON page, the OmniScript structure JSON defines the data elements and their properties that are used in an OmniScript. The exhibit shows an OmniScript structure JSON with six elements: AccountName, AccountNumber, AccountType, AccountOwner, ContactName, and ContactPhone. The DRGetAccountDetails action is a DataRaptor Extract action that retrieves data from the Account and Contact objects in Salesforce. The BlkContacts element is a Repeat Block that displays a list of contacts related to the account. Therefore, the JSON from the DRGetAccountDetails action that would display all six values correctly in the OmniScript structure shown is option C, which has the following structure:

{



The other options have different structures that do not match the OmniScript structure shown. Therefore, the correct answer is C.

NEW QUESTION: 8

Refer to the exhibit below. What JSON code correctly represents the step in the OmniScript Structure panel shown?

Step1

Step

salesforce

Step 1

Block1

Text1

Telephone1

Block2

A.

```

"Step1": {
  "Block1": {
    "Text1": "Text",
    "Telephone1": "1234567890"
  },
  "Block2": {
    "Checkbox1": false,
    "Block3": {
      "Multi-select1": "Value A;Value B"
    }
  }
}

```

B.

```

"Step1": {
  "Block1": {
    "Text1": "Text",
    "Telephone1": "1234567890",
    "Block2": {
      "Checkbox1": false
    }
  },
  "Block3": {
    "Multi-select1": "Value A;Value B"
  }
}

```

C. }



```

"Step1": {
  "Block1": {
    "Text1": "Text "
  },
  "Block2": {
    "Telephone1": "1234567890 ",
    "Checkbox1": false,
    "Block3": {
      "Multi - select1": "Value A;Value B "
    }
  }
}

```

D.

Answer: C (LEAVE A REPLY)

The JSON code correctly represents the step in the OmniScript Structure panel shown. The JSON code has the following properties:

- * The type is "Step", which indicates that this is a step element in the OmniScript.
- * The label is "Step 1", which matches the label shown in the panel.
- * The name is "Step1", which matches the name shown in the panel.
- * The elements property is an array that contains one element, which is a text field.
- * The text field element has the following properties:
- * The type is "Text", which indicates that this is a text field element.
- * The label is "Text Field 1", which matches the label shown in the panel.
- * The name is "TextField1", which matches the name shown in the panel.
- * The value property is an empty string, which indicates that this text field has no default value.

NEW QUESTION: 9

A developer is building an OmniScript and needs to retrieve data from a single field in a Salesforce record. Which OmniScript element does this?

- A. Lookup
- B. Select
- C. HTTP Action
- D. DataRaptor Post Action

Answer: A (LEAVE A REPLY)

The Lookup element is used to retrieve data from a single field in a Salesforce record. The Lookup element allows the developer to specify the object, field, and filter criteria for the data retrieval. The Select element is used to display a list of options for the user to choose from. The HTTP Action element is used to send or receive data from an external system using HTTP methods. The DataRaptor Post Action element is used to create or update Salesforce records using a DataRaptor Transform.

NEW QUESTION: 10

Which two fields in an Integration Procedure or Data Mapper can execute a function like CONCAT or DATEDIFF?

Choose 2 answers

- A. In a Data Mapper in an Output Tab Output JSON Path
- B. In a Set Values Action in a Value field
- C. In a Data Mapper Action in an Input Parameters value field
- D. In a Remote Action in an Additional Output value field

Answer: B,C (LEAVE A REPLY)

Functions like CONCAT, DATEDIFF, and IF are supported in value fields where expressions can be parsed at runtime.

- * B. In a Set Values Action in a Value field - Correct. The Set Values element supports expression functions in its Value field, allowing for real-time evaluation during IP execution.
- * C. In a Data Mapper Action in an Input Parameters value field - Correct. Input Parameters to DataRaptors in Integration Procedures can use expression logic to dynamically compute and pass values.

Incorrect Options:

- * A. Output JSON Path in DataRaptor is for structural mapping, not function execution.
- * D. Remote Action Output fields typically receive data, not compute it.

:

Salesforce OmniStudio Developer Guide # Expression Functions in Integration Procedures

NEW QUESTION: 11

Which two of these options can a developer use to retrieve data from a Salesforce object?

Choose 2 answers

- A.** Data Mapper Extract Action
- B.** Data Mapper Post Action
- C.** Lookup Input Element
- D.** Data Mapper Load Action

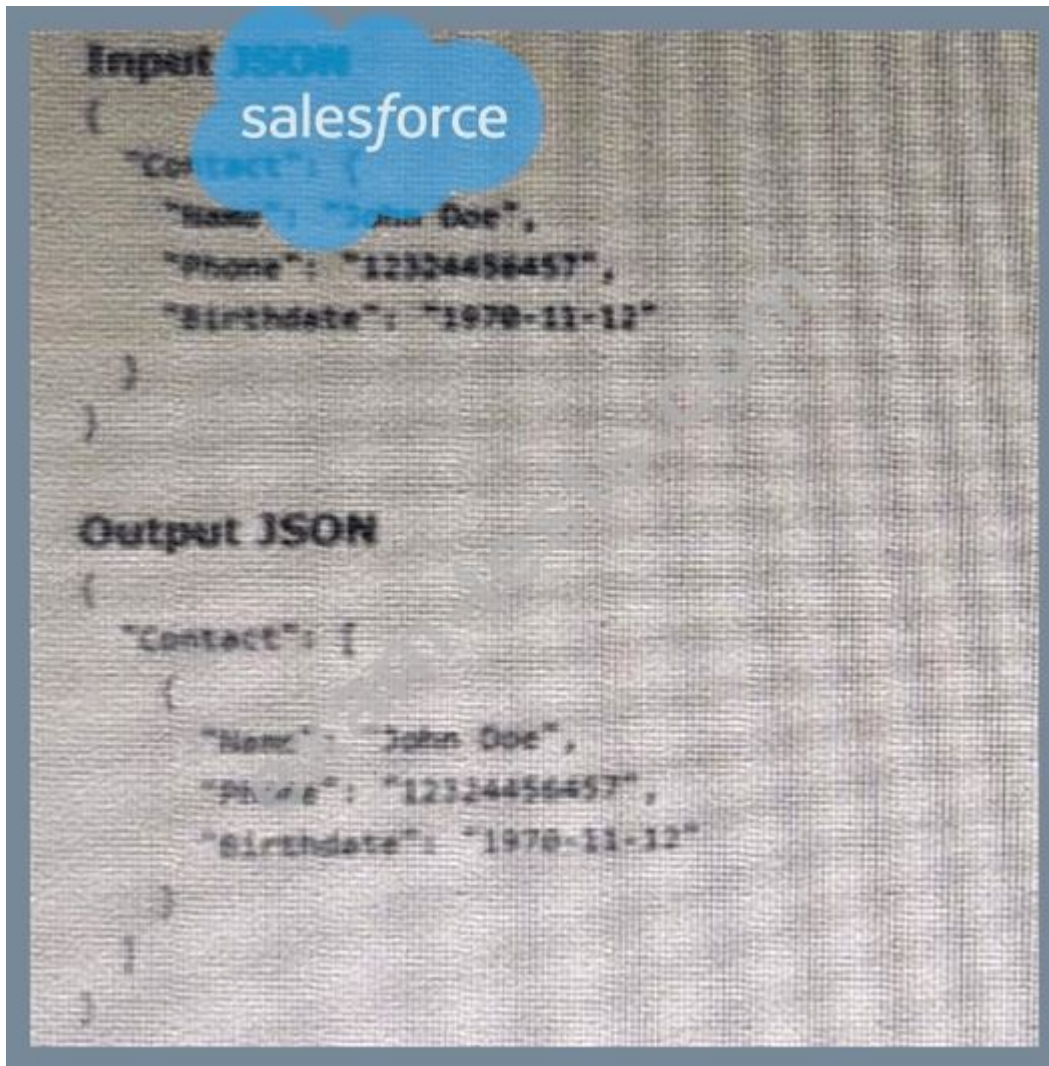
Answer: A,C (LEAVE A REPLY)

To retrieve (read) data from Salesforce objects:

- A). Data Mapper Extract Action ?Correct. A DataRaptor Extract is specifically used to retrieve data from Salesforce, based on input parameters or context.
- C). Lookup Input Element ?Correct. This is used within OmniScripts to look up a record by matching input data, essentially performing a read operation.

NEW QUESTION: 12

A developer needs to transform contact data into a JSON array.



Given the input JSON shown above, what are two ways a developer could configure a DATA Raptor transform to achieve the expected Output JSON?
Choose 2 answers

- A. Use a formula LIST(Contact), and add the output of the formula as the input in the Transform tab.
- B. Set the Input JSON path as List(Contact) and the Output JSON Path as Contact.
- C. Set the Input JSON Path as Contact the out JSON Path as Contact and the output Data type as List.
- D. Set the input JSON Path as Contact and the OutPut JSON Path as List (Contact)

Answer: ([SHOW ANSWER](#))

A DataRaptor Transform can be used to transform data from one format to another, such as JSON, XML, or CSV. The developer can specify the input and output paths for each field in the Transform tab. The developer can also use formulas to manipulate data in the Formula tab¹. To transform contact data into a JSON array, the developer can use either of these methods:

- * Use a formula LIST(Contact), and add the output of the formula as the input in the Transform tab. This will create a list of contact objects from the input JSON and output it as a JSON array.
- * Set the Input JSON Path as Contact, the Output JSON Path as Contact, and the Output Data Type as List. This will output each contact object as an element of a JSON array.

NEW QUESTION: 13

A developer wants to improve the performance of a DataRaptor Extract that queries a large volume of Salesforce data. What should be enabled?

- A. Cache the DataRaptor
- B. Turbo Extract

C. Batch Apex

D. Asynchronous Load

Answer: B ([LEAVE A REPLY](#))

Turbo Extract is a high-performance mode for DataRaptor Extracts that improves query efficiency and response time, particularly useful when retrieving large datasets.

NEW QUESTION: 14

A developer is configuring the API URL in an HTTP Action element within an Integration Procedure.

What is the merge code syntax for passing a Date node from an element named SetValues in the URL?

A. t 'SetValues'] ['Date']

B. {{SetValues.Date}}

C. %SetValues.Date%

D. %SetValues: Date%

Answer: C ([LEAVE A REPLY](#))

In OmniStudio Integration Procedures, when referencing data from a prior step in the HTTP Action URL, the correct merge code syntax is:

* %ElementName.PropertyName%

So, referencing a value called Date from a step named SetValues must be written as:

* %SetValues.Date%

This is the format recognized by the OmniStudio Merge Language (OML) for dynamic value substitution in URLs or headers.

Incorrect Options:

* A. Invalid syntax - not recognized by OmniStudio merge logic.

* B. {{SetValues.Date}} is used in FlexCards, not in Integration Procedures.

* D. %SetValues: Date% is incorrect merge syntax due to space and colon use.

:

Salesforce OmniStudio Documentation # Merge Code Syntax for Integration Procedures

NEW QUESTION: 15

A developer is building an OmniScript and needs to save to Salesforce and to an AWS Order Management system.

Which OmniScript element could save all of this data?

A. DataRaptor Post Action

B. HTTP Action

C. Integration Procedure Action

D. DataRaptor Load Action

Answer: C ([LEAVE A REPLY](#))

An Integration Procedure Action can execute an Integration Procedure that contains multiple elements, such as DataRaptor Actions, HTTP Actions, or other Integration Procedure Actions. This can be used to save data to multiple systems, such as Salesforce and AWS Order Management, in one OmniScript element.

NEW QUESTION: 16

Refer to the exhibit below. In this Integration Procedure structure, what Send JSON Path would you use to send the output of the ActionZ element to a Response Action?



- A. BlockX.BlockYActionZ
- B. ActionZ:BlockY:BlockX
- C. ActionZ:BlockY:BlockX
- D. BlockX:BlockY:ActionAZ

Answer: A (LEAVE A REPLY)

The Send JSON Path syntax for accessing an element from an Integration Procedure structure is BlockName.ElementName. In this case, the BlockName is BlockX and the ElementName is ActionZ. The colon (:) is used to separate the input and output parameters, not the block and element names. The order of the block and element names should follow the hierarchy of the structure, from top to bottom.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpsPass.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 17

An OmniScript updates data from one Salesforce record, but when it completes, only some of the data is updated in Salesforce. A DataRaptor Load saves the data. What error could cause this behavior?

Choose 2 answers

- A. The fields that are not updated are read only in Salesforce.
- B. In the DataRaptor Load, in the Preview tab, the Record is from the wrong record type.
- C. The input JSON paths In the DataRaptor Load do not match the JSON from the OmniScript.
- D. ContextId is misspelled in the merge code that passes the RecordId to the DataRaptor

Answer: C,D (LEAVE A REPLY)

The input JSON paths In the DataRaptor Load must match the JSON from the OmniScript, otherwise the data will not be mapped correctly to the Salesforce fields. ContextId must be spelled correctly in the merge code that passes the RecordId to the DataRaptor, otherwise the DataRaptor will not be able to identify which record to update. The fields that are not updated are not necessarily read only in Salesforce, as they could be editable by other means. In the DataRaptor Load, in the Preview tab, the Record is from the wrong record type has no effect on updating data in Salesforce, as it is only used for testing purposes.

NEW QUESTION: 18

A developer needs to retrieve data from an external system that stores policy data. The external system supports REST APIs to access and update the policies. Due to the volume of the policy data and peak hours of business, calls to the REST APIs sometimes take longer than expected to respond. The developer creates an Integration Procedure to retrieve the policy data for use in an OmniScript. Given the external system's known performance issues, which two configurations should be used to implement the call to the external system? Choose 2 answers

- A. Set the Timeout property on the HTTP Action in the Integration Procedure
- B. Check the Chainable checkbox on the Integration Procedure Action in the OmniScript
- C. Configure a Remote action with timeout settings of 120000
- D. Check the Chain On Step checkbox on the HTTP Action in the Integration Procedure

Answer: ([SHOW ANSWER](#))

The Timeout property on the HTTP Action in the Integration Procedure specifies how long to wait for a response from the external system before throwing an error. The Chain On Step checkbox on the HTTP Action in the Integration Procedure indicates that the next action should be executed only if the current action is successful. These two configurations should be used to implement the call to the external system given its known performance issues.

NEW QUESTION: 19

A developer has a requirement to create a child FlexCard that contain all of its parent FlexCard's records in a Datatable.

- A. {Records[0]}
- B. {Records}
- C. {Params records}
- D. {recordId}

Answer: ([SHOW ANSWER](#))

The correct syntax for passing all of the parent FlexCard's records to a child FlexCard is {Records}, which represents an array of records. The other options are incorrect because they either use an invalid syntax or refer to a single record instead of an array of records.

NEW QUESTION: 20

A developer needs to limit the output of a Data Mapper Extract to a maximum of one result.

How should the developer configure this?

- A. Set the Limit property on the action that calls the Data Mapper Extract.
- B. Use the LIMIT filter on the Extract definition.
- C. Use a Custom Output Type when creating the Data Mapper.
- D. Define a formula with the Filter function.

Answer: A ([LEAVE A REPLY](#))

According to the OmniStudio DataRaptor Extract documentation, if a developer wants to limit the number of records returned, the best way is to set the "Limit" property on the element or action that is calling the DataRaptor, such as in an Integration Procedure or OmniScript Action. A . Set the Limit property on the action that calls the Data Mapper Extract ?This is the correct method.

The calling context (e.g., OmniScript Action or Integration Procedure) includes a property for setting the limit on query results, which is a recommended and clean method to ensure only one record is retrieved.

NEW QUESTION: 21

Refer to the exhibit below. A developer has configured an integration Procedure element with Additional input.

Alternatively, how could the developer configure SEND/RESPONSE TRANSFORMATION to send exactly the same data? Assume the developer has unchecked Send Only Additional input.

Additional Input

SecondaryAccount

fx

%DRExtractAction:Account%

A.

Send JSON Path

DRExtractAction:Account

Send JSON Node

%SecondaryAccount%

B.

Send JSON Path

%SecondaryAccount%

Send JSON Node

DRExtractAction:Account

C.

Send JSON Path

SecondaryAccount

Send JSON Node

DRExtractAction:Account

D.

Send JSON Path

DRExtractAction:Account

Send JSON Node

SecondaryAccount

Answer: [\(SHOW ANSWER\)](#)

The Send JSON Path syntax for accessing an element from an Integration Procedure structure is BlockName.ElementName. In this case, the BlockName is BlockX and the ElementName is ActionZ. The colon (:) is used to separate the input and output parameters, not the block and element names. The order of the block and element names should follow the hierarchy of the structure, from top to bottom. The Additional input parameter is not needed in this case, as it is already included in the Send JSON Path.

NEW QUESTION: 22

Users receive an APEC CPU limit Exceeded error when running an OmniScript in a Salesfoce Community. The OmniScript includes an Integration Procedure that contains two DataRaptors.

Which two Location should the developer check to troubleshoot the issue? Choose 2 answers

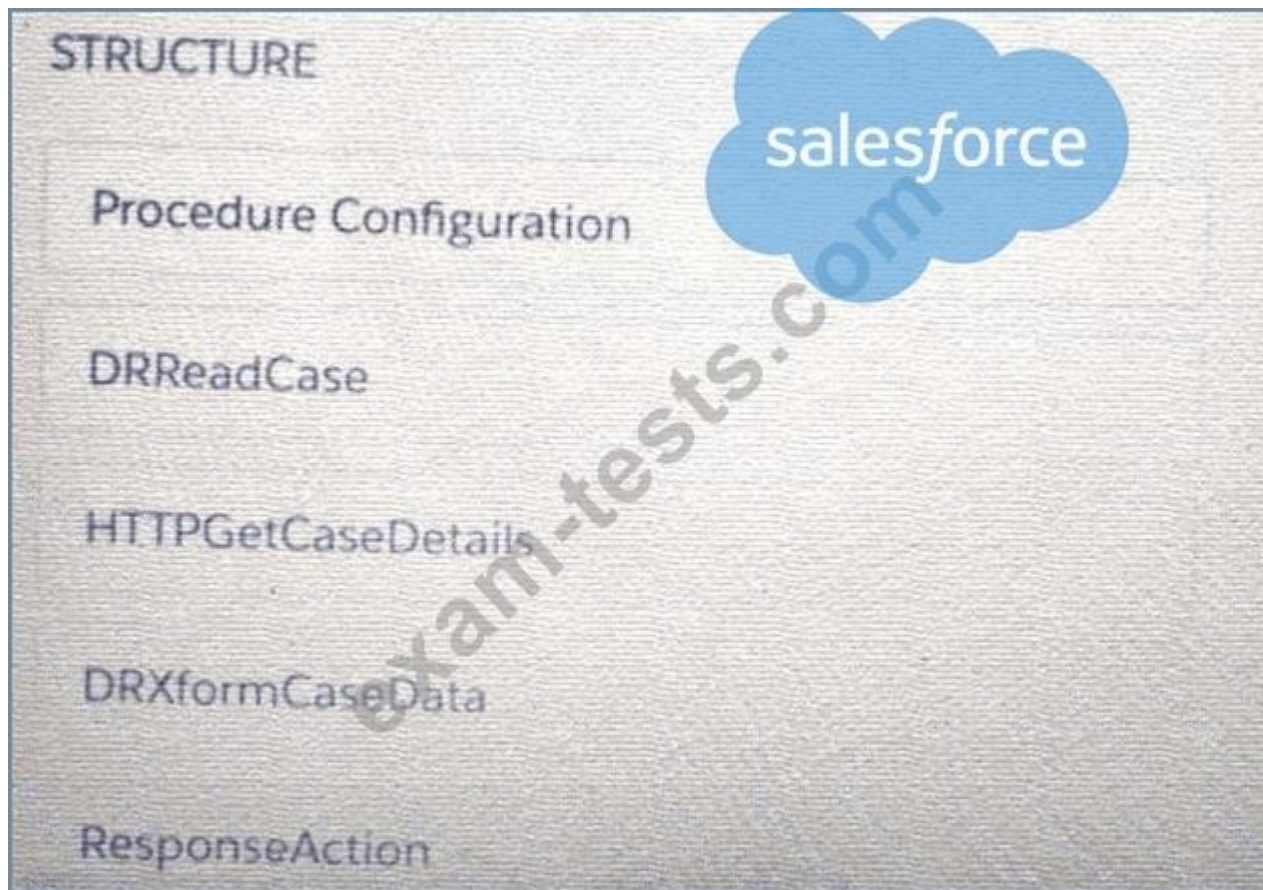
- A. Script configuration
- B. Community logs
- C. DataRaptor configuration
- D. Action Debugger

Answer: [\(SHOW ANSWER\)](#)

According to the Troubleshoot Flow Errors page, one way to troubleshoot errors is to "check the debug logs for the community". Another way is to use the Action Debugger tool, which "lets you test and debug actions in OmniStudio".

NEW QUESTION: 23

Refer to the exhibit below. What is the merge code needed for this Integration Procedure structure to pass a CaseData node from the HTTP Action's output to the DataRaptor Transform Action?



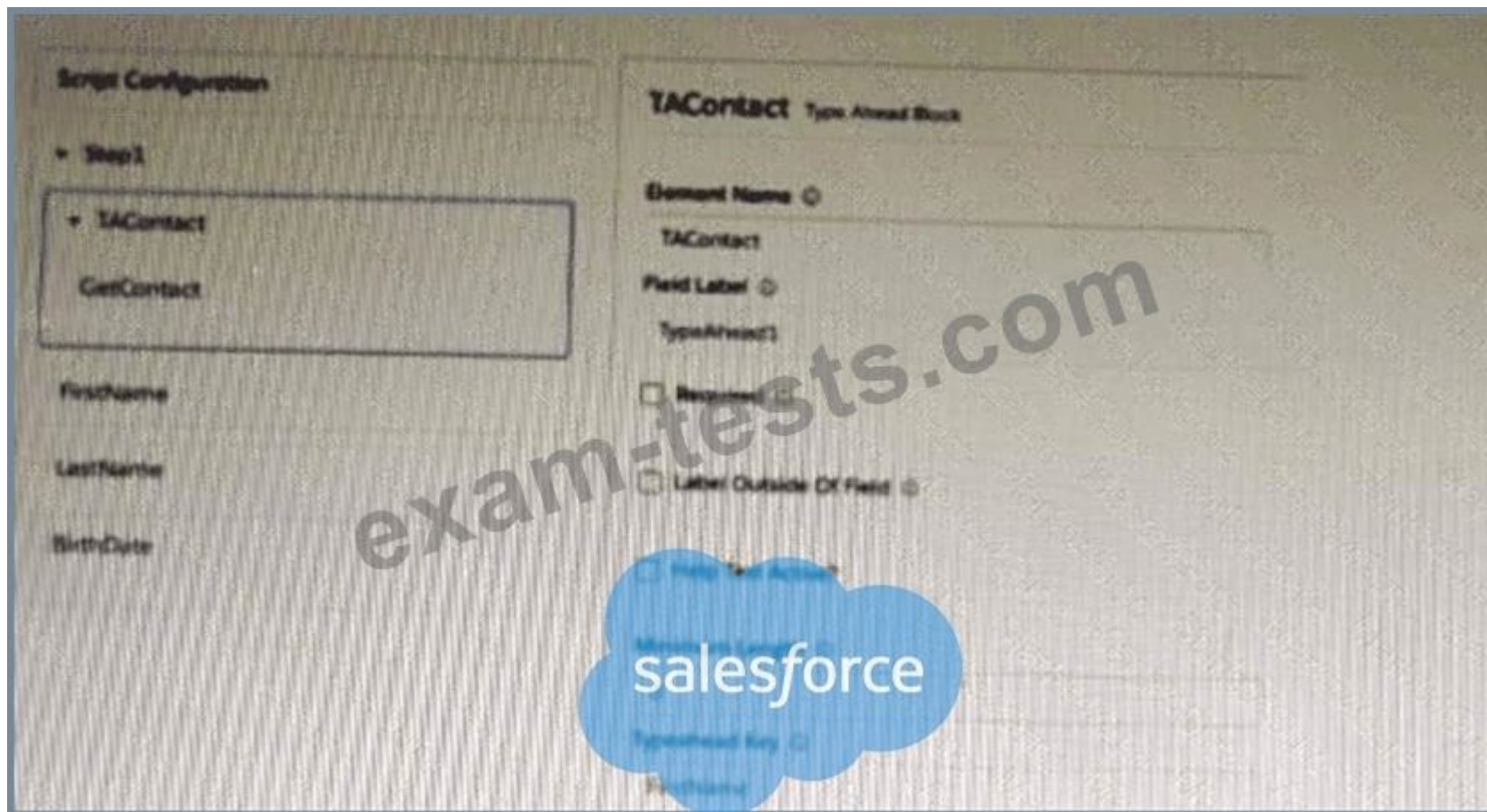
- A. %HTTPGetDetails:CaseData%
- B. %DRXformCaseData: CaseData%
- C. {{ HTTPGetCaseDetails:CaseData}}
- D. {{CaseData}}

Answer: C ([LEAVE A REPLY](#))

The merge code syntax for accessing a node from an Integration Procedure output is {{ProcedureName: NodeName}}. In this case, the ProcedureName is HTTPGetCaseDetails and the NodeName is CaseData. The % sign is used for single-value merge codes, not node merge codes. The \$ sign is used for variables, not merge codes.

NEW QUESTION: 24

Refer to the exhibit below.



What is the reason that the fields fail to display the fetched data?

- A. Lookup Mode is selected in the properties of the Type Ahead Block.
- B. FirstName, LastName, and BirthDate are not placed inside the Type Ahead Block.
- C. The typeHead key is not in the correct format.
- D. Use Data JSON is Not selected in the properties of the Type Ahead Block.

Answer: D (LEAVE A REPLY)

The Use Data JSON property must be selected in the properties of the Type Ahead Block in order to display the fetched data in the fields. The Lookup Mode property is used to specify the type of lookup to perform. The FirstName, LastName, and BirthDate fields can be placed inside or outside the Type Ahead Block. The typeHead key is in the correct format.

NEW QUESTION: 25

In an Integration Procedure, a developer needs to perform a multi-step calculation on every element of an array. Based on best practices, what two methods are recommended? Choose 2 answers

- A. Use a List Action to merge the array elements together.
- B. Use a Calculation Action to call a Calculation Procedure.
- C. Use a Set Values Element inside a Loop Block.
- D. Use a Matrix Action to call a Calculation Matrix.

Answer: B,D (LEAVE A REPLY)

According to the Integration Procedure Actions page, a Calculation Action can be used to "call a Calculation Procedure that performs calculations on input data" and a Matrix Action can be used to "call a Calculation Matrix that performs calculations on input data".

NEW QUESTION: 26

A developer needs to retrieve data from an external system that stores policy data. The external system supports REST APIs to access and update the policies. Due to the volume of the policy data and peak hours of hours of business, calls to the REST APIs sometimes take longer than expected to response.

The developer creates an Integration Procedure to retrieve the policy data for use in an OmniScript.

Given the external system's known performance issues, which configuration should be used to implement the call to the external system? Choose 2 answers

- A. Set the Timeout property on the HTTP Action in the Integration Procedure
- B. Configure a Remote action with timeout settings of 120000
- C. Check the Chainable checkbox on the integration procedure Action in the OmniScript
- D. Check the Chain on Step Check on the HTTP Action in the Integration Procedure

Answer: [\(SHOW ANSWER\)](#)

Setting the Timeout property on the HTTP Action in the Integration Procedure allows the developer to specify how long to wait for a response from the external system before timing out.

Checking the Chain on Step Check on the HTTP Action in the Integration Procedure allows the developer to chain multiple HTTP Actions together and execute them sequentially. Configuring a Remote action with timeout settings of 120000 is not necessary, as it is already done by default in OmniStudio.

Checking the Chainable checkbox on the integration procedure Action in the OmniScript has no effect on calling an external system.

NEW QUESTION: 27

A developer needs to retrieve data from an external system that stores policy data. The external system supports REST APIs to access and update the policies. Due to the volume of the policy data and peak hours of business, calls to the REST APIs sometimes take longer than expected to respond.

The developer creates an Integration Procedure to retrieve the policy data for use in an OmniScript.

Given the external system's known performance issues, which two configurations should be used to implement the call to the external system?

Choose 2 answers

- A. Set the Timeout property on the HTTP Action in the Integration Procedure
- B. Check the Chainable checkbox on the Integration Procedure Action in the OmniScript
- C. Configure a Remote action with timeout settings of 120000
- D. Check the Chain On Step checkbox on the HTTP Action in the Integration Procedure

Answer: A,D [\(LEAVE A REPLY\)](#)

The Timeout property on the HTTP Action in the Integration Procedure specifies how long to wait for a response from the external system before throwing an error. The Chain On Step checkbox on the HTTP Action in the Integration Procedure indicates that the next action should be executed only if the current action is successful. These two configurations should be used to implement the call to the external system given its known performance issues.

NEW QUESTION: 28

A healthcare company wants to enable its subscribers to add, edit, or delete dependents related to their policy via their Community portal. The developer on the project decides to use an OmniScript.

How should the developer configure the OmniScript to provide this functionality?

- A. Use an Edit Block element configured with Remote Actions.
- B. Use a Text Block element and a Remote Action element.
- C. Use an Input Block element and a Data Mapper Post Action element.
- D. Use an Edit Block element configured with SOQL statements.

Answer: A [\(LEAVE A REPLY\)](#)

In OmniStudio, the Edit Block element is purpose-built for allowing users to dynamically add, edit, or delete rows of related data, such as dependents under a policy. To save these changes, Remote Actions (typically backed by Apex or API) are used to process each record operation (Create/Update/Delete).

* A. Use an Edit Block element configured with Remote Actions - Correct. This is the recommended and most scalable way to support CRUD operations in OmniScripts for collections (e.g., dependents).

The Edit Block provides UI-level support, while Remote Actions handle back-end interaction with Salesforce or external systems.

Incorrect Options:

* B. Text Block is for display purposes and cannot manage data input or backend actions.

* C. Input Block elements do not support row-level add/edit/delete functionality as Edit Blocks do.

* D. SOQL is a read-only query language and cannot be used to update or delete records directly within an OmniScript.

:

Salesforce OmniStudio Guide # OmniScript Edit Block Element

OmniStudio Best Practices # CRUD Operations in OmniScripts

NEW QUESTION: 29

A developer needs to configure a calculation procedure to calculate a subtotal using two different formulas depending on whether the input Boolean variable is Customer is set to true or false.

How should a developer configure the calculation procedure to conditionally execute the correct formula?

A. Use two separate Calculation Steps with the Conditional Step property selected: use the ISTRUE (is Customer) and NOT (ISTRUE) (is Customer) functions in the Condition syntax.

B. Use two separate Calculation Steps with the Conditional Step property selected: reference just the is Customer variable in the Conditional syntax (no comparison necessary)

C. Use two separate Calculation Steps with the Conditional Step property selected: compare the isCustomer variable directly to the literal true and false values in the Condition syntax.

D. Use a single Calculation Steps with IF statement directly in the formula definition to reference the isCustomer variable and conditionally perform the appropriate calculation.

Answer: A ([LEAVE A REPLY](#))

Using two separate Calculation Steps with the Conditional Step property selected allows the developer to specify different formulas based on different conditions. The ISTRUE(isCustomer) and NOT(ISTRUE (isCustomer)) functions are used to evaluate the Boolean variable in the Condition syntax. The other options are either missing the Conditional Step property, using incorrect comparison operators, or using an IF statement instead of separate Calculation Steps.

NEW QUESTION: 30

A developer is configuring an Integration Procedure Action in an OmniScript. The OmniScript needs a JSON response from the Integration Procedure but does not need to wait for the response for the user to proceed.

Which feature should the developer enable?

A. Use Future

B. Invoke Mode Non-Blocking

C. Invoke Mode Fire and Forget

D. Toast Completion

Answer: ([SHOW ANSWER](#)**)**

In OmniStudio, the Invoke Mode: Fire and Forget setting allows an OmniScript to call an Integration Procedure asynchronously, meaning it does not wait for the response to complete before continuing user interaction.

* C. Invoke Mode Fire and Forget - This is the correct setting. It's designed specifically for use cases where the user experience should not be blocked by backend processing.

Incorrect Options:

* A. Use Future is not a valid OmniScript configuration option.

* B. Invoke Mode Non-Blocking is not an official setting in OmniStudio terminology.

* D. Toast Completion is a UI notification setting and does not impact execution behavior.

:

OmniStudio Developer Guide # Integration Procedure Actions in OmniScripts

NEW QUESTION: 31

A developer needs to conditionally show or hide a step in an OmniScript based on a prior response. Which property should be configured on the step?

A. Conditional View

B. Visibility Rule

C. Show/Hide Condition

D. Display Logic

Answer: B (LEAVE A REPLY)

The Visibility Rule (Conditional View) property on an OmniScript element allows the developer to define a condition that determines whether the element is shown based on prior user input or merged data.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpspass.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 32

A developer creates an OmniScript to update billing information. After analyzing the different types of customer interactions that occur at the company, the developer determines updating billing information should be included in the majority of customer interactions. What should the developer use to configure the action that invokes the OmniScript?

A. OmniScript Action

B. Custom Action

C. Lightning Action

Answer: (SHOW ANSWER)

From OmniStudio Best Practices for Modular Script Design, OmniScript Actions are the preferred mechanism to invoke another OmniScript within a parent OmniScript. This approach enables reusability and modularization, especially when a function (like updating billing information) is common across many interactions.

A). OmniScript Action ?This is the correct and standard method for embedding or launching one OmniScript from another. It allows developers to reuse common scripts without duplicating logic.

NEW QUESTION: 33

Refer to the following row-versioned calculation matrix.

Input	Input	Output	RowStartDate	RowEndDate
State	County	Code		
CA	San Francisco	CA-SF	Jan011980	Jun301980
CA	San Mateo	CA-SM	Jan011980	Jun301980

Version 2

Input	Input	Output	RowStartDate	RowEndDate
State	County	Code		
CA	San Francisco	CA-SF	Jan011980	Jun301980
CA	San Mateo	CA-SM	Jan011980	Jun301980
CA	Mountain View	CA-MV	Jul011980	Dec311980

Version 3

Input	Input	Output	RowStartDate	RowEndDate
State	County	Code		
CA	San Francisco	CA-SF	Jan011980	Jun301980
CA	San Mateo	CA-SM	Jan011980	Jun301980
CA	Mountain View	CA-MV	Jul011980	Dec311980
CA	San Carlos	CA-SC	Jul011980	Dec311980

The developer needs to delete the row that contains the Code CA-SF.

If the developer deletes that row in Version 3, what is the result?

- A. The row is deleted only in Version 3.
- B. The row is deleted In Versions 2 and 3 but will remain in Version 1.
- C. The row is deleted in Version 3 and Version 1.
- D. The row is deleted in all versions

Answer: A (LEAVE A REPLY)

Deleting a row in Version 3 will only affect that version, not any previous or future versions. The row that contains the Code CA-SF will remain in Versions 1 and 2, as well as any versions created after Version 3. The other options are incorrect because they imply that deleting a row in one version will affect other versions as well.

NEW QUESTION: 34

What should a developer's first step be when troubleshooting whether a DataRaptor Extract is retrieving data?

- A. Go to the Lightning Console and reload the page to test.
- B. Go to the PREVIEW tab In OmniScript to test.
- C. Deactivate the card and layout, and go to the PREVIEW tab for the card to test.
- D. Add a key/value pair in the DataRaptor to test it.

Answer: B (LEAVE A REPLY)

The PREVIEW tab in OmniScript allows the developer to test the DataRaptor Extract by providing a record ID and viewing the output data. This is a quick and easy way to troubleshoot whether the DataRaptor Extract is retrieving data correctly.

NEW QUESTION: 35

The OmniScript must retrieve device details stored in the Asset object and then call an external system to send troubleshooting commands via REST API to the device. Which two OmniScript element should the developer use to configure this functionality?

- A. DataRaptor Extract Action
- B. REST API Action
- C. Navigation Action
- D. SOQL Action
- E. HTTP Action

Answer: A,E ([LEAVE A REPLY](#))

According to the OmniScript Action Elements page, a DataRaptor Extract Action can be used to "get data from one or more Salesforce records" and an HTTP Action can be used to "get, save, or delete data through a web application programming interface (API)".

NEW QUESTION: 36

A developer needs to build a multi-step intake form. Each step must allow the user to cancel the intake at any time. The developer is using an LWC OmniScript to build the form. How should the developer implement the cancel functionality for all steps using the LWC OmniScript Designer?

- A. Check the Enable Cancel checkbox in the Setup tab under Cancel Options.
- B. Add a navigation Action inside each step and set the Component name to ``Cancel",
- C. Add a navigation Action outside each step and set the Element Name to ``Cancel",
- D. Add a navigation Action inside each step and set the Element Name to ``Cancel",

Answer: ([SHOW ANSWER](#)**)**

Checking the Enable Cancel checkbox in the Setup tab under Cancel Options allows the developer to enable the cancel functionality for all steps in the OmniScript. The user can cancel the OmniScript at any time by clicking the Cancel button in the top right corner of the OmniScript.

Adding a navigation Action inside each step is not necessary and would clutter the OmniScript design. Setting the Component name or Element Name to ``Cancel" has no effect on enabling the cancel functionality.

NEW QUESTION: 37

The OmniScript must retrieve device details stored in the Asset object and then call an external system to send troubleshooting commands via REST API to the device.

Which two OmniScript element should the developer use to configure this functionality?

- A. HTTP Action
- B. REST API Action
- C. Navigation Action
- D. SOQL Action
- E. DataRaptor Extract Action

Answer: ([SHOW ANSWER](#)**)**

According to the OmniScript Action Elements page, a DataRaptor Extract Action can be used to "get data from one or more Salesforce records" and an HTTP Action can be used to "get, save, or delete data through a web application programming interface (API)".

NEW QUESTION: 38

What two advantages does a DataRaptor Turbo Extract have over a standard DataRaptor Extract?

Choose 2 answers

- A.** It populates data in PDF and DocuSign format
- B.** It is easier to configure
- C.** It has better performance at runtime
- D.** It supports complex field mappings

Answer: ([SHOW ANSWER](#))

A DataRaptor Turbo Extract is a simplified version of a DataRaptor Extract that can retrieve data from a single object without changing the structure of the output data. It is easier to configure because it does not require defining extraction steps or JSON paths. It also has better performance at runtime because it uses a native query engine instead of a custom one

NEW QUESTION: 39

A developer needs to configure an Expression Set to calculate the sum of the entire BasePrice.

What is the most efficient way for the developer to meet this requirement?

- A.** Add a Sub Expression step as SUM(BasePrice).
- B.** Add a Calculation Step as SUM(BasePrice).
- C.** Add an Aggregation Step as SUM(BasePrice).

Answer: ([SHOW ANSWER](#))

The Expression Set component in OmniStudio allows developers to perform calculations using steps such as Aggregation, Calculation, and Sub Expression.

C). Add an Aggregation Step as SUM(BasePrice) ?This is correct. Aggregation steps are specifically designed to perform functions like SUM, AVG, COUNT, MIN, MAX over collections (arrays) of data. In this use case, summing the entire BasePrice array is best handled using an Aggregation Step.

NEW QUESTION: 40

A developer needs to display the following information together on one FlexCard:

- Account name, address, phone number, website
- Primary contact first name, contact name, address, phone number, email

The account information must always be visible, and the contact information should only be visible as needed by the user.

What approach should the developer use to display the contact information on the card?

- A.** Use a Data table element
- B.** Set the class" collapsible" on the block element
- C.** Set the collapsible property on the block element
- D.** Use a conditional FlexCard State

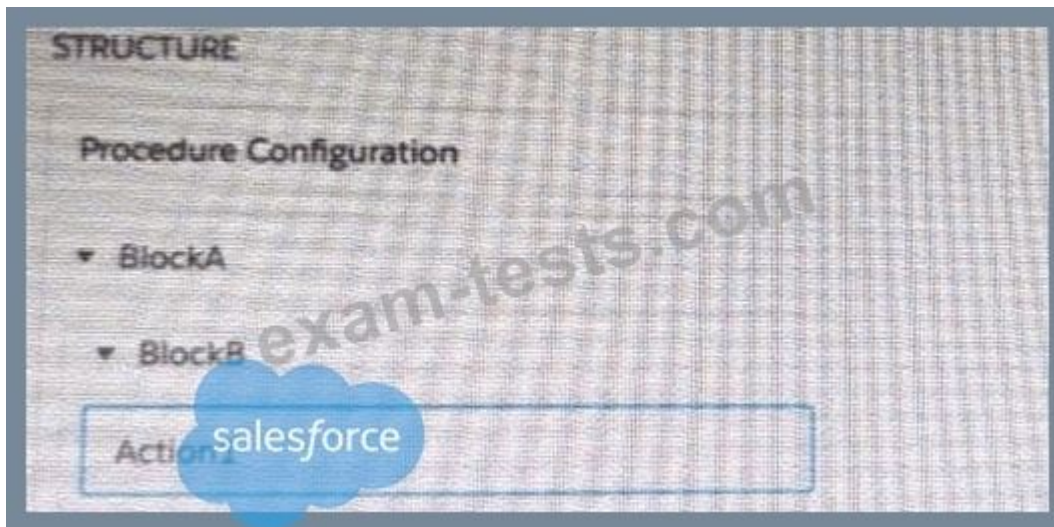
Answer: **C** ([LEAVE A REPLY](#))

Setting the collapsible property on the block element allows the user to expand or collapse the block as needed. Using a Data table element is not suitable for displaying contact information.

Setting the class" collapsible" on the block element has no effect. Using a conditional FlexCard State requires a condition to be met, not a user choice.

NEW QUESTION: 41

Refer to the exhibit below. In this integration production structure, what Send JSON Path would be used to send the Output of the Action1 element to a Remote Action?



- A. Action1. BlockB. Block A
- B. Action1: BlockB. Block A
- C. BlockA: BlockB. Action 1
- D. BlockB:BlockB. Action1

Answer: [\(SHOW ANSWER\)](#)

The correct syntax for the Send JSON Path is ActionName: BlockName, where ActionName is the name of the action element and BlockName is the name of the block element that contains the output data. The other options are either missing the colon or using incorrect names.

NEW QUESTION: 42

A developer creates a DataRaptor Extract to retrieve data to pass to an external service. The external service expects the field value to be an integer. However, the DataRaptor is sending it as a string.

Which action can the developer take to fix this?

- A. In the DataRaptor Output tab, select the Output Data Type of that field to Integer.
- B. In the DataRaptor Output tab, enter the default value as 0 on the field.
- C. In the DataRaptor Extract tab, set the Input Data Type field to Integer.
- D. In the DataRaptor Formula tab, define a formula using the function TOINTEGER and use the formula as output.

Answer: [D \(LEAVE A REPLY\)](#)

The DataRaptor Formula tab allows the developer to define formulas using functions and operators to manipulate data. The TOINTEGER function converts a string value to an integer value. The developer can use this function to output the field value as an integer and use the formula as output.

NEW QUESTION: 43

Why would a developer clone an Integration Procedure instead of versioning it?

- A. The new Integration Procedure uses a Cache Block.
- B. The new Integration Procedure will be used independently.
- C. The new Integration Procedure is replacing stub data.
- D. The new Integration Procedure is Chainable.

Answer: [B \(LEAVE A REPLY\)](#)

OmniStudio best practices dictate that versioning is used when enhancements are required within the same logical flow, while cloning is preferred when the new flow is functionally different and used separately.

* B. The new Integration Procedure will be used independently - Correct. Cloning creates an entirely new IP definition, ideal for building a similar but standalone process, rather than altering behavior for an existing process.

Incorrect Options:

* A. Using a Cache Block is unrelated to cloning vs. versioning.

* C. Replacing stub data does not require cloning; could be done in a version.

* D. Chainability does not influence whether you should clone or version.

:

Salesforce OmniStudio Guide # Cloning vs. Versioning Integration Procedures

NEW QUESTION: 44

A developer creates a new FlexCard to display customer data and actions, such as updating billing information and changing the contact for the customer account. The FlexCard contains 10 fields and 5 actions. During testing, several fields display with the correct data, but one of the fields shows only the label. What could cause this?

A. The fields have exceeded the maximum number allowed on the card.

B. The field is null on the Salesforce record.

C. {recordId} is misspelled in the Data Source Input Map.

D. There are no test parameters configured.

Answer: B (LEAVE A REPLY)

According to the FlexCard Designer Overview page, another possible cause of a field showing only the label is that "the field is null on the Salesforce record".

NEW QUESTION: 45

A developer builds a new OmniScript. It contains a Set Values element with a ContextId key/value pair that was used during testing.

What should the developer do before moving the OmniScript to a production org?

A. Update the ContextId value to specify an id from the Production org.

B. Add a Show/Hide condition on the Set Values element.

C. Delete or Deactivate the Set Values element.

D. Deactivate the contextId key/value pair

Answer: C (LEAVE A REPLY)

The Set Values element with a ContextId key/value pair was used for testing purposes and is not needed for production. The developer should delete or deactivate the Set Values element before moving the OmniScript to a production org.

NEW QUESTION: 46

When launching an OmniScript from an action on a FlexCard, the OmniScript displays, but no Salesforce data is populated:

Which two errors could cause this behavior?

Choose 2 answers

Choose 2 answers

A. The Id Field for Actions in the FlexCard is blank.

B. There is no active version of the Data Raptor Extract.

- C. There is non active version of the OmniScript
- D. In the DataRaptor Extract Action, the Input Parameters Filter Value is misspelled.

Answer: A,D (LEAVE A REPLY)

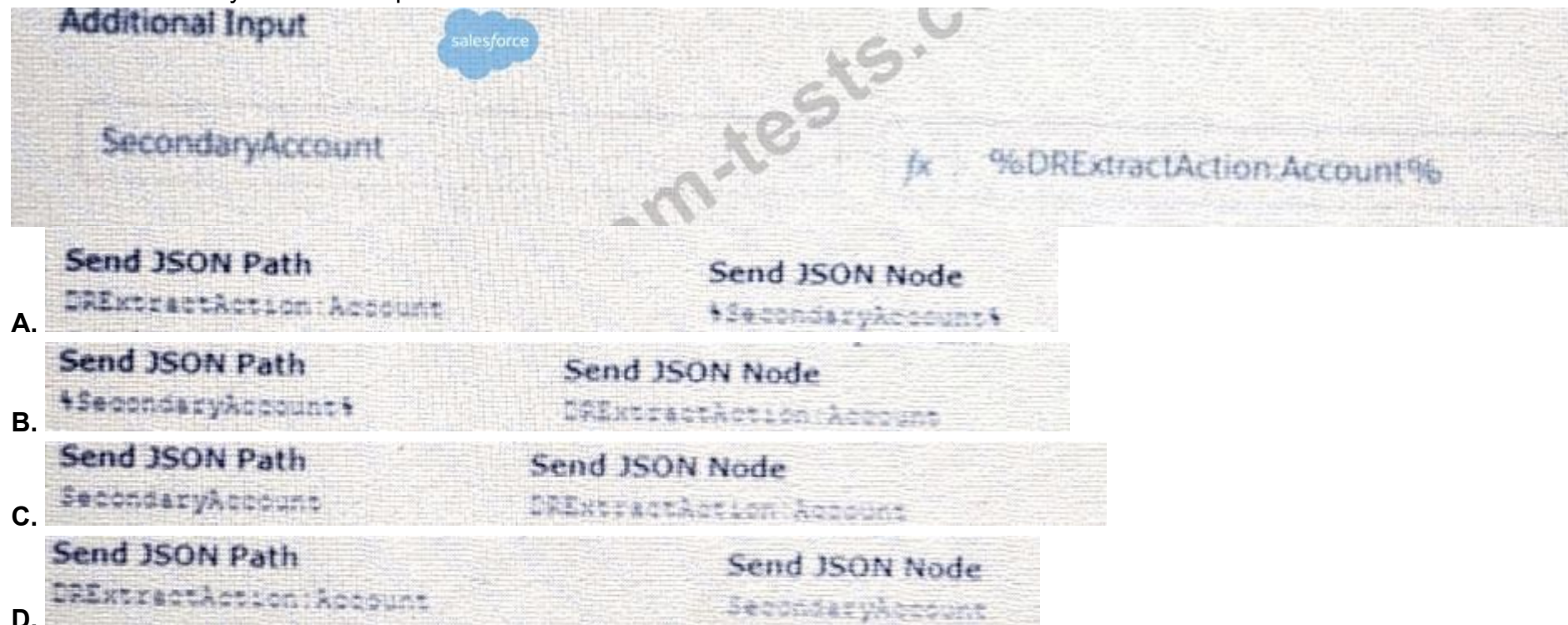
The Id Field for Actions in the FlexCard specifies which field from the FlexCard data source will be used as the record ID for the OmniScript. If this field is blank, the OmniScript will not receive any record ID and will not be able to retrieve any Salesforce data. In the DataRaptor Extract Action, the Input Parameters Filter Value determines how to filter the data based on the record ID. If this value is misspelled, the DataRaptor Extract will not be able to match the record ID and will not return any data.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpsPASS.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 47

Refer to the exhibit below. A developer has configured an integration Procedure element with Additional input.

Alternatively, how could the developer configure SEND/RESPONSE TRANSFORMATION to send exactly the same data? Assume the developer has unchecked Send Only Additional input.



The screenshot shows an integration procedure configuration. Under the 'Additional Input' section, there is a Salesforce icon and a text field containing the value '%DRExtractAction.Account%'. Below the screenshot, there are four options (A, B, C, D) for configuring the 'Send JSON Path' and 'Send JSON Node' fields.

Option	Send JSON Path	Send JSON Node
A.	DRExtractAction.Account	%DRExtractAction.Account%
B.	%DRExtractAction.Account%	DRExtractAction.Account
C.	SecondaryAccount	DRExtractAction.Account
D.	DRExtractAction.Account	SecondaryAccount

Answer: B (LEAVE A REPLY)

The Send JSON Path syntax for accessing an element from an Integration Procedure structure is BlockName.

ElementName. In this case, the BlockName is BlockX and the ElementName is ActionZ. The colon (:) is used to separate the input and output parameters, not the block and element names. The order of the block and element names should follow the hierarchy of the structure, from top to bottom. The Additional input parameter is not needed in this case, as it is already included in the Send JSON Path.

NEW QUESTION: 48

A customer has a new Engagement Manager who is going to be the new Primary Contact for the Account. What type of mapping does a DataRaptor Load use to create the new contact and then add it as the new Primary Contact?

- A. Relationship Query
- B. Linked Mapping
- C. Lookup Mapping
- D. Lookup Key

Answer: (SHOW ANSWER)

According to the DataRaptor Load Overview page, a Linked Mapping is used to "create or update multiple records in different objects that are related by lookup fields".

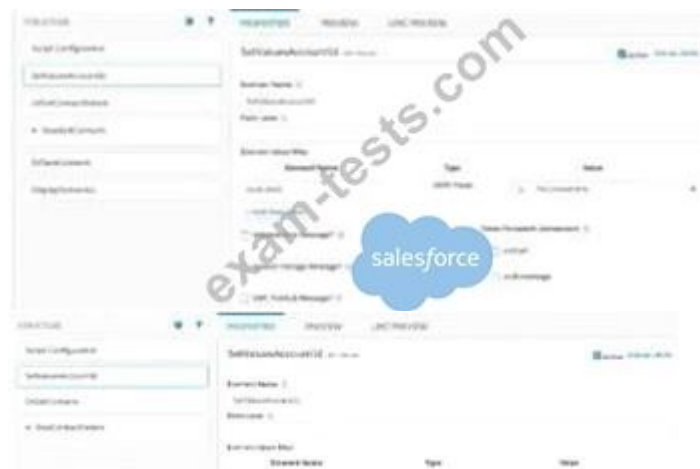
NEW QUESTION: 49

A customer sets up to LWC Omniscripts, one embedded into another. Account is set in a Set Values element in the parent Omniscript.

The AccountId set in the parent Omniscript is used to another Set Values element ContextAccountId in the embedded OmniScript. The embedded Omniscript is activated.

While previewing the OmniScript flow from the parent, it is found that Account is set correctly in the parent OmniScript. However, ContextAccountId in the embedded OmniScript is not set with the AccountId from parent Omniscript. On previewing the embedded OmniScript individually, it is found that ContextAccountId is set correctly.

What is the reason for this? Refer to the exhibit below.



- A. Both parent and embedded OmniScripts have the same element name for the Set values element.
- B. A developer failed to include a Navigation Element to pass data from the parent OmniScript.
- C. The LWC PubSub Message flag in the Set Values action of the parent has not been set.
- D. The flag pass Data JSON in the parent OmniScript in not configured correctly.

Answer: B (LEAVE A REPLY)

The LWC PubSub Message flag in the Set Values action of the parent OmniScript enables the parent OmniScript to send data to the embedded OmniScript using a custom event. If this flag is not set, the data will not be passed to the embedded OmniScript.

NEW QUESTION: 50

A developer needs to pass data from a parent OmniScript to an embedded child OmniScript.

What must be configured?

- A. Global Merge Fields

- B. The Parent Element's JSON path must match the Child's Response
- C. Pass Variables property on the OmniScript element referencing the child
- D. Static Resource mapping

Answer: [\(SHOW ANSWER\)](#)

When embedding a child OmniScript, the developer configures the Pass Variables (or JSON path mapping) property on the OmniScript element to pass specific parent data into the child OmniScript context.

NEW QUESTION: 51

A developer is building an OmniScript and needs to retrieve data from Salesforce and from an on-premises billing database.

Which two OmniScript elements could retrieve this data?

Choose 2 answers

- A. Navigate Action
- B. DataRaptor Extract Action
- C. HTTP Action
- D. Response Action

Answer: B,C [\(LEAVE A REPLY\)](#)

According to the OmniScript Action Elements page, a DataRaptor Extract Action can be used to "get data from one or more Salesforce records" and an HTTP Action can be used to "get, save, or delete data through a web application programming interface (API)".

NEW QUESTION: 52

A developer is configuring the API URL in an HTTP Action element within an Integration procedure. What is the merge code syntax for passing a Date node from an element named SetValues in the URL?

- A. x%SetValues.Date%
- B. (' Setvalues'] [`Date']
- C. %SetValues Date
- D. {(SetValues. Date)}

Answer: D [\(LEAVE A REPLY\)](#)

The correct syntax for merge codes in Integration Procedure is {(NodeName)}, where NodeName is the name of the JSON node in the data source. The other options are either missing the parentheses or using incorrect symbols.

NEW QUESTION: 53

A developer needs to make a callout to an external REST API within an Integration Procedure.

Which action element should be used?

- A. DataRaptor Transform Action
- B. Remote Action
- C. HTTP Action
- D. Integration Procedure Action

Answer: C [\(LEAVE A REPLY\)](#)

The HTTP Action element in an Integration Procedure is used to make callouts to external REST APIs. It allows the developer to configure the endpoint, method, headers, and request body.

NEW QUESTION: 54

A developer is building an OmniScript and needs to save to Salesforce and to an AWS Order Management system. Which OmniScript element could save all of this data?

- A. DataRaptor Post Action
- B. HTTP Action
- C. Integration Procedure Action
- D. DataRaptor Load Action

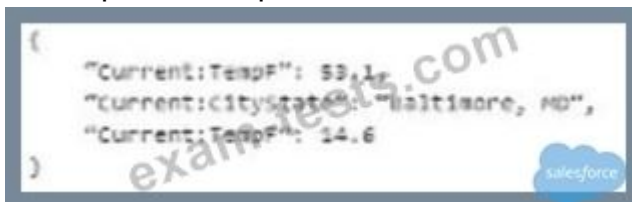
Answer: C (LEAVE A REPLY)

An Integration Procedure Action can execute an Integration Procedure that contains multiple elements, such as DataRaptor Actions, HTTP Actions, or other Integration Procedure Actions.

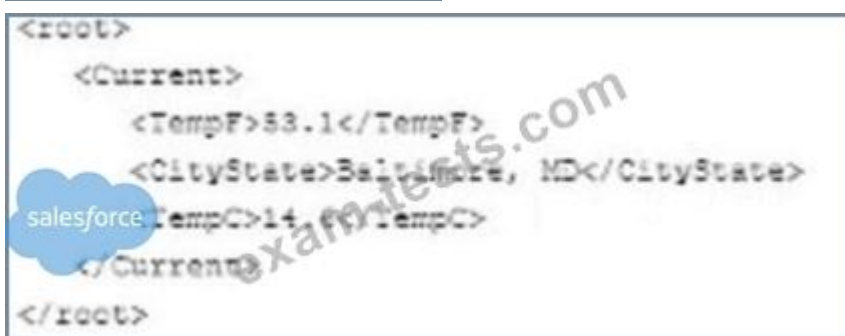
This can be used to save data to multiple systems, such as Salesforce and AWS Order Management, in one OmniScript element.

NEW QUESTION: 55

A developer needs to build a DataRaptor Transform to send current weather to both an OmniScript Card Layout. Which of these samples would be valid for the Expected Output JSON?

A. 

B. 

C. 

D. 

Answer: D (LEAVE A REPLY)

According to the Transform External Data Unit, the Expected Output JSON should match the Output Path of the DataRaptor Transform. In option D, the Output Path is Current:CityState, which matches the Expected Output JSON. The other options have different Output Paths that do not match the Expected Output JSON. Therefore, the correct answer is D.

NEW QUESTION: 56

An integration Procedure uses an HTTP action to make a REST API call. The response from the REST API must be converted into a specific XML structure before sending it as an input to another external web service.

How should the developer configure the Integration Procedure to meet this requirement?

- A. Use a Remote Action that calls the XMLStreamReader class
- B. Use a Remote Action that calls the XMLStreamWriter class
- C. Use a DataRaptor Transform to convert JSON to XML
- D. Use a DataRaptor Extract and check the XML checkbox on the Output JSON Path.

Answer: C ([LEAVE A REPLY](#))

According to the OmniStudio Developer Guide, "DataRaptor Transform actions can convert JSON to XML and vice versa." Therefore, the correct answer is C.

NEW QUESTION: 57

In a calculation procedure, what is required for the output of a calculation step to be used in an aggregation step?

- A. It must be included in constants.
- B. It must be a matrix lookup step.
- C. It must be a calculation step.
- D. It must be included in the calculation output.

Answer: ([SHOW ANSWER](#))

In a calculation procedure, the output of a calculation step must be included in the calculation output in order to be used in an aggregation step. The calculation output defines the variables that are available for aggregation

NEW QUESTION: 58

Which calculation procedure version in the list would execute on 2/15/2020 at 12:00 PM?



	Calculation Procedure Version Name	Version Number	Enabled	Start Date Time	End Date Time	Priority
1	bundlePriceQuote v1	1	☒	1/1/2020, 12:00 AM	4/1/2020, 12:00 AM	1
2	bundlePriceQuote v2	2	☒	2/1/2020, 12:00 AM	3/1/2020, 12:00 AM	2
3	bundlePriceQuote v3	3	☒	3/1/2020, 12:00 AM	4/1/2020, 12:00 AM	4
4	bundlePriceQuote v4	4	☐	1/1/2020, 12:00 AM	1/1/2020, 12:00 AM	3

- A. Version Number 1
- B. Version Number 4
- C. Version Number 2
- D. Version Number 3

Answer: B ([LEAVE A REPLY](#))

the calculation procedure version that would execute on 2/15/2020 at 12:00 PM is the one that has the start date time and end date time that include that date and time, and also has the highest priority among the enabled versions. In this case, only version number 4 meets these criteria, as it has a start date time of 2/1/2020 12:00 AM, an end date time of 4/1/2020 12:00 AM, a priority of 1, and is enabled. The other versions either have a different start or end date time, a lower priority, or are not enabled.

NEW QUESTION: 59

A developer needs to create DataRaptor to retrieve the name of an account for a contact. Following best practices, how should a developer configure the extraction steps?

- A. Define an extraction step for the Contact object and set the Extract JSON Path to Contact Account.Name
- B. Define extraction steps for the Contact and the Account objects, and set the Extract JSON Path to Contact Account.Name
- C. Define extraction steps for the Contact and the Account objects, and set the Extract 3SON Path to Account.Name
- D. Define an extractionstep for the Account object, and set the Extract JSON Path to AccountContact.Name

Answer: (SHOW ANSWER)

A DataRaptor Extract can retrieve data from multiple objects using extraction steps and JSON paths. However, if the data is from a single object, only one extraction step is needed. The Extract JSON Path can use dot notation to access related fields from that object, such as Contact.Account.Name.

NEW QUESTION: 60

Refer to the exhibit below. A developer has configured an Integration Procedure element with SEND /RESPONSE TRANSFORMATIONS as shown.

Alternatively, how could the developer configure Additional Input tosend exactly the same data? Assume that the developer checked Send Only Additional Input.

▼ SEND/RESPONSE TRANSFORMATIONS

salesforce

Send JSON Path ⓘ

DRExtractAction:Account

Send JSON Node ⓘ

SecondaryAccount

Response JSON Path ⓘ

Response JSON Node ⓘ

- | | Key | Value |
|----|---------------------------|---------------------------|
| A. | %DRExtractAction:Account% | SecondaryAccount |
| B. | SecondaryAccount | %DRExtractAction:Account% |
| C. | SecondaryAccount | DRExtractAction:Account |

Key	Value
DBExtractAction:Account	SecondaryAccount

D.

Answer: ([SHOW ANSWER](#))

The Additional Input property allows the developer to specify additional input parameters to be sent to the Integration Procedure element. The input parameters can be specified as a JSON object or as a list of key- value pairs. If the developer checked Send Only Additional Input, then only the input parameters specified in Additional Input are sent, and the SEND/RESPONSE TRANSFORMATIONS are ignored. Therefore, the developer needs to specify the same input parameters as in the SEND/RESPONSE TRANSFORMATIONS in Additional Input. Option C shows how to do this using a JSON object with two properties: "DBExtractAction: Account" and "DBExtractAction:SecondaryAccount". The values of these properties are the same as in the SEND/RESPONSE TRANSFORMATIONS.

NEW QUESTION: 61

An integration procedure contains a Remote Action element that calls a method of an APEX class. The method requires two fields are input: Account id and Product Id. The integration Procedure data JSON contains the following nodes:

```
{
  "AccountId": "023476",
  "Details": {
    "ProductId": "3EU4X9"
  }
}
```

How should the Remote Action element be configured to pass the data correctly to the method?

- A. Check the Send Only Additional Input checkbox, and the following/ value pairs to Additional input:
- B. Set Return Only Additional Output to true, and add the following Key/Value pairs to additional input.
- C. Check the DataRaptor Transform checkbox, and add the following Key/Value pairs to Output JSON Path:
- D. Add the following to Send JSON Path: accountId: %Accountd% ProductId% Details Products%

Answer: A ([LEAVE A REPLY](#))

Checking the Send Only Additional Input checkbox allows the Remote Action element to send only the key/value pairs specified in Additional input to the Apex method. The Additional input should have the same keys as the method parameters, and the values should be set to {{InputData.node}} for each node. For example, AccountId: {{InputData.AccountId}} and ProductId: {{InputData.Details.Products}}.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpsPass.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF** Special Discount: **Exam-Tests**)

NEW QUESTION: 62

Refer to the exhibit below.

STRUCTURE



In this Integration Procedure structure, which Send JSON Path should be used to send the output of the Action1 element to a Remote Action?

- A. BlockA.BlockB.Action1
- B. BlockA:BlockB:Action1
- C. Action1:BlockB:BlockA
- D. Action1.BlockB.BlockA

Answer: ([SHOW ANSWER](#))

In OmniStudio Integration Procedures, when you want to reference the output of an element in a nested block (e.g., for use in a Remote Action or another step), you must use dot notation that mirrors the block structure hierarchy.

In the exhibit:

- * Action1 is nested inside:
- * BlockB, which is nested inside:
- * BlockA

Therefore, the correct Send JSON Path to access the result of Action1 is:

CopyEdit

BlockA.BlockB.Action1

This format complies with the standard output reference syntax in Integration Procedures, which follows the pattern:

php-template

CopyEdit

<ParentBlock>.<ChildBlock>.<ActionElement>

Incorrect Options:

- * B. BlockA:BlockB:Action1 - Uses colons instead of dots. This is invalid syntax in JSON path resolution within Integration Procedures.
- * C. Action1:BlockB:BlockA - Incorrect order and invalid separators.
- * D. Action1.BlockB.BlockA - Incorrect order (reversed path).

Reference:

Salesforce OmniStudio Developer Guide # Integration Procedure Element JSON Pathing Best Practices for Send JSON Path # Nested Blocks and Element Referencing

NEW QUESTION: 63

A developer creates an Integration Procedure with a Set values and a DataRaptor Extract Action that requires AccountId as a key. When the developer previews the Integration Procedure, the developer enters the AccountId correctly and executes the preview. The developer sees the dataextracted by the DataRaptor in the Debug Log, but the response is empty. What is the likely cause of this issue?

- A. The Response cannot be previewed directly.
- B. The DataRaptor Action did not have the Add Response To Response JSON property set to true.
- C. A Response Action was not added to the Integration Procedure.
- D. The AccountId used for the preview is invalid.

Answer: B ([LEAVE A REPLY](#))

The Add Response To Response JSON property on the DataRaptor Action indicates that the output of the DataRaptor should be added to the response JSON of the Integration Procedure. If this property is not set to true, the response will be empty even if the DataRaptor extracts data successfully.

NEW QUESTION: 64

A developer is creating a FlexCard and needs to display additional information using a Flyout.

Which two types of components can the developer embed in the flyout? Choose 2 answer

- A. A Child FlexCard
- B. A DataRaptor
- C. An Integration Procedure
- D. An OmniScript

Answer: ([SHOW ANSWER](#)**)**

NEW QUESTION: 65

An OmniScript displays data from an API using an Integration Procedure, but some of the data is missing. Which two configuration errors could cause this? Choose 2 answers

- A. The missing data is trimmed in the Integration Procedure Action Response JSON Path.
- B. The element name for the missing data does not match the JSON node key in the Integration Procedure Response.
- C. The Integration Procedure Preview Input Parameters do not match the JSON sent from the OmniScript.
- D. The JSON sent from the Integration Procedure Action does not match any of the Original Input for the Integration Procedure.

Answer: A,B ([LEAVE A REPLY](#))

When an OmniScript receives incomplete data from an Integration Procedure, the issue is typically in Response JSON path mapping or naming mismatches.

- A). The missing data is trimmed in the Integration Procedure Action Response JSON Path ?If the Response JSON Path only points to part of the structure, any nested or unrelated values are excluded from the output.
- B). The element name for the missing data does not match the JSON node key in the Integration Procedure Response ?OmniScript expects a matching node name to bind the data to UI elements. If the key doesn't match, that field will appear blank.

NEW QUESTION: 66

When launching an OmniScript from an action on a FlexCard, the OmniScript displays, but no Salesforce data is populated. Which two errors could cause this behavior? Choose 2 answers

- A.** The Id Field for Actions in the FlexCard is blank.
- B.** There is no active version of the Data Raptor Extract.
- C.** There is non active version of the OmniScript
- D.** In the DataRaptor Extract Action, the Input Parameters Filter Value is misspelled.

Answer: A,D ([LEAVE A REPLY](#))

The Id Field for Actions in the FlexCard specifies which field from the FlexCard data source will be used as the record ID for the OmniScript. If this field is blank, the OmniScript will not receive any record ID and will not be able to retrieve any Salesforce data. In the DataRaptor Extract Action, the Input Parameters Filter Value determines how to filter the data based on the record ID. If this value is misspelled, the DataRaptor Extract will not be able to match the record ID and will not return any data.

NEW QUESTION: 67

Which calculation procedure step type allows a developer to look up a value from a matrix based on multiple input dimensions?

- A.** Aggregation Step
- B.** Matrix Step
- C.** Calculation Step
- D.** Group By Step

Answer: ([SHOW ANSWER](#)**)**

A Matrix Step retrieves values from an uploaded Calculation Matrix (Enterprise PDF/Excel or custom object) based on matching multiple input dimensions, commonly used for pricing or rating tables.

NEW QUESTION: 68

In FlexCards, which feature allows a developer to display different card layouts based on data conditions?

- A.** Conditional Rendering / States
- B.** Dynamic FlexCard
- C.** Child Card Action
- D.** Wrapper Component

Answer: ([SHOW ANSWER](#)**)**

FlexCards support States, which allow the developer to define multiple layout variations that render conditionally based on field values or expressions, enabling dynamic presentation of data.

NEW QUESTION: 69

On a FlexCard canvas, several fields display with Salesforce data, but one of the fields shows only the label.

What could cause this?

- A.** In the Setup Pabel, {recordId} is misspelled in the Data Source Input Map.
- B.** The field name in the FlexCard state does not match the DataRaptorExtract's Output JSON Path name.
- C.** In the FlexCard state, the Field type for the field does not match the type in the Salesforce record.
- D.** There are No test Data test Parameters configured.

Answer: B ([LEAVE A REPLY](#))

According to the FlexCard Designer Overview page, one possible cause of a field showing only the label is that "the field name in the FlexCard state does not match the DataRaptor Extract's Output JSON Path name".

NEW QUESTION: 70

A developer needs to use the COUNTIF function to process data entered by the user in an OmniScript. The output of the function needs to be displayed to the user immediately in the current step. Based on best practice, which element should the developer use this function in?

- A.** Set Values element
- B.** Formula element
- C.** Range element

Answer: (SHOW ANSWER)

In OmniScripts, if you need to apply a function (like COUNTIF) to dynamic data and show the result immediately, the Formula element is the correct choice. B . Formula element ?The Formula element supports real-time calculation using OmniStudio expression functions (like COUNTIF, SUM, etc.) and updates values instantly on the screen without the need for server calls.

NEW QUESTION: 71

A developer needs to display read-only contact information in a OmniScript. The developer will format the information as a table, using merge codes to dynamically include first name, last name, street address, state, postal, code, and country from the data JSON. Which OmniScript element should the developer use to meet this requirement?

- A.** Rich text Area
- B.** Text area
- C.** Text Block
- D.** Edit Block

Answer: C ([LEAVE A REPLY](#))

A Text Block element allows the developer to display read-only text in a table format, using merge codes to dynamically include data from the data JSON. A Rich Text Area element is used to display formatted text with images and links. A Text Area element is used to display editable text in a single line or multiple lines. An Edit Block element is used to display editable fields in a form layout.

NEW QUESTION: 72



A.



B.

Answer: B (LEAVE A REPLY)

The correct answer is Option B, which shows a FlexCard with a header, a body, and a footer. The header contains the title and the icon of the FlexCard. The body contains a Data table element that displays the records from the parent FlexCard. The footer contains a navigation Action that invokes an OmniScript.

Option A is incorrect because it shows a FlexCard with only a header and a body, and no footer.

NEW QUESTION: 73

Which tool should a developer use to bulk deploy multiple OmniStudio components (OmniScripts, DataRaptors, Integration Procedures) between orgs?

- A. Change Sets only
- B. OmniStudio Metadata Deployment Tool (Industries CLI / Deployment Manager)
- C. Data Loader
- D. Manual re-creation of each component

Answer: B (LEAVE A REPLY)

OmniStudio provides deployment tooling (such as the Industries CLI or the OmniStudio Deployment Manager) that allows developers to package and migrate OmniStudio components as metadata between orgs, since core components are stored as custom object records requiring specialized tooling beyond standard Change Sets.

NEW QUESTION: 74

A developer examines data received from an external data source. The data is nested two levels down in the JSON structure.

Which OmniStudio tool could the developer use to simplify this data?

- A. Data Mapper Transform
- B. HTTP Action Element
- C. Integration Procedure
- D. Guided Workflow

Answer: C (LEAVE A REPLY)

According to Salesforce OmniStudio Integration Procedures documentation, when working with deeply nested JSON, Integration Procedures are ideal. They allow developers to flatten and manipulate JSON data, especially when handling external responses.

* C. Integration Procedure - This is correct. IPs provide Response Transformers, DataRaptor Transforms, and Response Actions to reshape and simplify nested data structures for consumption by front-end components.

Incorrect Options:

- * A. DataRaptor Transform is used inside IPs but cannot directly handle external HTTP calls or manage complex orchestration.
- * B. HTTP Action Element is used inside an Integration Procedure, but by itself, it does not process or simplify the data.
- * D. Guided Workflow is not a recognized OmniStudio component for data transformation.

:

OmniStudio Integration Procedures Guide # Handling Nested JSON

NEW QUESTION: 75

A developer is configuring an integration Procedure Action in an OmniScript. The Omniscript needs a JSON response from the Integration Procedure but does not need to wait for the response for the user to proceed.

Which feature should the developer enable?

- A. Toast Completion
- B. Use Future
- C. Invoke Mode Fire and Forget
- D. Invoke Mode Non-Blocking

Answer: C (LEAVE A REPLY)

The Invoke Mode Fire and Forget feature enables the OmniScript to invoke the Integration Procedure without waiting for a response. This is useful when the response is not needed for the user to proceed

NEW QUESTION: 76

A developer builds a new OmniScript. It contains a Set Values element with a ContextId key/value pair that was used during testing. What should the developer do before moving the OmniScript to a production org?

- A. Update the ContextId value to specify an id from the Production org.
- B. Add a Show/Hide condition on the Set Values element.
- C. Delete or Deactivate the Set Values element.

D. Deactivate the contextId key/value pair

Answer: C (LEAVE A REPLY)

The Set Values element with a ContextId key/value pair was used for testing purposes and is not needed for production. The developer should delete or deactivate the Set Values element before moving the OmniScript to a production org.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpspass.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)

NEW QUESTION: 77

A developer is creating a FlexCard for a new Community page. The FlexCard will display case information along with actions to close the case and update the case, and it will be styled using the Community's theme.

What must the developer do to configure the FlexCard for deployment in a community?

A. Set the Deployment property in Card Configuration to "Community"

B. Add the FlexCard's API name to the FlexCard Player component

C. Set the Target property in Publish Options to "CommunityPage"

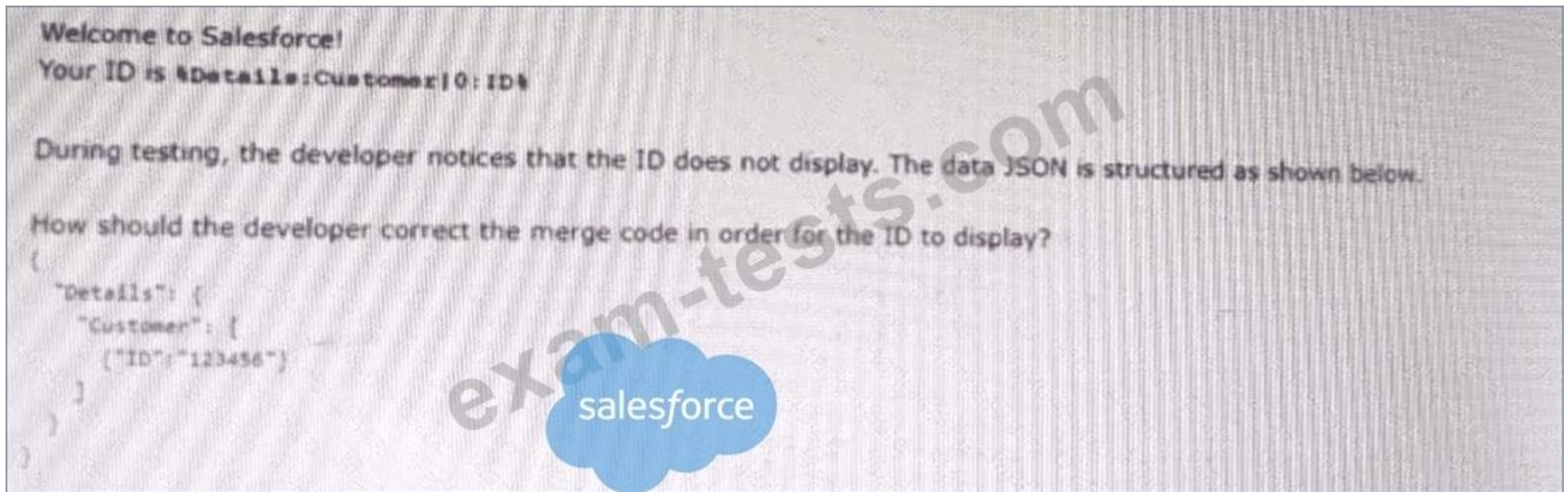
D. Configure the Component Visibility in the Custom Component

Answer: B (LEAVE A REPLY)

The FlexCard Player component is used to display FlexCards in a community page. The developer needs to add the FlexCard's API name to the FlexCard Player component to configure the FlexCard for deployment in a community.

NEW QUESTION: 78

A developer is creating an OmniScript that provisions trail orgs to their customers. The following text block in the Omniscript uses a merge code to display the ID for the new trail org:



- A. To %sDetails: Customer|0: ID\$
- B. To %Details: Customer|1: ID%
- C. To %%sDetails: Customer|0: ID%%
- D. To %Details: Customer|n: ID%

Answer: ([SHOW ANSWER](#))

The correct syntax for merge codes in OmniScript is %%sNodeName%%, where NodeName is the name of the JSON node in the data source. The other options are either missing the s or using incorrect symbols.

NEW QUESTION: 79

Which type of DataRaptor is used specifically to insert or update Salesforce records?

- A. DataRaptor Extract
- B. DataRaptor Transform
- C. DataRaptor Load
- D. DataRaptor Turbo Extract

Answer: C ([LEAVE A REPLY](#))

A DataRaptor Load is used to write data into Salesforce, performing inserts, updates, or upserts on specified objects based on the input data provided.

NEW QUESTION: 80

A developer needs to limit the of a DataRaptorExtract to a maximum of one result. How should the developer configure this?

- A. Define a formula with the Filter function
- B. Use a Custom Output Type when creating the DataRaptor
- C. Use the LIMIT filter on the Extract definition
- D. Set the Limit Property on the Action that calls the DataRaptor Extract.

Answer: C ([LEAVE A REPLY](#))

Using the LIMIT filter on the Extract definition allows the developer to specify the maximum number of results to return from the DataRaptor Extract.

Defining a formula with the Filter function is used to filter the results based on a condition. Using a Custom Output Type when creating the DataRaptor is

used to define the structure of the output data. Setting the Limit Property on the Action that calls the DataRaptor Extract has no effect on limiting the results.

NEW QUESTION: 81

An OmniScript displays data from an API using Integration Procedure, but some of the data is missing. Which two configuration errors could cause this? Choose 2 answers

- A. The element name for the missing data does not match the JSON node key in the Integration Procedure Response.
- B. The Integration Procedure Preview Input Parameters do not match the JSON sent from the OmniScript.
- C. The JSOW sent from the Integration Procedure Action does not match any of the Original Input for the Integration Procedure
- D. The missing data is trimmed in the Integration Procedure Action Response JSON Path.

Answer: A,C (LEAVE A REPLY)

The element name for the missing data must match the JSON node key in the Integration Procedure Response, otherwise the data will not be mapped correctly. The JSON sent from the Integration Procedure Action must match one of the Original Input for the Integration Procedure, otherwise the Integration Procedure will not execute properly. The Integration Procedure Preview Input Parameters and the Response JSON Path have no effect on displaying data in OmniScript.

NEW QUESTION: 82

In OmniScript, what is the purpose of the "Language" attribute when creating a new OmniScript?

- A. It determines which programming language is used for custom Apex extensions
- B. It allows multiple translated versions of the same OmniScript to exist, selected based on user locale
- C. It defines the syntax used for Formula elements
- D. It sets the default currency format for the OmniScript

Answer: B (LEAVE A REPLY)

The Language attribute, combined with Type and Sub Type, uniquely identifies an OmniScript version, allowing developers to create multiple language-specific versions that are served based on the user's locale settings.

Valid Plat-Dev-210 Dumps shared by BraindumpsPass.com for Helping Passing Plat-Dev-210 Exam! BraindumpsPass.com now offer the **newest Plat-Dev-210 exam dumps**, the BraindumpsPass.com Plat-Dev-210 exam **questions have been updated** and **answers have been corrected** get the **newest** BraindumpsPass.com Plat-Dev-210 dumps with Test Engine here: <https://www.braindumpspass.com/Salesforce/Plat-Dev-210-practice-exam-dumps.html> (130 Q&As Dumps, **40%OFF Special Discount: Exam-Tests**)