# MB-820$^{Q\&As}$

Microsoft Dynamics 365 Business Central Developer

## Pass Microsoft MB-820 Exam with 100% Guarantee

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

**https://www.pass4itsure.com/mb-820.html**

100% Passing Guarantee
100% Money Back Assurance

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

**QUESTION 1**

DRAG DROP

You are developing a test application to test the posting process of a sales order. You must provide the following implementation:

1.

Specify the value of post options (dialog: Ship, Invoice, Shipand; Invoice) as Invoice.

2.

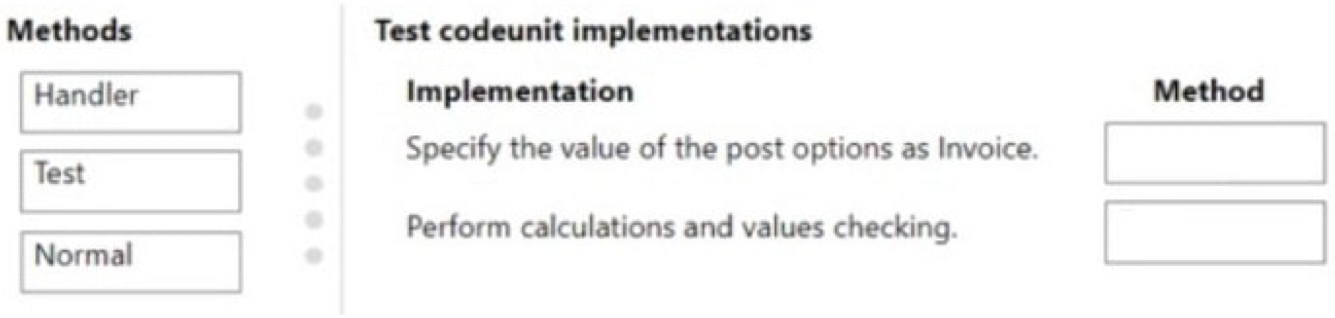
Perform calculations and values checking.

You need to complete the development of the test codeunit.

Which methods should you use? To answer, move the appropriate methods to the correct implementation. You may use each method once, more than once, or not at all. You may need to move the split bar between panes or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:



**Methods**

Handler

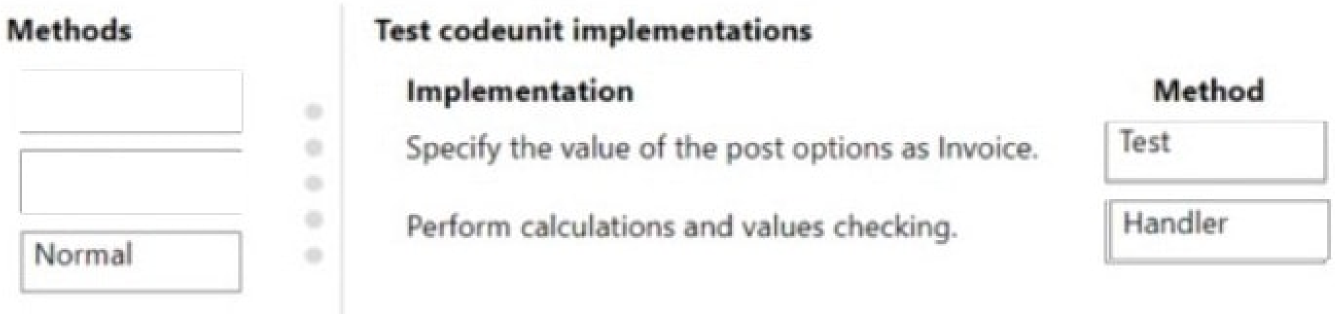
Test

Normal

**Test codeunit implementations**

| Implementation | Method |
|---|---|
| Specify the value of the post options as Invoice. | |
| Perform calculations and values checking. | |

Correct Answer:



**Methods**

Normal

**Test codeunit implementations**

| Implementation | Method |
|---|---|
| Specify the value of the post options as Invoice. | Test |
| Perform calculations and values checking. | Handler |

Specify the value of the post options as Invoice:

Test

Perform calculations and values checking:

Handler

In the context of Microsoft Dynamics 365 Business Central testing, the \\'Test\\' attribute is used to mark a method as a test method. This is where you would specify the action or the behavior you\\'re testing ?in this case, setting the post options

as Invoice. It\\'s within these test methods that you would simulate setting the posting option to "Invoice" programmatically.

For performing calculations and checking values, you would use \\'Handler\\' methods to handle specific business events or conditions that occur within the system, such as before or after posting a document. These handlers can ensure that

calculations are done correctly and that all validation checks pass before the document is posted. The \\'Normal\\' method would be a standard method that could be involved in the posting process, ensuring that all business logic is correctly

applied and that the calculations and value checks are as expected.

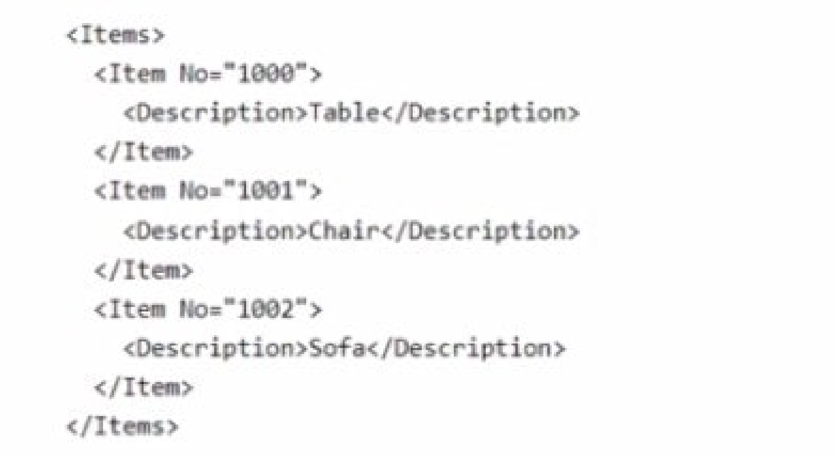
In a test codeunit, you would typically have test methods that call these handler and normal methods to verify the business logic in various scenarios, such as posting with different options or checking the results of calculations under different

conditions.

**QUESTION 2**

HOTSPOT

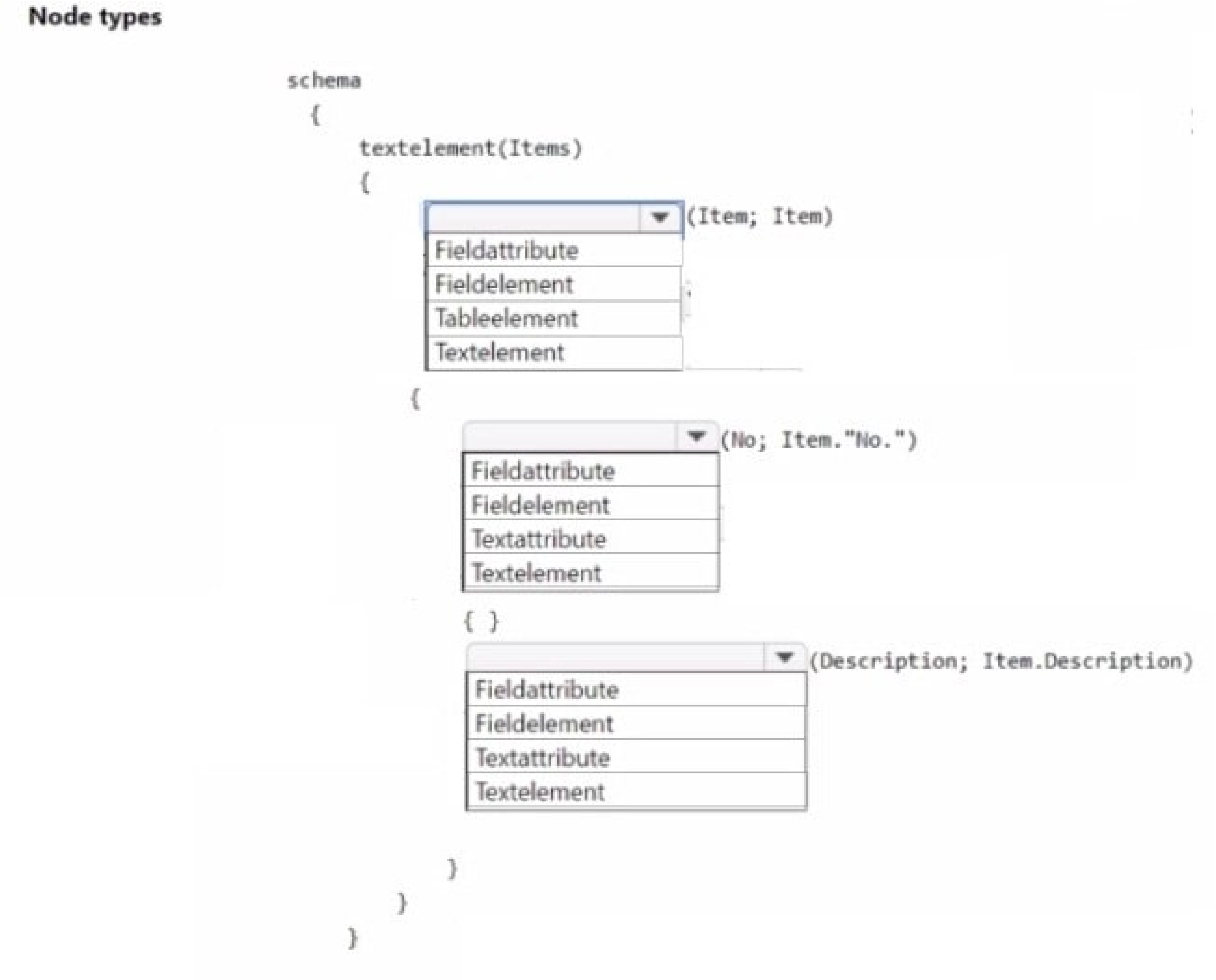
You have the following XML file sample for the Items list:



```
<Items>
  <Item No="1000">
    <Description>Table</Description>
  </Item>
  <Item No="1001">
    <Description>Chair</Description>
  </Item>
  <Item No="1002">
    <Description>Sofa</Description>
  </Item>
</Items>
```

You plan to create the next XML file by using an XMLport object. You need to complete the code segment to export the file in the required format How should you complete the code segment? To answer, select the appropriate options in the answer area.

Hot Area:



Node types

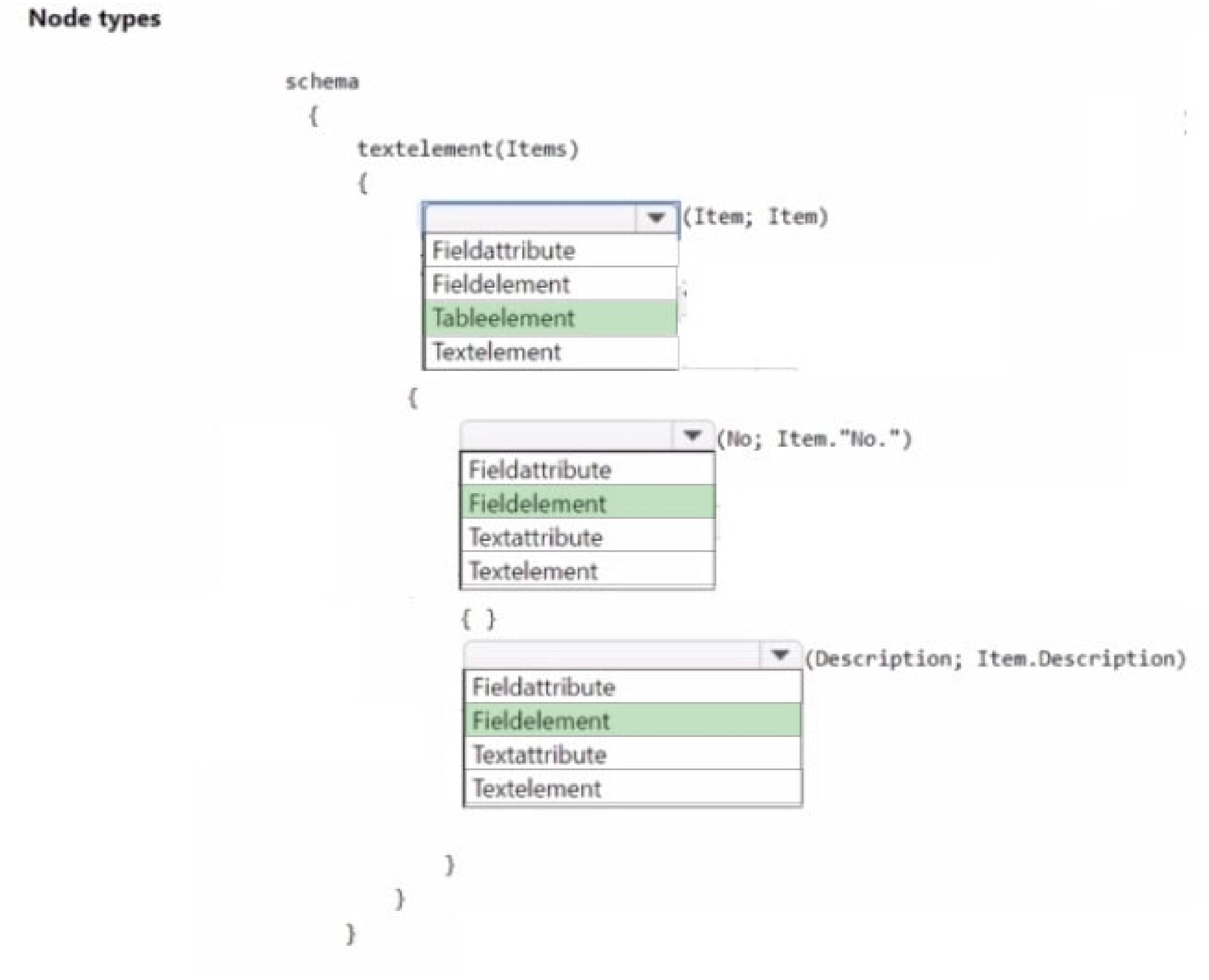
```
schema
 {
     textelement(Items)
     {
                        (Item; Item)

          {
                        (No; Item."No.")

               { }
                        (Description; Item.Description)

          }
     }
 }
```

Correct Answer:



Node types

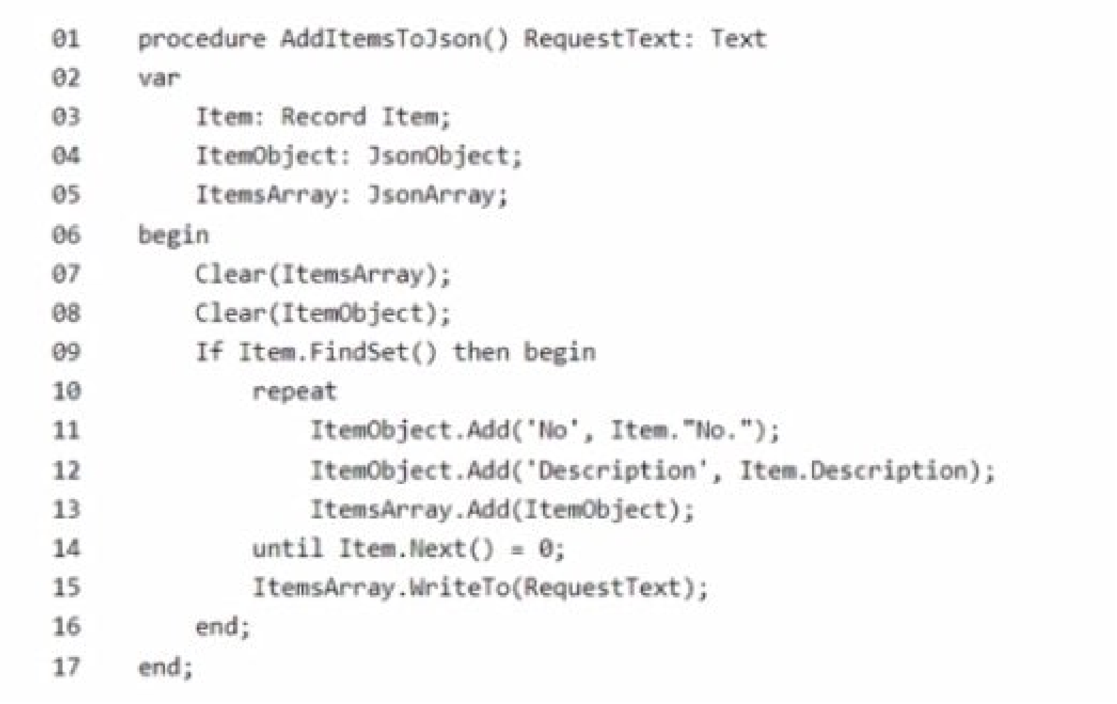
```
schema
 {
     textelement(Items)
     {
             (Item; Item)
             {
                   (No; Item."No.")
                   { }
                   (Description; Item.Description)

                }
            }
       }
```

**QUESTION 3**

HOTSPOT

You create an \\'AddItemsToJson" procedure and publish it.



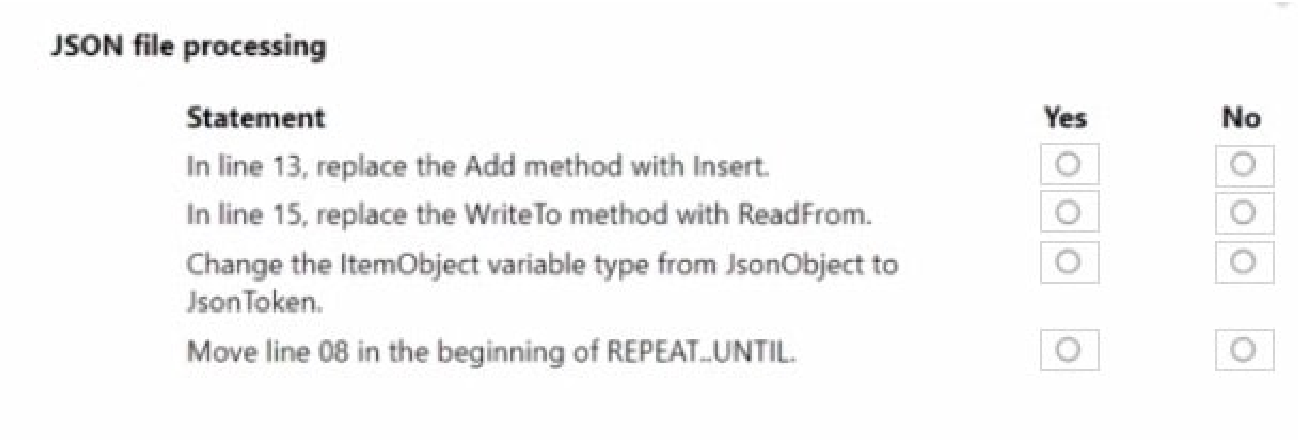
```
01    procedure AddItemsToJson() RequestText: Text
02    var
03        Item: Record Item;
04        ItemObject: JsonObject;
05        ItemsArray: JsonArray;
06    begin
07        Clear(ItemsArray);
08        Clear(ItemObject);
09        If Item.FindSet() then begin
10            repeat
11                ItemObject.Add('No', Item."No.");
12                ItemObject.Add('Description', Item.Description);
13                ItemsArray.Add(ItemObject);
14            until Item.Next() = 0;
15            ItemsArray.WriteTo(RequestText);
16        end;
17    end;
```

The procedure fails to run.

You need to fix the errors in the code.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

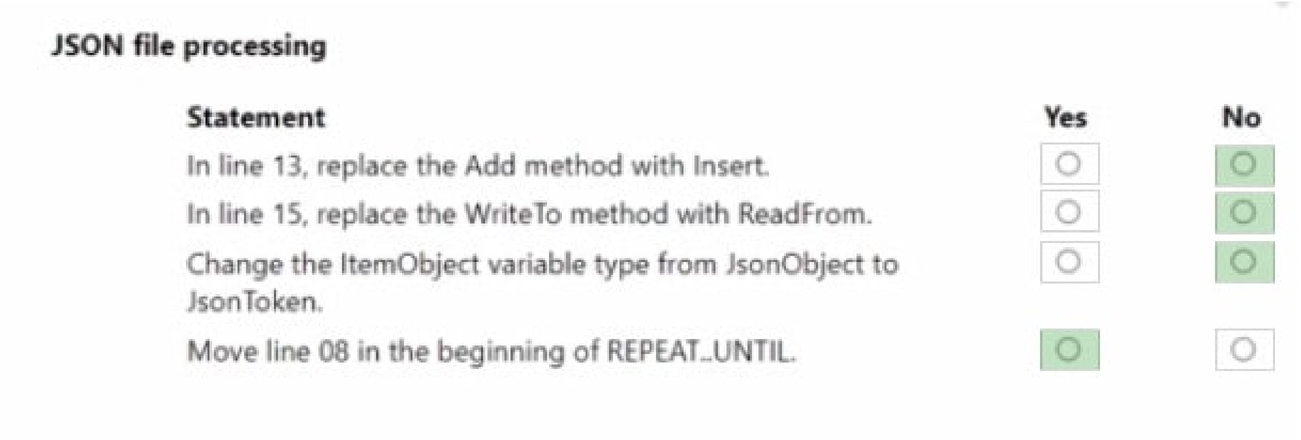
Hot Area:



JSON file processing

| Statement | Yes | No |
|---|---|---|
| In line 13, replace the Add method with Insert. | ○ | ○ |
| In line 15, replace the WriteTo method with ReadFrom. | ○ | ○ |
| Change the ItemObject variable type from JsonObject to JsonToken. | ○ | ○ |
| Move line 08 in the beginning of REPEAT..UNTIL. | ○ | ○ |

Correct Answer:



JSON file processing

| Statement | Yes | No |
|---|---|---|
| In line 13, replace the Add method with Insert. | ○ | [✓ ○] |
| In line 15, replace the WriteTo method with ReadFrom. | ○ | [✓ ○] |
| Change the ItemObject variable type from JsonObject to JsonToken. | ○ | [✓ ○] |
| Move line 08 in the beginning of REPEAT..UNTIL. | [✓ ○] | ○ |

In line 13, replace the Add method with Insert. = NO In line 15, replace the WriteTo method with ReadFrom. = NO Change the ItemObject variable type from JsonObject to JsonToken. = NO Move line 08 in the beginning of REPEAT .. UNTIL.

= YES The provided code is intended to serialize a list of items from the Item table into a JSON array format. Here is a breakdown of the code and the necessary corrections:

In line 13, "ItemsArray.Add(ItemObject)": This line is correctly using the Add method to add the ItemObject to the ItemsArray. The Add method is the correct method to use for adding items to a JsonArray. Therefore, there is no need to replace

Add with Insert.

In line 15, "ItemsArray.WriteTo(RequestText)": The WriteTo method is used correctly to serialize the ItemsArray into a JSON formatted string and store it in the RequestText variable. The ReadFrom method is used for the opposite operation,

i.e., to deserialize a JSON formatted string into a JsonArray, which is not the goal in this context. Hence, no change is needed here. Change the ItemObject variable type from JsonObject to JsonToken: The ItemObject variable is intended to

hold JSON objects representing individual items, making JsonObject the appropriate type. JsonToken is not a type used in this context within AL for Business Central, and thus the variable type should remain as JsonObject.

Move line 08, "Clear(ItemObject)": This line should be moved inside the repeat loop to ensure that the ItemObject is cleared for each item in the loop. Placing it before the repeat would only clear it once before the loop starts, which could lead

to incorrect serialization as the previous item\\'s properties would not be cleared from the ItemObject.

The logic for serializing records into JSON is a common operation when interfacing with APIs or web services in Business Central, and the pattern shown in the code is typical for such operations.

**QUESTION 4**

DRAG DROP

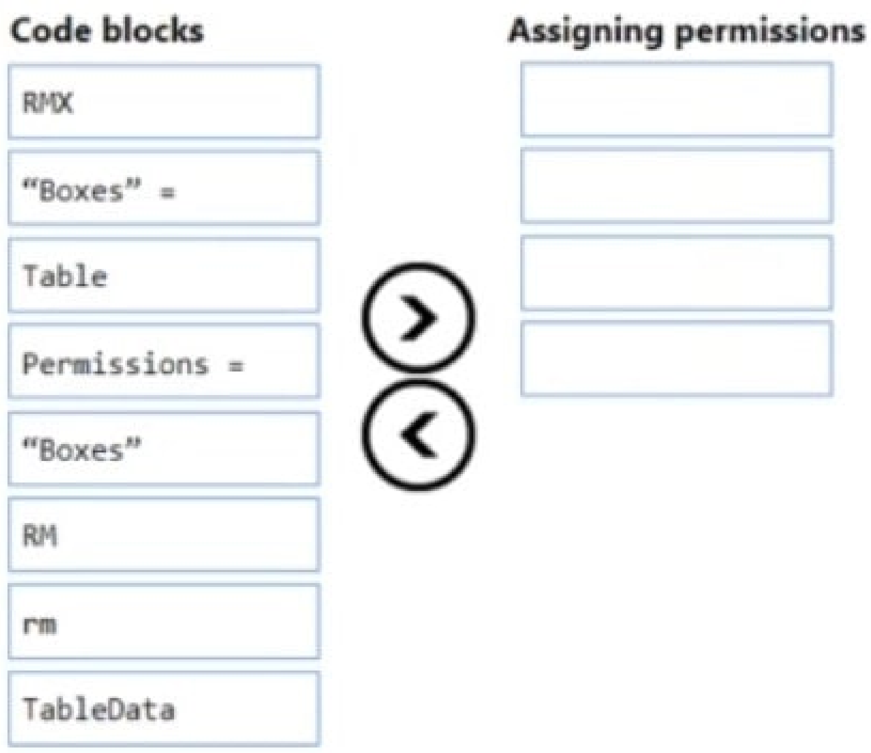
You create a codeunit that works with a table named Boxes. You plan to filter the records and then modify them.

You get an error that you do not have permission to work with the Boxes table.

You need to assign the Indirect permissions for the Boxes table to the codeunit.

Which four code blocks should you use in sequence to assign the correct permission? To answer, move the appropriate code blocks from the list of code blocks to the answer area and arrange them in the correct order.

Select and Place:

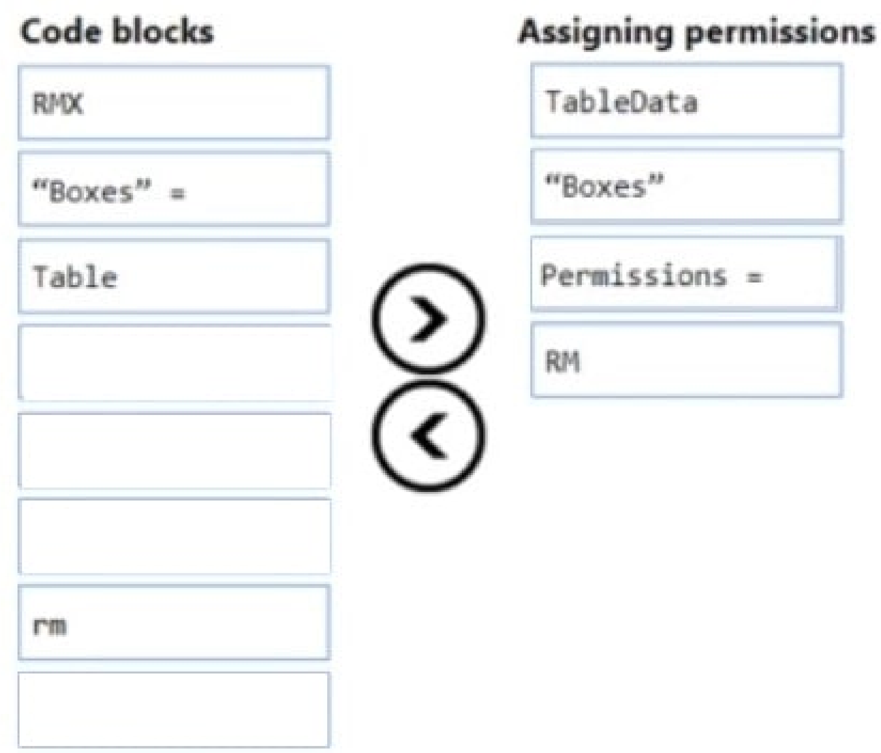


Code blocks

- RMX
- “Boxes” =
- Table
- Permissions =
- “Boxes”
- RM
- rm
- TableData

Assigning permissions

Correct Answer:



To assign the indirect permissions for the Boxes table to the codeunit, use the following code blocks in sequence: TableData "Boxes" = Permissions RIM Assigning permissions:In Business Central, to assign permissions within a codeunit, you need to specify the table that the permissions apply to, followed by the type of permission. The sequence starts by indicating that we are defining table data permissions (TableData ). Then, we specify the table in question ("Boxes" =). After that, we state that we are setting permissions (Permissions). Finally, we assign the RIM permissions, which stands for Read, Insert, and Modify permissions. The Indirect permission allows the codeunit to read, insert, and modify records in the Boxes table indirectly, meaning these operations can be performed by the codeunit when it is called by a user who has direct permissions for these operations.

**QUESTION 5**

You create a Business Central report.

You need to insert values on the Request page to be saved for the next time the report is run.

What should you do?

A. Set the Transact! on Type property to Update.

B. Declare a Savevalues\\' variable and assign it to true on the OnOpenPage () trigger.

C. Set the Use Request Page property to true.

D. Set the SaveValues property to true.

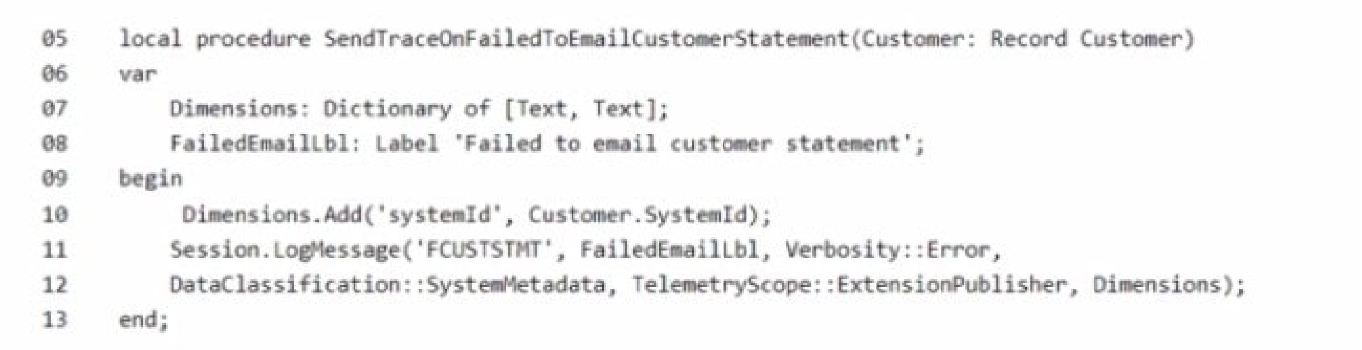
Correct Answer: D

To ensure that the values inserted on the Request page of a Business Central report are saved for the next time the report is run, the SaveValues property (D) should be set to true. This property is available on the Request page of the report and, when set to true, allows the system to remember the values entered by the user, so they do not have to reenter them each time they run the report. This feature enhances user experience by reducing repetitive data entry and ensuring consistency in report parameters across multiple executions. The other options mentioned, such as setting the Transaction Type property to Update (A) or declaring a Savevalues variable in the OnOpenPage trigger (B), are not directly related to saving user input on a report\\'s Request page.

**QUESTION 6**

HOTSPOT

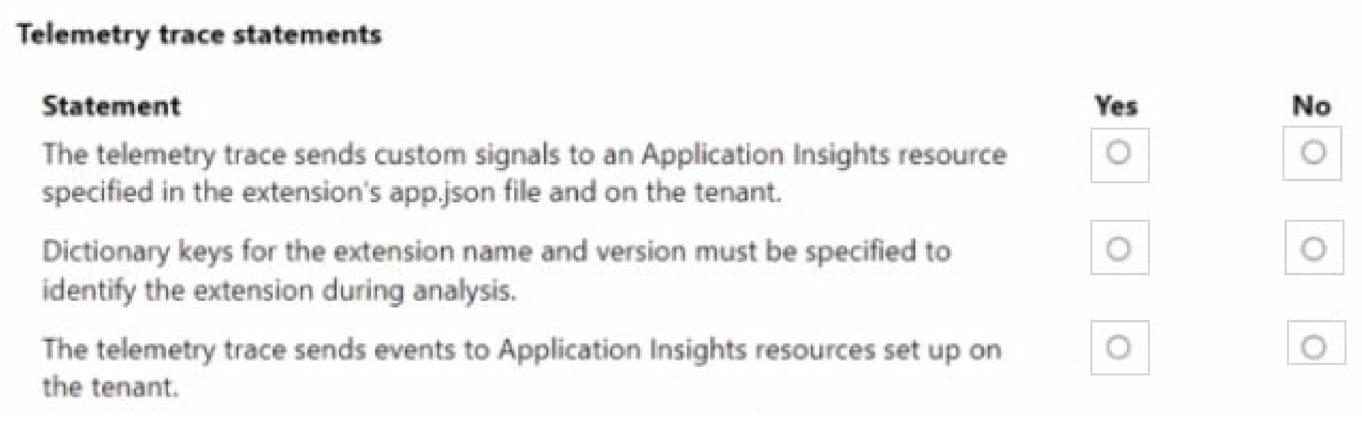
A company is setting up a custom telemetry trace signal to send traces on failed customer statement emails.



```
local procedure SendTraceOnFailedToEmailCustomerStatement(Customer: Record Customer)
var
    Dimensions: Dictionary of [Text, Text];
    FailedEmailLbl: Label 'Failed to email customer statement';
begin
     Dimensions.Add('systemId', Customer.SystemId);
    Session.LogMessage('FCUSTSTMT', FailedEmailLbl, Verbosity::Error,
    DataClassification::SystemMetadata, TelemetryScope::ExtensionPublisher, Dimensions);
end;
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

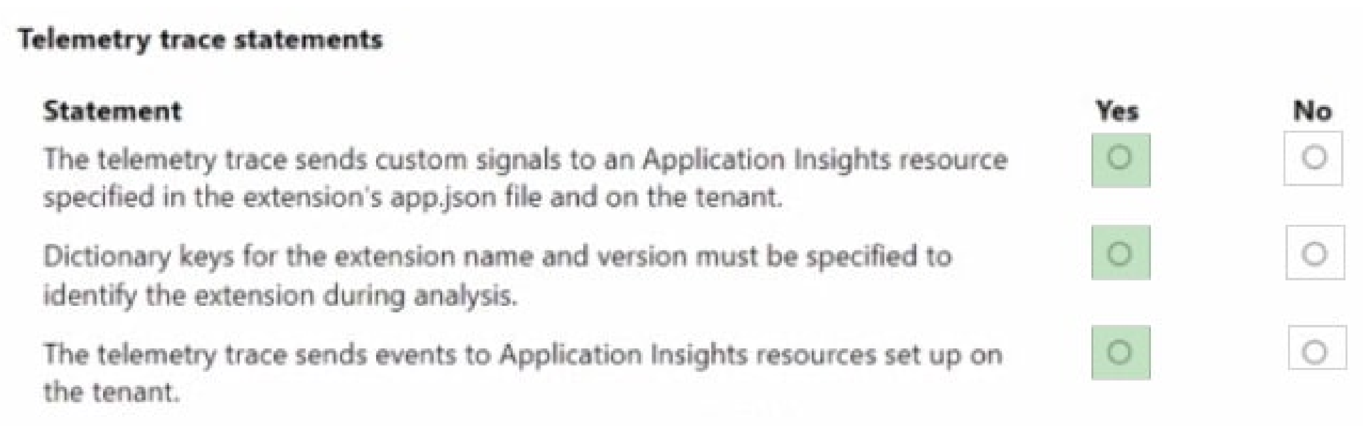
Hot Area:



**Telemetry trace statements**

| Statement | Yes | No |
|---|---|---|
| The telemetry trace sends custom signals to an Application Insights resource specified in the extension's app.json file and on the tenant. | ○ | ○ |
| Dictionary keys for the extension name and version must be specified to identify the extension during analysis. | ○ | ○ |
| The telemetry trace sends events to Application Insights resources set up on the tenant. | ○ | ○ |

Correct Answer:



**Telemetry trace statements**

| Statement | Yes | No |
|---|---|---|
| The telemetry trace sends custom signals to an Application Insights resource specified in the extension's app.json file and on the tenant. | [✓ ○] | ○ |
| Dictionary keys for the extension name and version must be specified to identify the extension during analysis. | [✓ ○] | ○ |
| The telemetry trace sends events to Application Insights resources set up on the tenant. | [✓ ○] | ○ |

The telemetry trace sends custom signals to an Application Insights resource specified in the extension\\'s app.json file and on the tenant. = YES Dictionary keys for the extension name and version must be specified to identify the extension

during analysis. = YES

The telemetry trace sends events to Application Insights resources set up on the tenant. = YES

Telemetry in Business Central allows developers to collect custom telemetry for extensions using Application Insights. The telemetry trace is used to send custom signals to an Application Insights resource. This resource is typically specified

in the app.json file of the extension and must be configured on the tenant where the extension is installed. The use of dictionary keys for the extension name and version is a best practice to identify the extension during analysis in Application

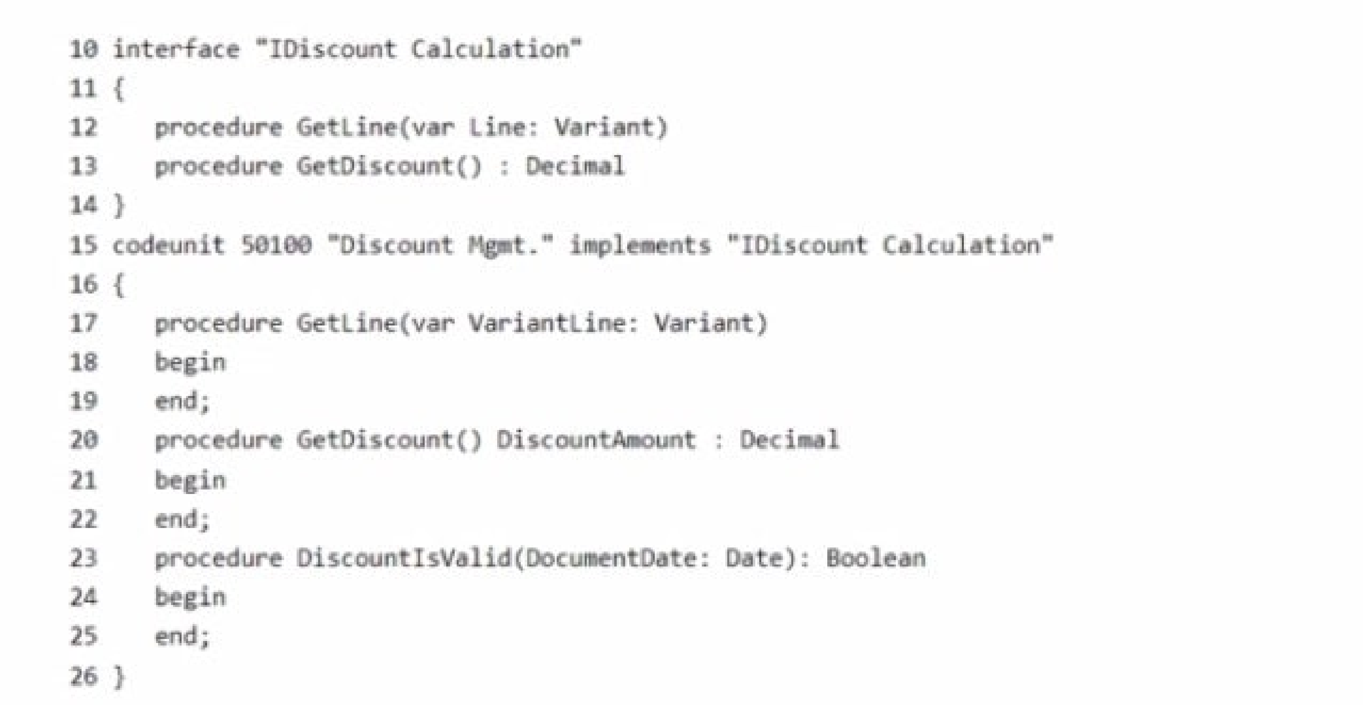
Insights. These keys can be added to the telemetry trace to ensure that when the data is collected, it\\'s clear which extension the data is associated with.

Finally, it is correct that the telemetry trace sends events to Application Insights resources that are set up on the tenant, enabling the collection and analysis of telemetry at the tenant level.

**QUESTION 7**

HOTSPOT

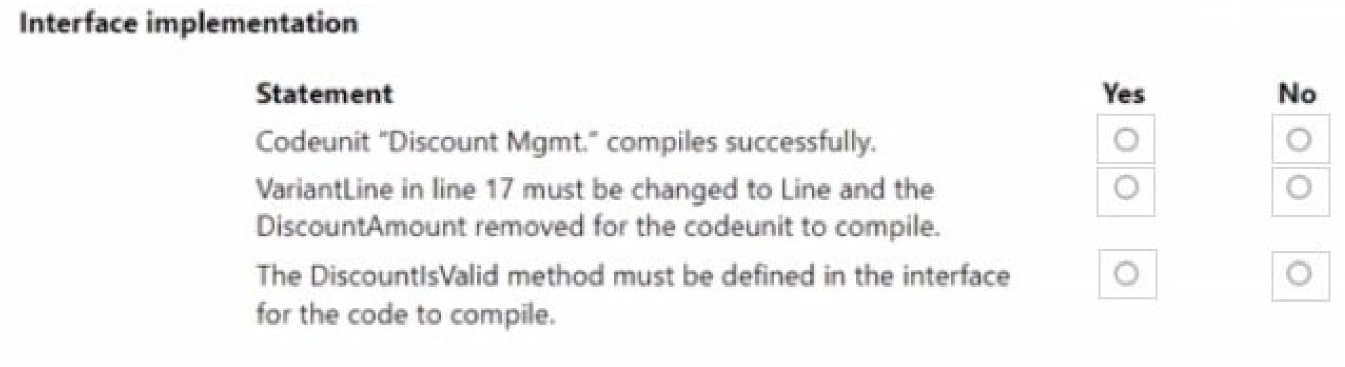
You have a per tenant extension that contains the following code.



```
10 interface "IDiscount Calculation"
11 {
12     procedure GetLine(var Line: Variant)
13     procedure GetDiscount() : Decimal
14 }
15 codeunit 50100 "Discount Mgmt." implements "IDiscount Calculation"
16 {
17     procedure GetLine(var VariantLine: Variant)
18     begin
19     end;
20     procedure GetDiscount() DiscountAmount : Decimal
21     begin
22     end;
23     procedure DiscountIsValid(DocumentDate: Date): Boolean
24     begin
25     end;
26 }
```

For each of the following statements, select Yes if the statement is true. Otherwise, select No. NOTE: Each correct selection is worth one point.

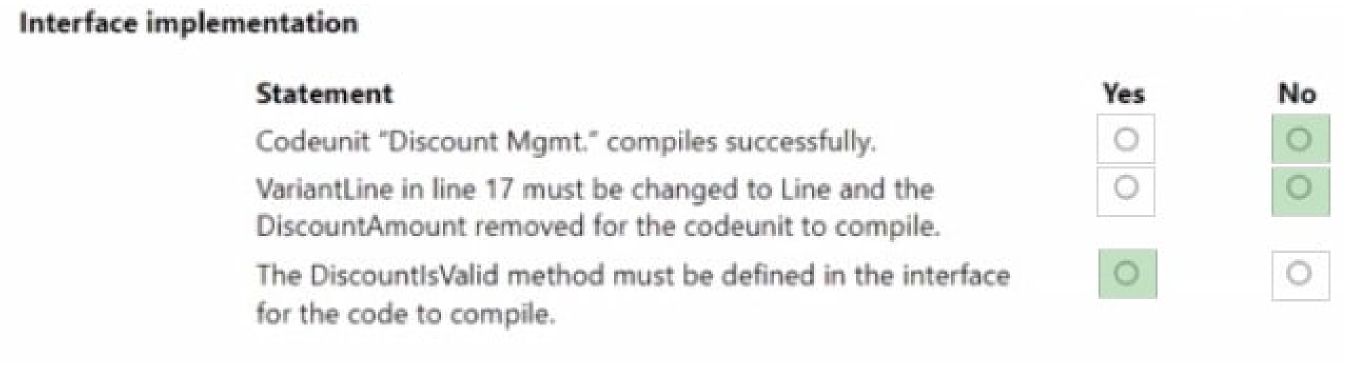
Hot Area:



Interface implementation

| Statement | Yes | No |
|---|---|---|
| Codeunit "Discount Mgmt." compiles successfully. | ○ | ○ |
| VariantLine in line 17 must be changed to Line and the DiscountAmount removed for the codeunit to compile. | ○ | ○ |
| The DiscountIsValid method must be defined in the interface for the code to compile. | ○ | ○ |

Correct Answer:



Interface implementation

| Statement | Yes | No |
|---|---|---|
| Codeunit "Discount Mgmt." compiles successfully. | ○ | [✓ ○] |
| VariantLine in line 17 must be changed to Line and the DiscountAmount removed for the codeunit to compile. | ○ | [✓ ○] |
| The DiscountIsValid method must be defined in the interface for the code to compile. | [✓ ○] | ○ |

Codeunit "Discount Mgmt." compiles successfully. = NO VariantLine in line 17 must be changed to Line and the DiscountAmount removed for the codeunit to compile. = NO

The DiscountIsValid method must be defined in the interface for the code to compile. = YES

The codeunit "Discount Mgmt." will not compile successfully as is because the DiscountIsValid method is not defined in the "IDiscount Calculation" interface, yet it is being declared in the codeunit which implements this interface. AL requires

that all procedures in the codeunit that implements an interface must be defined in the interface itself.

The VariantLine in line 17 does not need to be changed to Line, nor does the DiscountAmount need to be removed for the codeunit to compile. These are valid declarations in AL and they are correctly implemented in the codeunit. The Variant

data type in AL is used to handle various data types and DiscountAmount is a valid return type for a procedure.

For the code to compile successfully, the DiscountIsValid method must be included in the interface because AL enforces that any codeunit implementing an interface must implement all the methods defined in that interface.

**QUESTION 8**

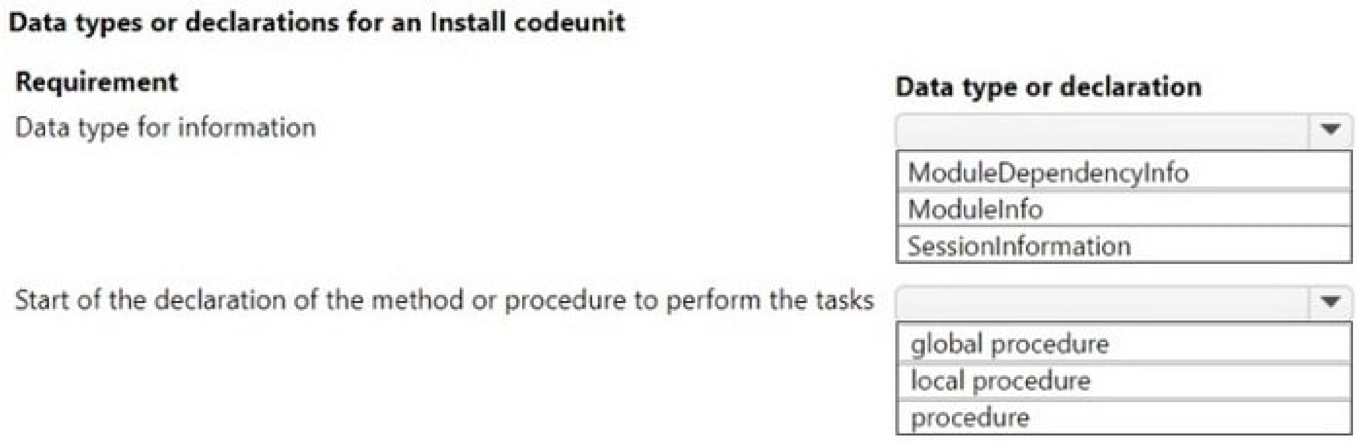
HOTSPOT

You need to create the Install codeunit that is required in the extension used for installing or updating the Housekeeping app.

Which data type or declaration should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

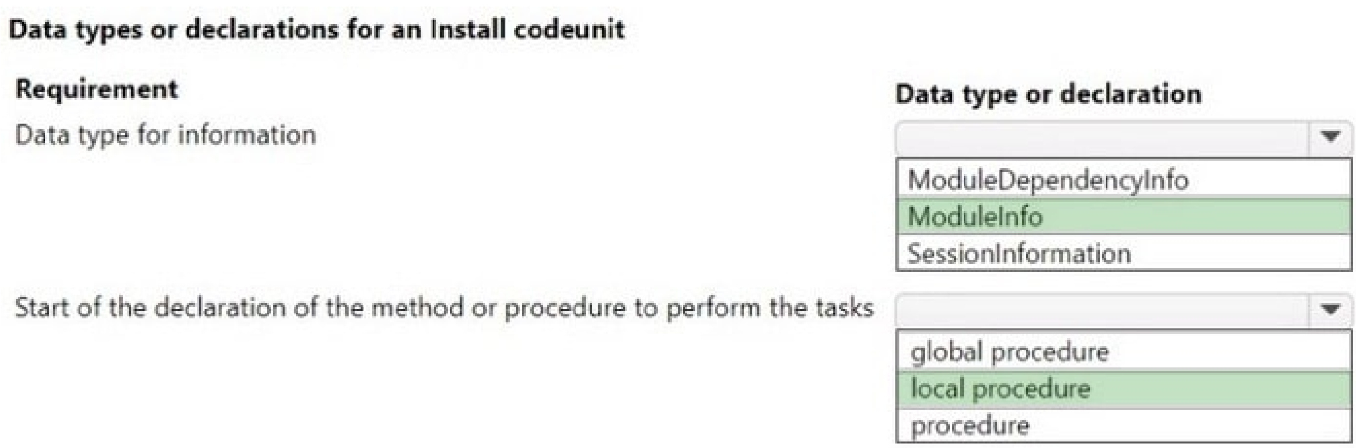
Hot Area:



**Data types or declarations for an Install codeunit**

| Requirement | Data type or declaration |
|---|---|
| Data type for information | ModuleDependencyInfo<br>ModuleInfo<br>SessionInformation |
| Start of the declaration of the method or procedure to perform the tasks | global procedure<br>local procedure<br>procedure |

Correct Answer:



**Data types or declarations for an Install codeunit**

| Requirement | Data type or declaration |
|---|---|
| Data type for information | ModuleDependencyInfo<br>ModuleInfo<br>SessionInformation |
| Start of the declaration of the method or procedure to perform the tasks | global procedure<br>local procedure<br>procedure |

Box 1: ModuleInfo

Data type for information

ModuleInfo

Represents information about an application consumable from AL.

Scenario: Department-specific requirements. Housekeeping department

The department requires the development of an extension with a new API page named RoomsAPI.

*

The housekeeping team will use RoomsAPI to publish room details, update when work is complete, or provide repair notifications from the canvas app.

*-> This custom API page must expose a custom table named Rooms and have an ID 50000. The table must be able to update from the PMS. The PMS team must know the *endpoint to connect to the custom API*.

Note: Data Types and Methods in AL

The following data types are available as part of the AL Language. Each data type has various methods that support it.

*

ModuleInfo

Represents information about an application consumable from AL.

Incorrect:

*

ModuleDependencyInfo

Provides information about a dependent module.

*

SessionInformation

Is a complex data type for exposing Session information into AL.

Box 2: local procedure Procedure scope To declare a local method, start the declaration with local:

local procedure Mymethod(); To declare a global method, omit local: procedure Mymethod();

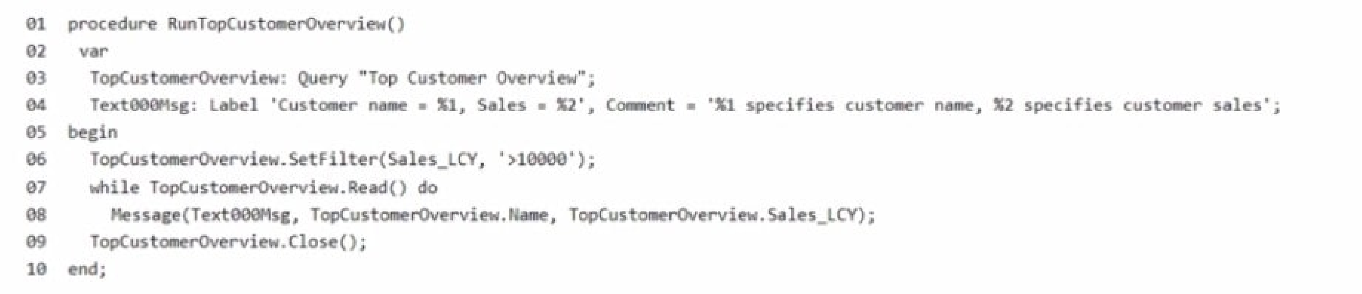
Scenario: The code required to perform tasks cannot be accessible from other parts of the application. Reference: https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/methods-auto/library https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/devenv-al-methods

## QUESTION 9

HOTSPOT

You create a query that contains a procedure to display the top customers.

The procedure breaks at runtime.



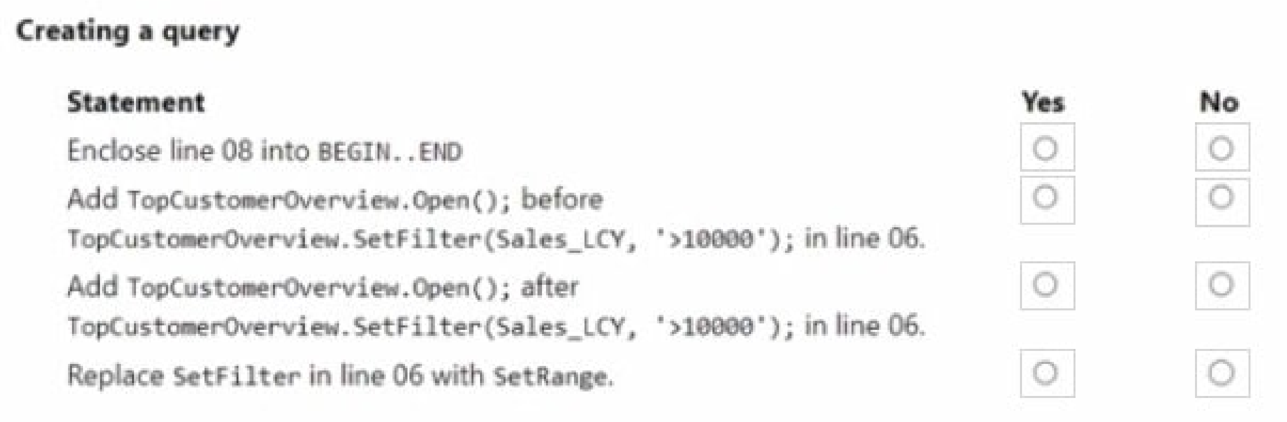
```
01  procedure RunTopCustomerOverview()
02   var
03    TopCustomerOverview: Query "Top Customer Overview";
04    Text000Msg: Label 'Customer name = %1, Sales = %2', Comment = '%1 specifies customer name, %2 specifies customer sales';
05  begin
06    TopCustomerOverview.SetFilter(Sales_LCY, '>10000');
07    while TopCustomerOverview.Read() do
08      Message(Text000Msg, TopCustomerOverview.Name, TopCustomerOverview.Sales_LCY);
09    TopCustomerOverview.Close();
10  end;
```

You need to fix the code.

For each of the following statements, select Yes if the statement is true. Otherwise, select No.

NOTE: Each correct selection is worth one point.

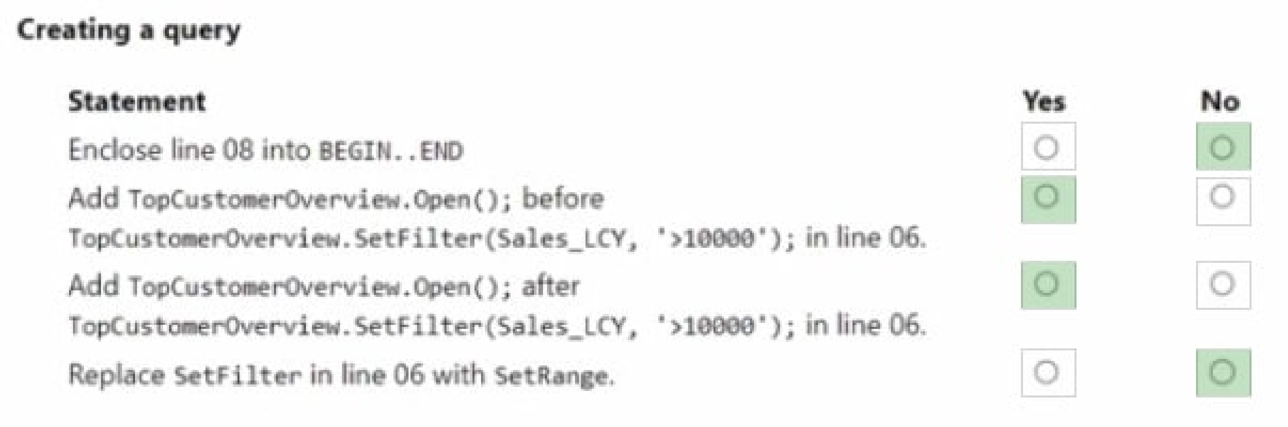
Hot Area:



### Creating a query

| Statement | Yes | No |
|---|---|---|
| Enclose line 08 into BEGIN..END | ○ | ○ |
| Add TopCustomerOverview.Open(); before TopCustomerOverview.SetFilter(Sales_LCY, '>10000'); in line 06. | ○ | ○ |
| Add TopCustomerOverview.Open(); after TopCustomerOverview.SetFilter(Sales_LCY, '>10000'); in line 06. | ○ | ○ |
| Replace SetFilter in line 06 with SetRange. | ○ | ○ |

Correct Answer:



Creating a query

| Statement | Yes | No |
|---|---|---|
| Enclose line 08 into `BEGIN..END` | ○ | ● |
| Add `TopCustomerOverview.Open();` before `TopCustomerOverview.SetFilter(Sales_LCY, '>10000');` in line 06. | ● | ○ |
| Add `TopCustomerOverview.Open();` after `TopCustomerOverview.SetFilter(Sales_LCY, '>10000');` in line 06. | ● | ○ |
| Replace `SetFilter` in line 06 with `SetRange`. | ○ | ● |

Enclose line 08 into BEGIN .. END = NO Add TopCustomerOverview.Open(); before = YES TopCustomerOverview.SetFilter(Sales_LCY, \\'>10000\\'); in line 06. Add TopCustomerOverview.Open(); after TopCustomerOverview.SetFilter(Sales_LCY, \\'>10000\\'); in line 06. = YES Replace SetFilter in line 06 with SetRange. = NO

The code provided has a runtime error because the query TopCustomerOverview must be opened before it can be read from. Therefore, TopCustomerOverview.Open(); should be added before trying to read from the query, which is not present in the code. Enclosing line 08 into a BEGIN .. END block is unnecessary because it is a single statement, and AL does not require a BEGIN .. END block for single statements within trigger or procedure bodies. TopCustomerOverview.SetFilter(Sales_LCY, \\'>10000\\'); is a correct method to set a filter for the query, and using SetRange instead is not necessary unless the requirement is specifically to set a range of values, which is not indicated in the procedure\\'s description. In summary, for the procedure to run correctly, the query must be opened after setting the filter and before attempting to read from it. The SetFilter method is correct for the intended operation, and there\\'s no requirement to use SetRange or to enclose the Message call in a BEGIN .. END block.

**QUESTION 10**

HOTSPOT

A company plans to customize its per tenant extension reports. The company has the following requirements for the customization:

1.

Child data items must not be displayed on the request page for some master detail reports.

2.

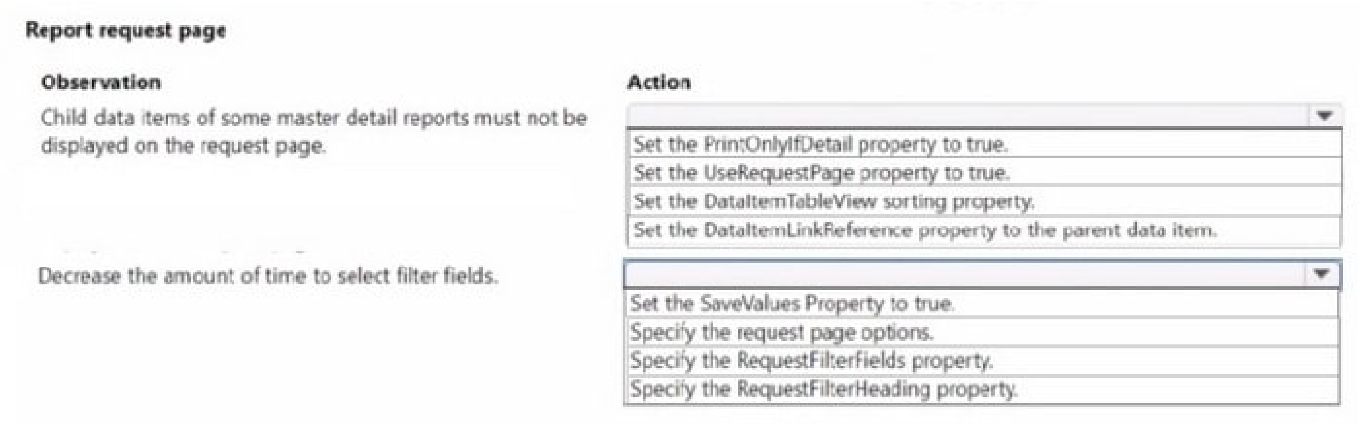
Selecting key filter fields takes users too much time. The customization must decrease the amount of time to select the fields.

You need to optimize the report request page.

Which actions should you configure? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

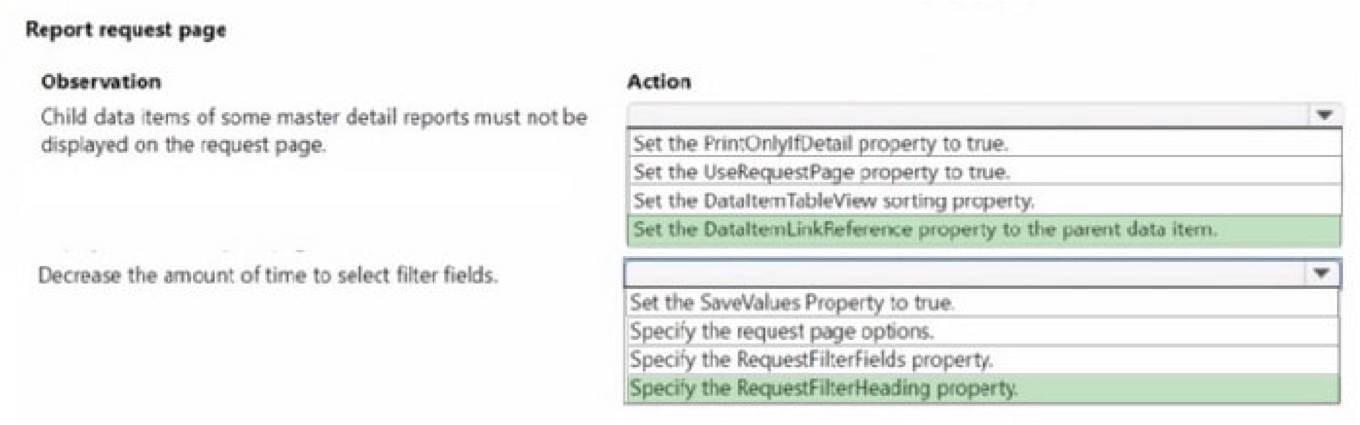
Hot Area:



Report request page

| Observation | Action |
|---|---|
| Child data items of some master detail reports must not be displayed on the request page. | ▼<br>Set the PrintOnlyIfDetail property to true.<br>Set the UseRequestPage property to true.<br>Set the DataItemTableView sorting property.<br>Set the DataItemLinkReference property to the parent data item. |
| Decrease the amount of time to select filter fields. | ▼<br>Set the SaveValues Property to true.<br>Specify the request page options.<br>Specify the RequestFilterFields property.<br>Specify the RequestFilterHeading property. |

Correct Answer:



Report request page

| Observation | Action |
|---|---|
| Child data items of some master detail reports must not be displayed on the request page. | ▼<br>Set the PrintOnlyIfDetail property to true.<br>Set the UseRequestPage property to true.<br>Set the DataItemTableView sorting property.<br>[✓ Set the DataItemLinkReference property to the parent data item.] |
| Decrease the amount of time to select filter fields. | ▼<br>Set the SaveValues Property to true.<br>Specify the request page options.<br>Specify the RequestFilterFields property.<br>[✓ Specify the RequestFilterHeading property.] |

For the given requirements, you should configure the following actions:

For child data items not to be displayed on the request page for some master- detail reports, set the DataItemLinkReference property to the parent data item. To decrease the amount of time to select key filter fields, specify the

RequestFilterHeading property.

In Dynamics 365 Business Central, when customizing report request pages, certain properties can be set to control the behavior and display of the report options:

Hiding Child Data Items:The DataItemLinkReference property is used to link a child data item to a parent data item in the data model of a report. Setting this property correctly will ensure that the child data items are related to the correct

parent data item and will be displayed or hidden accordingly on the request page. If the goal is to prevent child data items from being displayed, you need to make sure they are correctly linked and configured to not appear. Optimizing Filter

Field Selection:The RequestFilterHeading property is used to group filter fields on the request page. By specifying this property, you can create a more organized and user-friendly interface, which can significantly speed up the process of

selecting filters. This property allows you to categorize filters into headings, making it quicker and easier for users to find and set the necessary filters for the report.

By adjusting these properties on the report request page as part of the per tenant extension customization, you will address the company\\'s requirements to optimize the user experience when running reports.