



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.geekcert.com/mb-820.html>

100% Passing Guarantee
100% Money Back Assurance

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

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





QUESTION 1

HOTSPOT

You need to create the API page according to the requirements.

How should you complete the code segment? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:



Code segment for API page

```

    page 50100 ItemAPI
{
    PageType = API;
    Caption = 'ItemApi';
    APIPublisher = 'contoso';
    APIGroup = 'sales';
    APIVersion = 'v1.0';
    EntityName = 'item';
    EntitySetName = 'items';
    SourceTable = Item;
    ODataKeyFields = SystemId;

```

InsertAllowed = false;
DelayedInsert = true;
DataAccessIntent = ReadOnly;

ModifyAllowed = false;
Editable = false;
UsageCategory = Lists;

```

    layout
    {
        area(Content)
        {
            repeater(GroupName)
            {
                field(no; Rec."No.")
                {
                }
                field(salesTotal; Rec."Sales (LCY)")
                {
                }
            }
        }
    }
}

```



Correct Answer:



Code segment for API page

```

    page 50100 ItemAPI
{
    PageType = API;
    Caption = 'ItemApi';
    APIPublisher = 'contoso';
    APIGroup = 'sales';
    APIVersion = 'v1.0';
    EntityName = 'item';
    EntitySetName = 'items';
    SourceTable = Item;
    ODataKeyFields = SystemId;

```

```

InsertAllowed = false;
DelayedInsert = true;
DataAccessIntent = ReadOnly;

```

```

ModifyAllowed = false;
Editable = false;
UsageCategory = Lists;

```

```

    layout
    {
        area(Content)
        {
            repeater(GroupName)
            {
                field(no; Rec."No.")
                {
                }
                field(salesTotal; Rec."Sales (LCY)")
                {
                }
            }
        }
    }
}

```



Box 1: `DataAccessIntent = readOnly;`

Scenario:

Contoso, Ltd. must create an API in Business Central to expose item details to the mobile application.

The API must have the lowest possible impact on the production environment when used during working hours.

The API must only support Get operations.

DataAccessIntent Property

Sets the data access intent of the page.

Applies to

Page

Report

Query

Property Value

*

ReadOnly

Intent to access records, but not to modify them. Read-only pages are run against a replica of the database leading to improved performance, but preventing modifications to the database records.

*

ReadWrite

Intent to access and modify records.

Syntax

`DataAccessIntent = ReadOnly|ReadWrite;`

Remarks

For reports, API pages, and API queries, the Business Central server can use read-only database replicas on Azure SQL Database and SQL Server. If replicas are enabled, use this property to reduce the load on the primary database. Using

ReadOnly might also improve performance when viewing objects. ReadOnly works as a hint for the server to route the connection to a secondary (read-only) replica, if one is available. When a workload is executed against the replica, insert/

delete/modify operations aren't possible. If any of these operations are executed against the replica, an exception is thrown at runtime.

Box 2: `Editable = false;`

Note: It only applies to pages of the type API. For such, The Editable property must be set to false.



Reference: <https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/properties/devenv-dataaccessintent-property>

QUESTION 2

You are developing an app that will be published to Microsoft AppSource.

The app requires code analyzers to enforce some rules. You plan to add the analyzers to the settings.json file.

You need to activate the analyzers for the project.

Which three code analyzers should you activate to develop the app for AppSource? Each correct answer presents part of the solution

NOTE: Each correct selection is worth one point.

- A. CodeCop
- B. UICop
- C. a custom rule set
- D. PerTenantExtensionCop
- E. AppSourceCop

Correct Answer: ADE

When developing an app for Microsoft AppSource, it is crucial to adhere to specific guidelines and standards to ensure compatibility and compliance. The three code analyzers you should activate are:

CodeCop (A): This is the default analyzer for AL language extensions. It enforces the AL Coding Guidelines, ensuring that the code follows best practices for readability, maintainability, and performance. It checks for a wide range of issues,

from syntax errors to best practice violations, making it essential for any AL development.

PerTenantExtensionCop (D): This analyzer is specifically designed for extensions that are intended to be installed for individual tenants. It includes rules that ensure the extension does not interfere with the per-tenant customizations and

adheres to the guidelines for extensions that can be safely installed and uninstalled without affecting the underlying application. AppSourceCop (E): This analyzer is tailored for extensions that are intended for publication on Microsoft

AppSource. It enforces additional rules that are specific to AppSource submissions, such as checking for the use of reserved object ranges and ensuring that all prerequisite dependencies are correctly declared. This is crucial for ensuring

that your app meets all the requirements for listing on AppSource. By activating these three analyzers, developers can ensure their app adheres to the standards required for AppSource, as well as maintain high code quality and compatibility

with Business Central.



QUESTION 3

HOTSPOT

You need to define the properties of the comments field of the Non-conformity page.

How should you complete the code segment? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

ExtendedDataType property

```

group(commentsGroup)
{
    field("comments"; NonConformityComments)
    {
        ApplicationArea = All;
        MultiLine = True;
        MultiLine = False;
        NotBlank= True;
        NotBlank= False;
        DataType
        ExtendedDataType
        ExtendedDatatype
        RichDataType
    }
}

```

MultiLine = True;
 MultiLine = False;
 NotBlank= True;
 NotBlank= False;

LongContent
 None
 RichContent
 TextRichContent

```

var
    NonConformityComments: Text;

```

Correct Answer:



ExtendedDataType property

```
group(commentsGroup)
{
    field("comments"; NonConformityComments)
    {
        ApplicationArea = All;
        MultiLine = True;
        MultiLine = False;
        NotBlank = True;
        NotBlank = False;
        ExtendedDatatype = RichContent;
    }
}
```

```
var
    NonConformityComments: Text;
```

Scenario: When a purchase order with incorrect quantity or quality issues is received, the entity must create a non-conformity document in the system. The following information must be included in the document:

- * Comments: can include comments with rich text and pictures to illustrate quality problems
- * etc.

Box 1: MultiLine = true;

Rich Text and content controls

The Rich Text feature in Business Central is designed to handle multimedia content, such as social media posts, email bodies, quick annotations, file contents, or any field that requires a mix of text, tables, links, and images.

Both ExtendedDataType and Multiline are required to render a Rich Text Editor ExtendedDataType = RichContent; MultiLine = true;



Box 2: ExtendedDataType

Box 3: RichContent

Reference: <https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/developer/devenv-richtext-content-controls>

QUESTION 4

HOTSPOT

You plan to create a table to hold client data.

You have the following data integrity requirements:

1.

Lookups into other records must be established.

2.

Validate if a record exists in a destination record.

You need to select the table field property to use for each requirement.

Which table field property should you use? To answer, select the appropriate options in the answer area.

NOTE: Each correct selection is worth one point.

Hot Area:

Build tables

Requirement

Establish lookups into other records.

Validate if a record exists.

Table field property

	▼
DataClassification	
ExternalAccess	
TableRelation	
ValidateTableRelation	
	▼
CalcFormula	
Access	
AccessByPermission	
ValidateTableRelation	

Correct Answer:



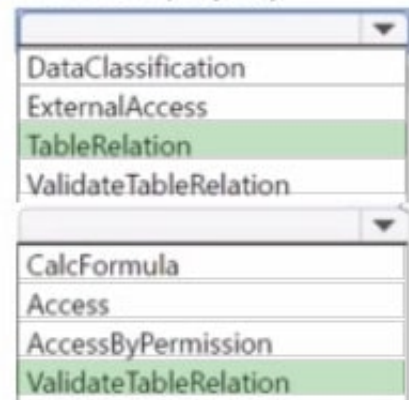
Build tables

Requirement

Establish lookups into other records.

Validate if a record exists.

Table field property



For the data integrity requirements, the table field properties to use are:

To establish lookups into other records, use the TableRelation property. To validate if a record exists in a destination record, use the ValidateTableRelation property.

In Business Central, when creating tables to hold data, maintaining data integrity is crucial:

TableRelation Property: This property is used to create a relationship between the field in one table and a field in another table, which is typically used for lookups. When you set the TableRelation property on a field, it allows users to select from a list of values that exist in the related table.

ValidateTableRelation Property: This property is used to ensure that the value entered in a field matches one of the values in a related table. If a user tries to enter a value that doesn't exist in the related table, an error will occur.

QUESTION 5

You need to determine if you have unwanted incoming web service calls in your tenant during the last seven days.

Which two KQL queries should you use? Each correct answer presents a complete solution.

NOTE: Each correct selection is worth one point.

A. `traces | where timestamp > ago(7d) | where customDimensions has \"RT0008\" | where customDimensions.category in (\"ODataV4\", \"ODataV3\", \"Api\")`

B. `traces | where timestamp > ago(7d) | where customDimensions has \"RT0008\" | where customDimensions.category == \"SOAP\"`

C. `traces | where timestamp > ago(7d) | where customDimensions == \"RT0008\" | where customDimensions.category == \"SOAP\"`

D. `traces | where timestamp > ago(7d) | where customDimensions has \"RT0008\" | where customDimensions.category != \"ODataV4\"`

E. `traces | where timestamp > ago(7d) | where customDimensions has \"RT0008\" | where customDimensions.category in (\"ODataV4\", \"Api\")`



Correct Answer: BC

Analyzing Incoming Web Services Request Telemetry

Web services telemetry gathers data about SOAP, OData, and REST API requests to the service. It provides information like the request's endpoint, time to complete, HTTP status codes, and more.

What type of web services are called (REST API, OData, or SOAP)?

The custom dimension category hold information about the type of endpoint (REST API, OData, or SOAP) being called.

This KQL code illustrates how you can find which endpoints are being called and which types of endpoints are being used. Such information might be useful, should you want to modernize your use of web services to use REST APIs, which is

recommended over using SOAP web services or OData web services based on pages.

```
// Incoming Web Service Requests - endpoint information
traces
| where timestamp > ago(1d) // change as needed
| where customDimensions has "RT0008"
| where customDimensions.eventId == "RT0008"
| project timestamp
// endpoint information
, category = customDimensions.category // API, ODataV3, ODataV4, or SOAP
, endpoint = customDimensions.endpoint // URI
```

Scenario:

Contoso, Ltd. interacts with external services provided by customers and partners. Different applications interact with SOAP and OData endpoints exposed from Business Central.

SOAP

Contoso, Ltd. plans to dismiss using the SOAP protocol for integrations.

Contoso, Ltd. must be able to detect if external applications are using its Business Central SOAP endpoints.

Reference:

<https://learn.microsoft.com/en-us/dynamics365/business-central/dev-itpro/administration/telemetry-webservices-trace>

[MB-820 PDF Dumps](#)

[MB-820 VCE Dumps](#)

[MB-820 Brindumps](#)