



PL-400^{Q&As}

Microsoft Power Platform Developer

Pass Microsoft PL-400 Exam with 100% Guarantee

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

<https://www.pass4itsure.com/pl-400.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****DRAG DROP**

You are developing an inventory tracking component for a warehouse. You plan to use the component with a Power Apps app.

Truck drivers will use tablet devices to confirm their cargo manifest. Warehouse employees will use their phones to pick and fulfill orders. The phone-based version must run in a Power Apps runtime container.

You need to create a single Power Apps component that can be used on both tablet devices and phones.

How should you complete the code segment? To answer, drag the appropriate code segments to the correct locations. Each code segment may be used once, more than once, or not at all. You may need to drag the split bar between panes

or scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

Code Segments	Answer Area
<div>getValue</div>	<pre>var gTablet1 = 2; var gPhone1 = 3; var gTablet2 = "Web"; var gPhone2 = "Mobile"; if (context.client. <input type="text"/> () == "Mobile") { if (context.client. <input type="text"/> () == gPhone1) { . . . } else if . . . { . . . } }</pre>
<div>getTitle</div>	
<div>getFormFactor</div>	
<div>getClient</div>	

Correct Answer:

**Code Segments**

getValue

getTitle

Answer Area

```
var gTablet1 = 2;
var gPhone1 = 3;
var gTablet2 = "Web";
var gPhone2 = "Mobile";
if (context.client.getClient() == "Mobile") {

    if (context.client.getFormFactor() == gPhone1) {

        . . .
    }
    else if . . . {

        . . .
    }
}
```

Box 1: getClient

Code example:

```
var isCrmForMobile = (Xrm.Page.context.client.getClient() == "Mobile")
```

```
if (isCrmForMobile)
```

```
{
```

```
// Code for CRM for phones and tablets only goes here. } else {
```

```
// Code for web browser or CRM for Outlook only goes here. } if (Xrm.Page.context.client.getClient() == "Mobile"
andand Xrm.Page.context.client.getFormFactor() == 3) {
```

```
// Add code that should only run in CRM for phones here } // Code for any client goes here.
```

Box 2: getFormFactor

Reference: <https://stackoverflow.com/questions/47071328/navigating-in-the-dynamics-365-crm-app>**QUESTION 2****DRAG DROP**

Five high schools test a custom app from AppSource. They provide feedback that the Course credit entity should include additional fields that cover information shared by the schools.

You do not have access to each high school organization.

Each high school administrator must be able to apply the updates to the Course credit entity.



You need to deliver a custom program that creates the additional fields.

Which four actions should you perform in sequence? To answer, move the appropriate actions from the list of actions to the answer area and arrange them in the correct order.

Select and Place:

Actions

Answer Area

Retrieve the Course credit entity metadata by using RetrieveEntityRequest with LogicalName.

Retrieve the Course credit entity metadata by using RetrieveEntityRequest with MetadataId.

Define the AttributeMetadata for each new field.

Call the CreateAttributeRequest constructor for each new field.

Call the RetrieveAttributeRequest with LogicalName for each new field.

Call the login logic.

Retrieve the Course credit entity metadata by using RetrieveEntityRequest with EntityFilters = Attributes and LogicalName.



Correct Answer:



Actions	Answer Area
Retrieve the Course credit entity metadata by using RetrieveEntityRequest with LogicalName.	Call the login logic.
Retrieve the Course credit entity metadata by using RetrieveEntityRequest with MetadataId.	Retrieve the Course credit entity metadata by using RetrieveEntityRequest with EntityFilters = Attributes and LogicalName.
	Define the AttributeMetadata for each new field.
Call the CreateAttributeRequest constructor for each new field.	Call the RetrieveAttributeRequest with LogicalName for each new field.

Step 1: Call the login logic.

Step 2: Retrieve the Course credit entity metadata by using RetrieveEntityRequest with EntityFilters = Attributes and LogicalName

The RetrieveEntityRequest.EntityFilters property gets or sets a filter to control how much data for the entity is retrieved.

Step 3: Define the AttributeMetaData for each new field.

Step 4: Call the RetrieveAttributeRequest with LogicalName for each new field. The RetrieveAttributeRequest contains the data that is needed to retrieve attribute metadata.

QUESTION 3

Note: This question is part of a series of questions that present the same scenario. Each question in the series contains a unique solution that might meet the stated goals. Some question sets might have more than one correct solution, while

others might not have a correct solution.

After you answer a question in this section, you will NOT be able to return to it. As a result, these questions will not appear in the review screen.

An organization has an Azure SQL Database instance that must be synchronized daily with data from Dynamics 365 Sales. A large amount of data might need to be synchronized on some days.

You need to reduce the time required to synchronize data.

Solution:



Write a SSIS package to connect to the source and target. Develop the SSIS package to find the records by the Modified on field. Create or update the records in the database instance based on results.

Does the solution meet the goal?

A. Yes

B. No

Correct Answer: B

Instead enable change tracking for entities that will be synchronized, and use the Data Export Service to sync data between the database and Dynamics 365 Sales.

Reference: <https://docs.microsoft.com/en-us/power-platform/admin/enable-change-tracking-control-data-synchronization>
<https://docs.microsoft.com/en-us/power-platform/admin/replicate-data-microsoft-azure-sql-database>

QUESTION 4

DRAG DROP

A company uses Microsoft 365. You are developing a model-driven app.

The app must meet the following requirements:

1.

Use SharePoint Online for document storage.

2.

Send emails by using Exchange Online.

You need to configure integrations.

What should you configure? To answer, drag the appropriate configuration options to the correct requirements. Each configuration option may be used once, more than once, or not at all. You may need to drag the split bar between panes or

scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:



Configuration options

Server-side synchronization

Server-based integration

Dual-write

System settings

Requirement

Email

Document storage

Configuration option

Configuration option

Configuration option

Correct Answer:



Configuration options

Dual-write
System settings

Requirement

Email

Document storage

Configuration option

Server-side synchronization

Server-based integration

Box 1: Server-side synchronization

Configure default email processing and synchronization: set server-side synchronization to be the default configuration method for newly created users.

Box 2: Server-side integration.

If your organization is already using document management with Microsoft Dynamics CRM List Component, you must switch to server-based SharePoint integration. If your organization has not deployed document management, when a

System Administrator logs in an alert message will be displayed to enable server-based SharePoint integration.

QUESTION 5

DRAG DROP

You are creating a Web API.

The API must be able to perform the following actions:

1.



Create a column in a Microsoft Dataverse table.

2.

Update a column for an existing row.

Which HTTP methods should you use? To answer, drag the appropriate HTTP methods to the correct requirements. Each HTTP method may be used once, more than once, or not at all. You may need to drag the split bar between panes or

scroll to view content.

NOTE: Each correct selection is worth one point.

Select and Place:

HTTP methods

GET

POST

PATCH

ACCEPT

Answer Area

Requirement

Create a column.

Update a column for an existing row.

HTTP method

HTTP method

HTTP method

Correct Answer:

HTTP methods

GET

ACCEPT

Answer Area

Requirement

Create a column.

Update a column for an existing row.

HTTP method

POST

PATCH

Box 1: POST Create table definitions To create a table definition, POST the JSON representation of the entity definition data to the EntityDefinitions entity set path. The entity must include the definition for the primary name attribute. You don't need to set values for all the properties. The items on this list except for Description are required, although setting a description is a recommended best practice.



Box 2: PATCH Basic update Update operations use the HTTP PATCH verb. Pass a JSON object containing the properties you want to update to the URI that represents the record.

Reference: <https://docs.microsoft.com/en-us/power-apps/developer/data-platform/webapi/create-update-entity-definitions-using-web-api> <https://docs.microsoft.com/en-us/power-apps/developer/data-platform/webapi/update-delete-entities-using-web-api>

[Latest PL-400 Dumps](#)

[PL-400 VCE Dumps](#)

[PL-400 Practice Test](#)