Study guide for Exam PL-400: Microsoft Power Platform Developer

Purpose of this document

This study guide should help you understand what to expect on the exam and includes a summary of the topics the exam might cover and links to additional resources. The information and materials in this document should help you focus your studies as you prepare for the exam.

Useful links	Description
Review the skills measured as of September 19, 2022	This list represents the skills measured AFTER the date provided. Study this list if you plan to take the exam AFTER that date.
Review the skills measured prior to September 19, 2022	Study this list of skills if you take your exam PRIOR to the date provided.
Change log	You can go directly to the change log if you want to see the changes that will be made on the date provided.
How to earn the certification	Some certifications only require passing one exam, while others require passing multiple exams.
Certification renewal	Microsoft associate, expert, and specialty certifications expire annually. You can renew by passing a **free** online assessment on Microsoft Learn.
Your Microsoft Learn profile	Connecting your certification profile to Learn allows you to schedule and renew exams and share and print certificates.
Passing score	A score of 700 or greater is required to pass.
Exam sandbox	You can explore the exam environment by visiting our exam sandbox.
Request accommodations	If you use assistive devices, require extra time, or need modification to any part of the exam experience, you can request an accommodation.

Useful links	Description
Take a practice test	Are you ready to take the exam or do you need to study a bit more?

Updates to the exam

Our exams are updated periodically to reflect skills that are required to perform a role. We have included two versions of the Skills Measured objectives depending on when you are taking the exam.

We always update the English language version of the exam first. Some exams are localized into other languages, and those are updated approximately eight weeks after the English version is updated. Other available languages are listed in the **Schedule Exam** section of the **Exam Details** webpage. If the exam isn't available in your preferred language, you can request an additional 30 minutes to complete the exam.

Note

The bullets that follow each of the skills measured are intended to illustrate how we are assessing that skill. Related topics may be covered in the exam.

Note

Most questions cover features that are general availability (GA). The exam may contain questions on Preview features if those features are commonly used.

Skills measured as of September 19, 2022

Audience profile

Candidates for this exam design, develop, test, secure, and troubleshoot Microsoft Power Platform solutions. Candidates implement components of a solution that include application enhancements, custom user experiences, system integrations, data conversions, and custom process automation.

Candidates must have strong applied knowledge of Microsoft Power Platform services, including in-depth understanding of capabilities, boundaries, and constraints. Candidates should have a basic understanding of application lifecycle management (ALM) practices for Microsoft Power Platform.

Candidates should have development experience that includes Microsoft Power Platform services, JavaScript, JSON, TypeScript, C#, HTML, .NET/ASP.NET, RESTful Web APIs, Microsoft Azure, and Microsoft 365.

- Create a technical design (10–15%)
- Configure Microsoft Dataverse (10–15%)
- Create and configure PowerApps (5–10%)
- Configure business process automation (5–10%)
- Extend the user experience (15–20%)

- Extend the platform (20–25%)
- Develop integrations (5–10%)

Create a technical design (10–15%)

Validate requirements and design technical architecture

- Design and validate the technical architecture for a solution
- Design authentication and authorization strategy
- Determine whether you can meet requirements with out-of-the-box functionality
- Determine when to use Logic Apps versus Power Automate flows
- Determine when to use serverless computing, plug-ins, or Power Automate
- Determine when to build a virtual table data source provider and when to use connectors

Design solution components

- Design a Microsoft Dataverse data model
- Design Power Apps reusable components
- Design custom connectors
- Design server-side components
- Determine when to extend business process flows by using server-side and client-side code or Power Automate

Describe Microsoft Power Platform extensibility points

- Describe Power Virtual Agents extensibility points including Bot Framework skills and Power Automate flows
- Describe Power Apps portal extensibility points including CRUD APIs and custom styling
- Describe Dataverse custom APIs and their uses

Configure Microsoft Dataverse (10–15%)

Configure security to support development

- Troubleshoot operational security issues
- Create or update security roles and column-level security profiles
- Configure business units and teams

Implement tables and columns

- Configure tables and table options
- Configure columns
- Configure relationships and types of behaviors

Implement application lifecycle management (ALM)

- Create solutions and manage solution components
- Import and export solutions

- Manage solution dependencies
- Implement source control for projects including solutions and code assets
- Create and use environment variables
- Describe how to use Package Deployer and associated tools to create a package
- Describe application lifecycle management concepts

Create and configure Power Apps (5–10%)

Create model-driven apps

- Configure a model-driven app
- Configure forms
- Configure views
- Configure commands and buttons

Create canvas apps

- Create and configure a canvas app or a custom page
- Implement complex formulas to manage control events and properties
- Build reusable component libraries
- Test an app by using Test Studio
- Embed an app in Microsoft Teams

Manage and troubleshoot apps

- Troubleshoot app issues by using Monitor and other browser-based debugging tools
- Identify and resolve connector and API errors
- Optimize app performance including pre-loading data and query delegation

Configure business process automation (5–10%)

Configure Power Automate

- Build a cloud flow
- Configure steps to use Dataverse connector actions and triggers
- Implement complex expressions in flow steps
- Implement error handling

Implement processes

- Create and configure business process flows
- Create and configure business rules
- Create, manage, and interact with business process flows by using server-side and client-side code
- Troubleshoot processes

Extend the user experience (15–20%)

Apply business logic using client scripting

- Create JavaScript or Typescript code that targets the Client API object model
- Register an event handler
- Create client-side scripts that target the Dataverse Web API

Create a Power Apps Component Framework (PCF) component

- Describe the code component lifecycle
- Initialize a new code component
- Configure a code component manifest
- Implement component interfaces
- Package, deploy, and consume a component
- Configure and use Device, Utility, and WebAPI features

Create a command button function

- Create a command function
- Design command button rules and actions
- Manage dependencies between JavaScript libraries

Extend the platform (20–25%)

Create a plug-in

- Describe the plug-in execution pipeline
- Design and develop a plug-in
- Debug and troubleshoot a plug-in
- Implement business logic by using pre-images and post-images
- Perform operations on data by using the Organization service API
- Optimize plug-in performance by configuring concurrency, and transactions
- Configure a Dataverse custom API message
- Register custom assemblies by using the Plug-in Registration Tool
- Develop a plug-in that targets a custom action message

Create custom connectors

- Create a definition for the API
- Configure API security
- Use policy templates to modify connector behavior at runtime
- Create custom connectors for public APIs by using Postman

Use platform APIs

- Interact with data and processes by using the Dataverse Web API or the Organization Service

- Implement API limit retry policies
- Optimize for performance, concurrency, transactions, and batching
- Perform authentication by using OAuth

Process workloads

- Process long-running operations by using Azure Functions
- Configure scheduled and event-driven function triggers in Azure Functions
- Authenticate to Microsoft Power Platform by using managed identities

Develop Integrations (5–10%)

Publish and consume Dataverse events

- Publish an event by using the API
- Publish an event by using the Plug-in Registration Tool
- Register service endpoints including webhooks, Azure Service Bus, and Azure Event Hub
- Implement a Dataverse listener for an Azure solution
- Create an Azure Function that interacts with Microsoft Power Platform

Implement data synchronization

- Configure table change tracking
- Read table change records by using platform APIs
- Create and use alternate keys

Study resources

We recommend that you train and get hands-on experience before you take the exam. We offer self-study options and classroom training as well as links to documentation, community sites, and videos.

Study resources	Links to learning and documentation
Get trained	Choose from self-paced learning paths and modules or take an instructor led course
Find documentation	Developer learning catalog - Power Apps Planning a Power Apps project Canvas apps Model-driven apps Portals Microsoft Dataverse Tables and metadata Dataverse developer Model-driven apps developer

Study resources	Links to learning and documentation
	Canvas apps developer Implement healthy ALM using solutions Use Microsoft Power Platform Build Tools
Ask a question	Microsoft Q&A | Microsoft Docs
Get community support	Power Apps - Power Platform Community Microsoft Power Automate - Power Platform Community
Follow Microsoft Learn	Microsoft Learn - Microsoft Tech Community
Find a video	#LessCodeMorePower | Shows Browse other Microsoft Learn shows

Change log

Key to understanding the table: The topic groups (also known as functional groups) are in bold typeface followed by the objectives within each group. The table is a comparison between the two versions of the exam skills measured and the third column describes the extent of the changes.

Skill area prior to September 19, 2022	Skill area as of September 19, 2022	Changes
Audience profile		No change
Create a technical design	**Create a technical design**	No change
Validate requirements and design technical architecture	Validate requirements and design technical architecture	No change
Design solution components	Design solution components	No change
Describe Microsoft Power Platform extensibility points	Describe Microsoft Power Platform extensibility points	No change
Configure Microsoft Dataverse	**Configure Microsoft Dataverse**	No change
Configure security to support development	Configure security to support development	No change
Implement tables and columns	Implement tables and columns	No change

Skill area prior to September 19, 2022	Skill area as of September 19, 2022	Changes
Implement application lifecycle management (ALM)	Implement application lifecycle management (ALM)	Minor
Create and configure Power Apps	**Create and configure Power Apps**	No change
Create model-driven apps	Create model-driven apps	No change
Create canvas apps	Create canvas apps	Minor
Manage and troubleshoot apps	Manage and troubleshoot apps	No change
Configure business process automation	**Configure business process automation**	No change
Configure Power Automate	Configure Power Automate	No change
Implement processes	Implement processes	No change
Extend the user experience	**Extend the user experience**	No change
Apply business logic using client scripting	Apply business logic using client scripting	No change
Create a Power Apps Component Framework (PCF) component	Create a Power Apps Component Framework (PCF) component	No change
Create a command button function	Create a command button function	No change
Extend the platform	**Extend the platform**	No change
Create a plug-in	Create a plug-in	Minor
Create custom connectors	Create custom connectors	No change
Use platform APIs	Use platform APIs	No change
Process workloads	Process workloads	No change
Develop Integrations	**Develop Integrations**	No change
Publish and consume Dataverse events	Publish and consume Dataverse events	No change
Implement data synchronization	Implement data synchronization	No change

Skills measured prior September 19, 2022

Audience profile

Candidates for this exam design, develop, test, secure, and troubleshoot Microsoft Power Platform solutions. Candidates implement components of a solution that include application enhancements, custom user experiences, system integrations, data conversions, and custom process automation.

Candidates must have strong applied knowledge of Microsoft Power Platform services, including in-depth understanding of capabilities, boundaries, and constraints. Candidates should have a basic understanding of application lifecycle management (ALM) practices for Microsoft Power Platform.

Candidates should have development experience that includes Microsoft Power Platform services, JavaScript, JSON, TypeScript, C#, HTML, .NET/ASP.NET, RESTful Web APIs, Microsoft Azure, and Microsoft 365.

- Create a technical design (10–15%)
- Configure Microsoft Dataverse (10–15%)
- Create and configure PowerApps (5–10%)
- Configure business process automation (5–10%)
- Extend the user experience (15–20%)
- Extend the platform (20–25%)
- Develop integrations (5–10%)

Create a technical design (10–15%)

Validate requirements and design technical architecture

- Design and validate the technical architecture for a solution
- Design authentication and authorization strategy
- Determine whether you can meet requirements with out-of-the-box functionality
- Determine when to use Logic Apps versus Power Automate flows
- Determine when to use serverless computing, plug-ins, or Power Automate
- Determine when to build a virtual table data source provider and when to use connectors

Design solution components

- Design a Microsoft Dataverse data model
- Design Power Apps reusable components
- Design custom connectors
- Design server-side components
- Determine when to extend business process flows by using server-side and client-side code or Power Automate

Describe Microsoft Power Platform extensibility points

- Describe Power Virtual Agents extensibility points including Bot Framework skills and Power Automate flows
- Describe Power Apps portal extensibility points including CRUD APIs and custom styling
- Describe Dataverse custom APIs and their uses

Configure Microsoft Dataverse (10–15%)

Configure security to support development

- Troubleshoot operational security issues
- Create or update security roles and column-level security profiles
- Configure business units and teams

Implement tables and columns

- Configure tables and table options
- Configure columns
- Configure relationships and types of behaviors

Implement application lifecycle management (ALM)

- Create solutions and manage solution components
- Import and export solutions
- Manage solution dependencies
- Implement source control for projects including solutions and code assets
- Create and use environment variables
- Describe how to use Package Deployer and associated tools to create a package
- Automate deployments
- Implement GitHub actions

Create and configure Power Apps (5–10%)

Create model-driven apps

- Configure a model-driven app
- Configure forms
- Configure views
- Configure commands and buttons

Create canvas apps

- Create and configure a canvas app or custom page
- Implement complex formulas to manage control events and properties
- Build reusable component libraries
- Test an app by using Test Studio

- Embed an app in Microsoft Teams

Manage and troubleshoot apps

- Troubleshoot app issues by using Monitor and other browser-based debugging tools
- Identify and resolve connector and API errors
- Optimize app performance including pre-loading data and query delegation

Configure business process automation (5–10%)

Configure Power Automate

- Build a cloud flow
- Configure steps to use Dataverse connector actions and triggers
- Implement complex expressions in flow steps
- Implement error handling

Implement processes

- Create and configure business process flows
- Create and configure business rules
- Create, manage, and interact with business process flows by using server-side and client-side code
- Troubleshoot processes

Extend the user experience (15–20%)

Apply business logic using client scripting

- Create JavaScript or Typescript code that targets the Client API object model
- Register an event handler
- Create client-side scripts that target the Dataverse Web API

Create a Power Apps Component Framework (PCF) component

- Describe the code component lifecycle
- Initialize a new code component
- Configure a code component manifest
- Implement component interfaces
- Package, deploy, and consume a component
- Configure and use Device, Utility, and WebAPI features

Create a command button function

- Create a command function
- Design command button rules and actions
- Manage dependencies between JavaScript libraries

Extend the platform (20–25%)

Create a plug-in

- Describe the plug-in execution pipeline
- Design and develop a plug-in
- Debug and troubleshoot a plug-in
- Implement business logic by using pre-images and post-images
- Perform operations on data by using the Organization service API
- Optimize plug-in performance by configuring concurrency, transactions, and batching
- Configure a Dataverse custom API message
- Register custom assemblies by using the Plug-in Registration Tool
- Develop a plug-in that targets a custom action message

Create custom connectors

- Create a definition for the API
- Configure API security
- Use policy templates to modify connector behavior at runtime
- Create custom connectors for public APIs by using Postman

Use platform APIs

- Interact with data and processes by using the Dataverse Web API or the Organization Service
- Implement API limit retry policies
- Optimize for performance, concurrency, transactions, and batching
- Perform authentication by using OAuth

Process workloads

- Process long-running operations by using Azure Functions
- Configure scheduled and event-driven function triggers in Azure Functions
- Authenticate to Microsoft Power Platform by using managed identities

Develop Integrations (5–10%)

Publish and consume Dataverse events

- Publish an event by using the API
- Publish an event by using the Plug-in Registration Tool
- Register service endpoints including webhooks, Azure Service Bus, and Azure Event Hub
- Implement a Dataverse listener for an Azure solution
- Create an Azure Function that interacts with Microsoft Power Platform

Implement data synchronization

- Configure table change tracking

- Read table change records by using platform APIs
- Create and use alternate keys