Check your knowledge

1. Your social media engagement officer has requested your help in boosting followers and retweets on Twitter. How could you help her get more information to better understand and subsequently increase engagement?

- (●) Configure a Power BI report to capture and analyze data from Twitter, allowing you to better understand why certain posts elicit more responses. ✓

  Correct. Power BI allows you to create visuals and better understand your data. Once you understand trends in what followers like, you can post more of that content and increase engagement.

- ( ) Power Apps using portals can create a new customer site for our followers.
- ( ) Power Automate can handle our content approvals for us, reducing the time it takes to produce new content and ensure our quality procedure is followed.

3. Someone has added an item in SharePoint which prompts a workflow to run in Power Automate. What type of operation have you used to start your workflow?

- (●) Trigger ✓

  Correct.A trigger is an operation that tells a workflow to begin or prompts some type of action.

- ( ) Action
- ( ) Function-based

4. A client likes the idea of implementing a Power Platform solution, but is concerned about the ability to interact with a custom API. How should you respond?

- (●) The Power Platform offers the ability to create custom connectors for this purpose which allow you to connect to Power Apps and Power Automate. ✓

  Correct. You can build out a custom connector to bridge your app or workflow to the API.

- ( ) The Power Platform has over 270 connectors to use in these situations.
- ( ) The Power Platform uses connectors that hold a series of functions available for developers.

5. Someone asks you to describe a connector. How would you respond?

- (●) Connectors connect your data source to your app, workflow, or dashboard. ✓

  Correct. Connectors allow functions and information to pass from your data source to your app or workflow.

- ( ) Connectors hold a series of functions available for developers.
- ( ) Connectors are a cloud-based service that makes it practical and simple for line-of-business users to build workflows that automate time-consuming business tasks and processes across applications and services.

Key takeaways

Here are the five key takeaways:

1. Power Platform is a system that enables users to do three key actions on data that help them drive business: gain insights from data (Analyze), drive intelligent business processes via apps they build (Act), and automate business processes (Automate).

2. Power BI helps you analyze and visualize data on a unified platform with data from internal and external sources.

3. Power Apps helps you build and deploy customized apps that work across web and mobile, embedded or standalone, on any device.

4. Connectors are bridges that allow you to send information from your data source to your app or workflow and back.

5. Power Automate helps you create automation workflows, from simple to advanced scenarios.

Resources

Use these resources to discover more.

Tip

To open a resource link, right-click and select "Open in a new tab or window". That way, you can check out the resource and easily return to the module tab to unlock your achievement when done.

Power BI

- Power BI
- Power BI customer showcase

Power Apps

- Power Apps
- Power Apps Resources

Power Automate

- Power Automate
- Power Automate Documentation

More on Connectors

- Connector Reference
- Overview of canvas-app connectors for Power Apps

Getting started with Custom Connectors

- Using a blank custom connector
- From an OpenAPI definition
- From a Postman collection
- Use custom connectors in a PowerApps canvas app

Get started with Power Virtual Agents

- Power Virtual Age

https://docs.microsoft.com/en-us/learn/modules/introduction-common-data-service/1-introduction

Introduction

- 3 minutes

Common Data Service is a cloud-based, low-code data service and app platform, which allows you to leverage the security and connectivity of Microsoft services. Common Data Service connects easily to all aspects of the Power Platform so that you can fully control, automate, and strengthen your business. With standard entities and fields, as well as the ability to easily define relationships between your data, Common Data Service was built for powerful, scalable solutions.

Overview of Common Data Service

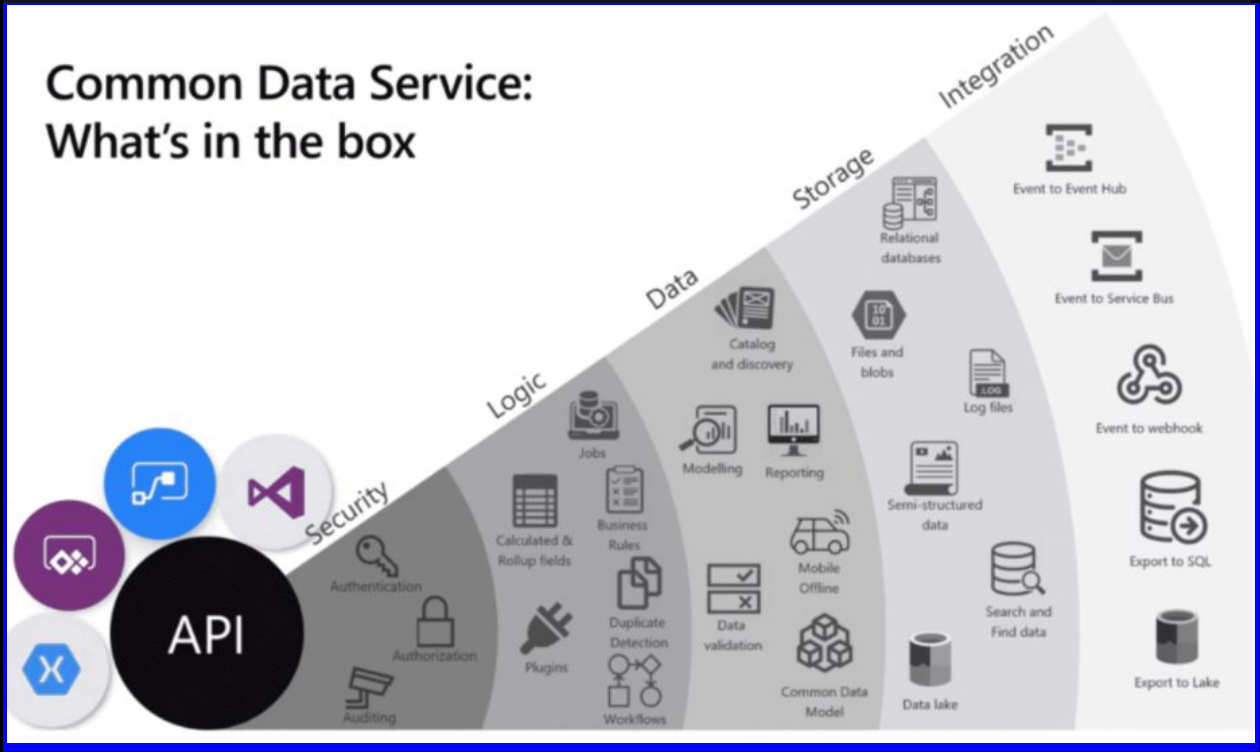
The Common Data Service is a cloud-based solution that easily structures a variety of data and business logic to support interconnected applications and processes in a secure and compliant manner. ‘

Managed and maintained by Microsoft, Common Data Service is available globally but deployed geographically to comply with your potential data residency.

It is not designed for stand-alone use on your servers, so you will need an internet connection to access and use it.

The Common Data Service is designed to be your central data repository for business data, and you might even be using it already.

Behind the scenes, it powers many Microsoft Dynamics 365 solutions such as Field Service, Marketing, Customer Service, and Sales. It is also available as part of Power Apps and Power Automate with native connectivity built right in. The AI Builder and Portals features of the Power Platform also utilize the Common Data Service.



Security: Common Data Service handles authentication with Azure Active Directory (Azure AD) to allow for conditional access and multi-factor authentication. It supports authorization down to the row and field level and provides rich auditing capabilities.

Logic: Common Data Service allows you to easily apply business logic at the data level. Regardless of how a user is interacting with the data, the same rules apply. These rules could be related to duplicate detection, business rules, workflows, or more.

Data: Common Data Service offers you the control to shape your data, allowing you to discover, model, validate, and report on your data. This control ensures your data looks the way you want regardless of how it is used.

Storage: Common Data Service stores your physical data in the Azure cloud. This cloud-based storage removes the burden of worrying about where your data lives or how it scales. These concerns are all handled for you.

Integration: Common Data Service connects in different ways to support your business needs. APIs, webhooks, eventing, and data exports give you flexibility to get data in and out.

Common Data Service defined

A Common Data Service database is a single instance of Common Data Service which stores data in a set of standard and custom data structures called entities. An entity is a logical set of records that is used to store data. Entities are like tables in a relational database, but there are subtle differences. Records within an entity contain many fields to manage individual pieces of information about a single record.

Scalability

A Common Data Service database supports large data sets and complex data models. Entities can hold millions of items, and you can extend the storage in each instance of a Common Data Service database to four (4) terabytes per instance. The amount of data that is available in your instance of Common Data Service is based upon the number and type of licenses that are associated with it. Data storage is pooled between all licensed users, so you can allocate storage as needed for each solution that you build. Additional storage can be purchased if you need more storage than what is offered within standard licensing.
Tip

Common Data Service supports transactional multi-user applications, where quick response to user demand is the priority. It is not intended to be a platform for long running or batch processing.

Common Data Model vs. Common Data Service

The standard entity design in a Common Data Service database is based upon an open data model standard called Common Data Model. Common Data Model is a logical design that includes a set of open-sourced, standardized, extensible data entities and relationships that Microsoft and its partners have published in an industry-wide initiative called the Open Data Initiative. This collection of predefined entities, attributes, semantic metadata, and relationships form the basis of the Common Data Model.

Open Data Initiative

The Common Data Model is the output of the Open Data Initiative. It provides a platform for a single, comprehensive view of your data, bringing together and enriching data from all your lines of business, across all your systems, to deliver real-time intelligence back into your applications and services. Microsoft has partnered with SAP and Adobe on the Open Data Initiative.

Common Data Service structure and benefits

The structure of a Common Data Service database is based upon the definitions and schema in the Common Data Model. The key benefit of using Common Data Model as the basis of a Common Data Service database is simplified integration of any solutions that use a Common Data Model schema, because the standard entities of the solution are the same. You will also be able to take advantage of a rich ecosystem of solutions that vendors have built from using Common Data Model. Best of all, there is practically no limit to how far you can extend a Common Data Service database.

Identify entities and fields in Common Data Service

- 3 minutes

An entity is similar to a table in a database or in an Excel workbook in the way it stores data. It is a logical structure containing records that are made up of fields or put more simply, rows and columns.

ypes of entities

The two types of entities are:

- Standard - The base set of entities that are created for every instance of a Common Data Service database. You can add more fields to any entity, but you can only delete fields from a custom entity.
- Complex - Entities that contain complex, server-side business logic, including real-time workflows or plug-ins. Some of the entities that are used in Dynamics 365 applications are complex. Care must be taken if you add server-side logic to ensure that users have the proper license to use the complex entity. Additional information about complex entities can be accessed by following the link within the summary unit of this module.

Fields

Fields are a way to store a discrete piece of information within a record in an entity. You might think of them as a column in Excel. Fields have types, meaning that you can store data of a certain type in a field that matches that data type. For example, if you have a solution that requires dates, then you would store the date in a field with the type of Date. Similarly, if you want to store a number, then you store the number in a field with the type of Number.

The number of fields within an entity varies from a few fields to a hundred or more. If you need more than a few hundred fields in an entity, you might want to reconsider how you are structuring data storage for your solution because, likely, there is a better way.

Every database in Common Data Service starts with a standard set of entities and each standard entity has a standard set of fields.
Tip

Always use standard entities and fields when possible. You can rename an entity if that makes the entity more understandable in the context of your solution. Always review the list of standard entities and make sure a standard entity will not meet your needs before you create a new entity.

Understand relationships

- 3 minutes

To make an efficient and scalable solution for most of the solutions that you build, you will need to split up data into different containers (entities). Trying to store everything into a single container would likely be inefficient and difficult to work with and understand.

The following example helps illustrate this concept.

Imagine that you need to create a system to manage sales orders. You will need a product list along with the inventory on hand, cost of the item, and the selling price. You also need a master list of customers with their addresses and credit ratings. Finally, you will need to manage invoices of sales that you make so you will want a way to store invoice data. The invoice should include information such as date, invoice number, and salesperson, customer information including address and credit rating, and a line item for each item on the invoice. Line items should include a reference to the product that you sold and be able to provide the proper cost and price for each product and decrease the quantity on hand based upon the quantity that you sold in that line item.

Trying to create a single entity to support the functionality that was previously described would be inefficient. A better way to approach this business scenario is to create the following four entities:

- Customers
- Products
- Invoices
- Line Items

Creating an entity for each of these items and relating them to one another will allow you to build an efficient solution that can scale while maintaining high performance. Splitting the data into multiple entities also means that you will not have to store repetitive data or support huge records with large amounts of blank data. Additionally, reporting will be much easier if you split the data into separate entities.

Entities that relate to one another have a relational connection. Relationships between entities exist in many forms, but the two most common are one-to-many and many-to-many, both of which are supported by Common Data Service.

One-to-many relationships are also known as parent-child relationships. In the previous invoice example, the invoice entity would be the parent and the line items would be a child entity. An invoice can have zero, one, or many line items (child records), but the line item will always be related to just one invoice (parent record). Typically, the child records will not exist without a parent record.

A field that only allows unique values, such as invoice number, is used to identify the parent record. This unique field is called a key. The same value (the parent key) is stored in the related child records. This field is called a foreign key when the child record is used to store the parent key value. Ingeniously, filtering is used to display child records with a value in the foreign key that matches the key value in the parent record.

This allows applications to display the child records (line items in the previous example) that belong to a particular parent record (invoice in the previous example). This concept underlies many business software applications.

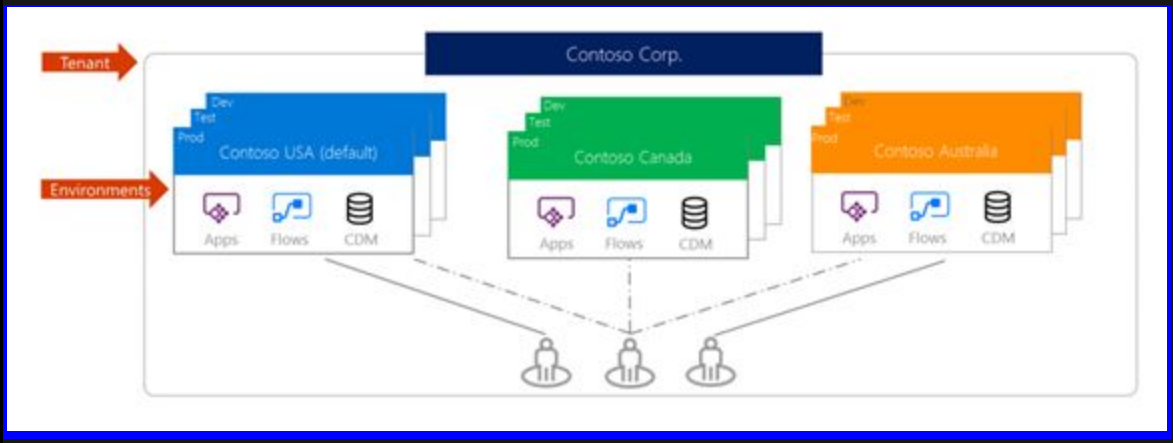
Splitting data into different entities makes for an efficient solution design that can scale, but knowing how to split up the data into entities can be difficult. Thankfully, Common Data Service already contains many of the entities that most organizations will need. Using standard entities and extending them will ensure that you are building solutions around a proven, scalable way of storing the data that is used by your solutions.

Environments in Common Data Service

- 3 minutes

Environments are used to store, manage, and share your organization's business data, apps, and flows in the Power Platform. Each environment allows you to provision one Common Data Service database for use within that environment. Common Data Service environments allow you to manage user access, security settings, and the storage that is associated with that database.

Each environment is created under a Microsoft Azure Active Directory (Azure AD) tenant, and its resources can only be accessed by users within that tenant. An environment is also bound to a geographic location, like the United States. When you create a Common Data Service database in an environment, that database is created within datacenters in that geographic location. Any items that you create in that environment (including connections, gateways, flows that are using Power Automate, and more) are also bound to their environment's location.



You can create more than one environment to manage solution development and data storage by setting up one environment for development, another for testing, and another

for production use. Also, you can set up an environment based on a geographical location. For example, you might set up an environment for Europe and another for Asia. Each of these environments will have zero or only one instance of Common Data Service.

Business rules

- 3 minutes

In Common Data Service you can define business rules. Business rules allow you to apply and maintain business logic at the data layer instead of the app layer. Put more simply, if you create business rules in Common Data Service, they are in effect regardless of how you interact with the data.

An example business rule usage is when they are in canvas or model-driven apps to set or clear values in one or many fields in an entity. They can also be used to validate stored data or show error messages. Model-driven apps can use business rules to show or hide fields, enable or disable fields, and create recommendations based on business intelligence.

Tip

Business rules are usually defined for an entity and apply to all forms, but you can define a business rule for a specific model-driven form. Canvas apps cannot have a business rule applied to a specific form, but they are still enforced when interacting with the data.

Business rules give you a powerful way to enforce rules, set values, or validate data regardless of the form that is used to input data. Additionally, business rules are effective in helping to increase the accuracy of data, simplify application development, and streamline the forms presented to end users.

Business rules can be used by canvas apps or model-driven apps to do the following:

- Set field values
- Clear field values
- Validate data and show error messages

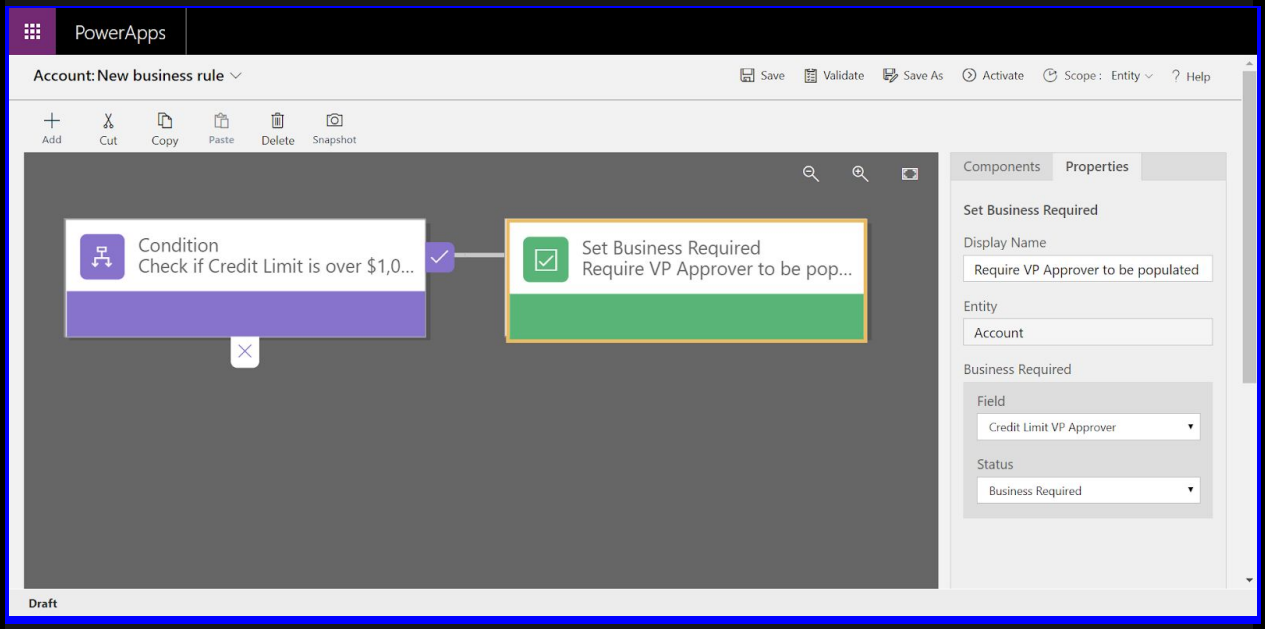
Model-driven apps can also use business rules to:

- Show or hide fields (model-driven apps only)
- Enable or disable fields (model-driven apps only)

- Create business recommendations based on business intelligence (model-driven apps only)

Below is an example of a simple, yet powerful use of business rules. The business rule is configured to change the field Credit Limit VP Approver to be a required field if the Credit limit is set to greater than $1,000,000.

If the credit limit is less than $1,000,000 then the field is optional.



Administer

- 3 minutes

Common Data Service has a rich set of administrative options that you can use to create new instances of a database or tailor access and features that are available for users of each Common Data Service database instance. Several administrative portals are available for you to use to administer Common Data Service settings.

The Power Apps Admin center is discussed in this unit because it will satisfy most of your administrative needs. However, a few other administrative options are available, which are covered in the Manage permissions and administration for Common Data Service learning path.

Power Apps Admin center

Most of the administration settings that you will need are available in the Power Apps Admin center. You should always check for administration settings as your first step when looking to administer Common Data Service.

Settings are grouped into the following broad categories and are accessible by selecting the link on the left-hand side of the portal, as shown in the following figure.

- Environments - This section lists all instances of Common Data Service.
- Data policies - This section lets you set up policies to restrict which data connectors can be used with Common Data Service to limit what data can flow into or out of Common Data Service entities.
- Data integration - This section lets you create or add pre-defined connections and monitor these connections between Common Data Service and other data stores like Salesforce or SQL Server.
- Tenant - This section lets you monitor licenses and quotas.

1. How can business rules be used in a canvas or model-driven app?

- (●) To validate data and show error messages. ✓

  Correct. Common Data Service allows business rules to be set to apply and maintain business logic at the data layer.

- ( ) Show or hide fields.
- ( ) Manage solution development and data storage.

2. Your manager has built a solution using the Common Data Service using a few of the standard Entities included out of the box. He wants to capture the date that new records are added to the Entity, but he isn't sure how. He notices that a "Date" field is already inside his Entity, but he wants the field to have a custom name of "Date Confirmed". What would you recommend he do?

- (●) Create a new Entity with the name "Date Confirmed". ✗

  Incorrect. Entities are similar to tables and records within them can have many fields. You wouldn't want to create a new Entity just to store a date when a field inside the existing entity already does that.

- ( ) Use the built-in Date field within the Entity to store the date values, but rename the field to "Date Confirmed".
- ( ) Create a new Environment with the name "Date Confirmed".

2. Your manager has built a solution using the Common Data Service using a few of the standard Entities included out of the box. He wants to capture the date that new records are added to the Entity, but he isn't sure how. He notices that a "Date" field is already inside his Entity, but he wants the field to have a custom name of "Date Confirmed". What would you recommend he do?

- ○ Create a new Entity with the name "Date Confirmed".
- ◉ Use the built-in Date field within the Entity to store the date values, but rename the field to "Date Confirmed". ✓

  Correct. When building a solution in the Common Data Service, it is best practice to use the out-of-the-box entities and fields where possible. In this case, the Entity already has a "Date" field. Best practices also state that you simply rename the existing fields if needed, so that would be the recommended solution here.
- ○ Create a new Environment with the name "Date Confirmed".

3. One of your co-workers has set up a Model-driven application using the Common Data Service. The app captures information about computer hardware assets, including their price. Your co-worker is concerned that users of the application are going to put incorrect values into fields - for instance, they might put in the wrong cost center when submitting a new record. How could this be prevented?

- ◉ Use Business Rules to automatically validate the cost center for the currently-logged in user. ✓

  Correct. A business rule could be set up to validate that the cost center the user has entered into the Model-driven app matches the cost center that is assigned to their department.
- ○ Use Power Automate to validate that the field value matches what the table is expecting.
- ○ Use a Canvas App instead of a Model-driven app.

3. One of your co-workers has set up a Model-driven application using the Common Data Service. The app captures information about computer hardware assets, including their price. Your co-worker is concerned that users of the application are going to put incorrect values into fields - for instance, they might put in the wrong cost center when submitting a new record. How could this be prevented?

- ○ Use Business Rules to automatically validate the cost center for the currently-logged in user.
- ◉ Use Power Automate to validate that the field value matches what the table is expecting. ✗

  Incorrect. You don't need a Flow to solve this issue, because the Common Data Service has built-in capabilities to handle this sort of data validation without introducing any additional workflow solutions.
- ○ Use a Canvas App instead of a Model-driven app.

Key takeaways

Here are the three key takeaways:

1. The Common Data Service uses standard entities, fields, and relationships to help you build powerful, scalable data solutions

2. Make your data work for you so that you can get the most of it by splitting it up into logical chunks.

3. Using the Common Data Service, you can break your data into various environments to better manage and secure important information.

Resources

Use these resources to discover more.

Tip

To open a resource link, right-click and select "Open in a new tab or window". That way, you can check out the resource and easily return to the module tab to unlock your achievement when done.

Common Data Service

- Introduction to Common Data Service
- Common Data Service Documentation
- Common Data Model
- Licensing

Entities, fields, and relationships

- Create and Manage Entities in Common Data Service
- Complex entities

- Restricted entities requiring Dynamics 365 licenses
- Create and manage fields within an entity in Common Data Service
- Create a relationship between entities in Common Data Service

Relationships and business rules

- Create and manage environments in Common Data Service
- Define and create business rules in Common Data Service

What is Power Apps and what it can do for you

- 3 minutes

Do you have inefficient or legacy business processes that you would like to modernize? Are you still moving information around using paper or even a shared Excel workbook? Do you want to be able to perform these business processes from different devices like PCs or mobile phones? Then you need Power Apps.

Power Apps is a no-code/low-code platform for building apps that builds off of concepts similar to formulas in an Excel workbook such as SUM and TEXT. You can use Power Apps to build simple solutions like vehicle inspection forms and status reports or complex business solutions for purchasing processes and inventory management. If you can envision an app to solve a business problem, then you can use your existing skills to build it. Although this module is geared towards business users with little background in computer science and coding, Power Apps offers advanced functionality and the ability for seasoned developers to design complex applications with ease.

Work with your data where it lives

When modernizing a paper-based process, there are likely systems in your organization with data you can leverage. With Power Apps, you have choices. With over 275 connectors you can easily connect to data, using the underlying data service and app platform, Common Data Service, or a multitude of online and on-premises data sources. Some common data sources include:

- Common Data Service

- SharePoint
- Dynamics 365
- SQL Server and Azure SQL
- Office 365

You don't have to choose just one data source. Power Apps easily supports multiple data connections allowing you to bring data together from many platforms into a single app.

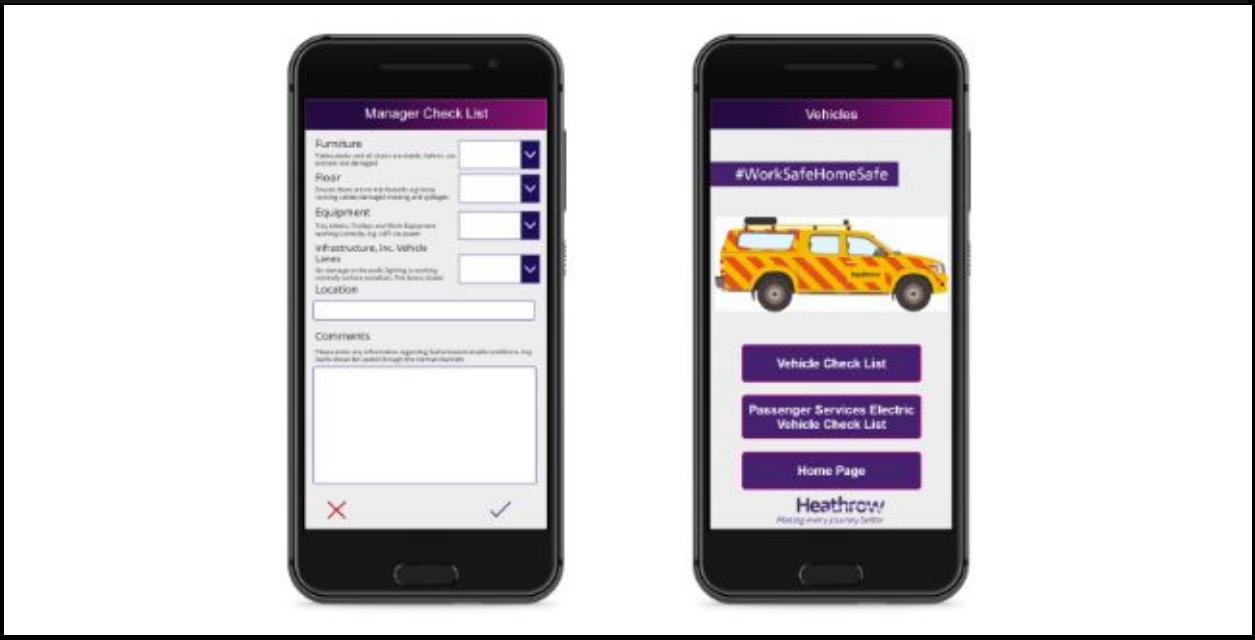
Different types of Power Apps for different scenarios

Power Apps can create three types of apps: canvas, model-driven, and portals. Each is suited to different scenarios and end users.

Canvas apps

Canvas apps are a great option when you want to build an app from a blank canvas. You start by choosing the screen size: tablet or mobile, then you have a blank screen from which to build. You can interact with data in your app by adding data sources. Drag and drop various controls and add the desired functionality by writing Excel style formulas. Canvas apps provide you complete flexibility when building your apps.

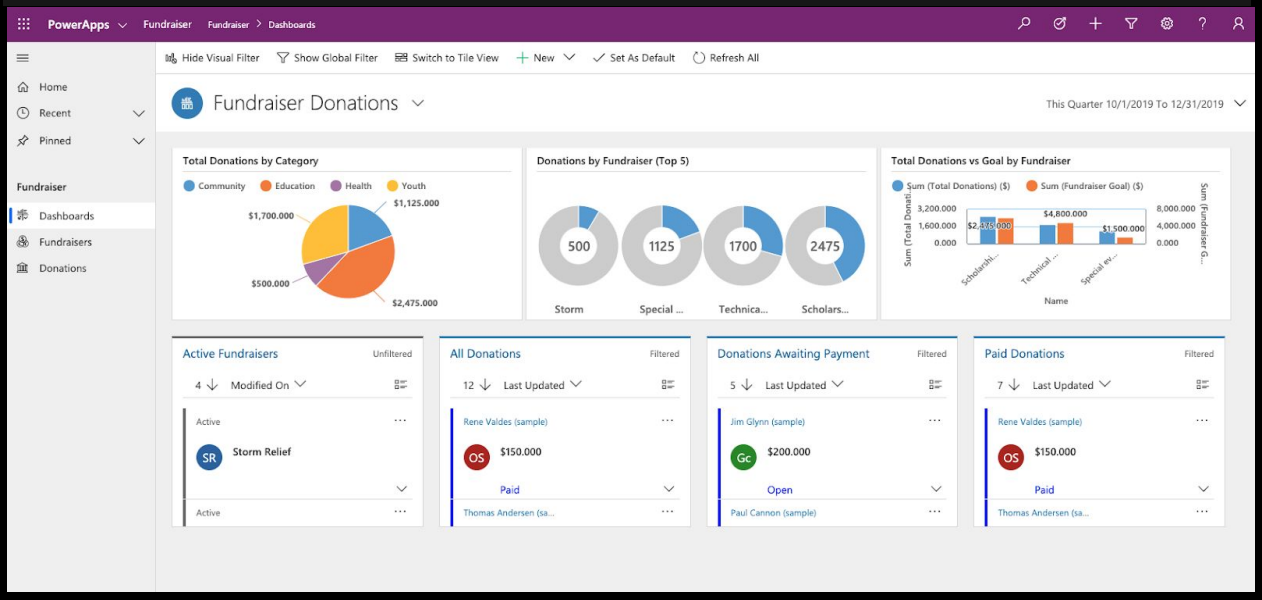
Below are a couple of examples of a mobile canvas app built by Heathrow Airport.



Model-driven apps

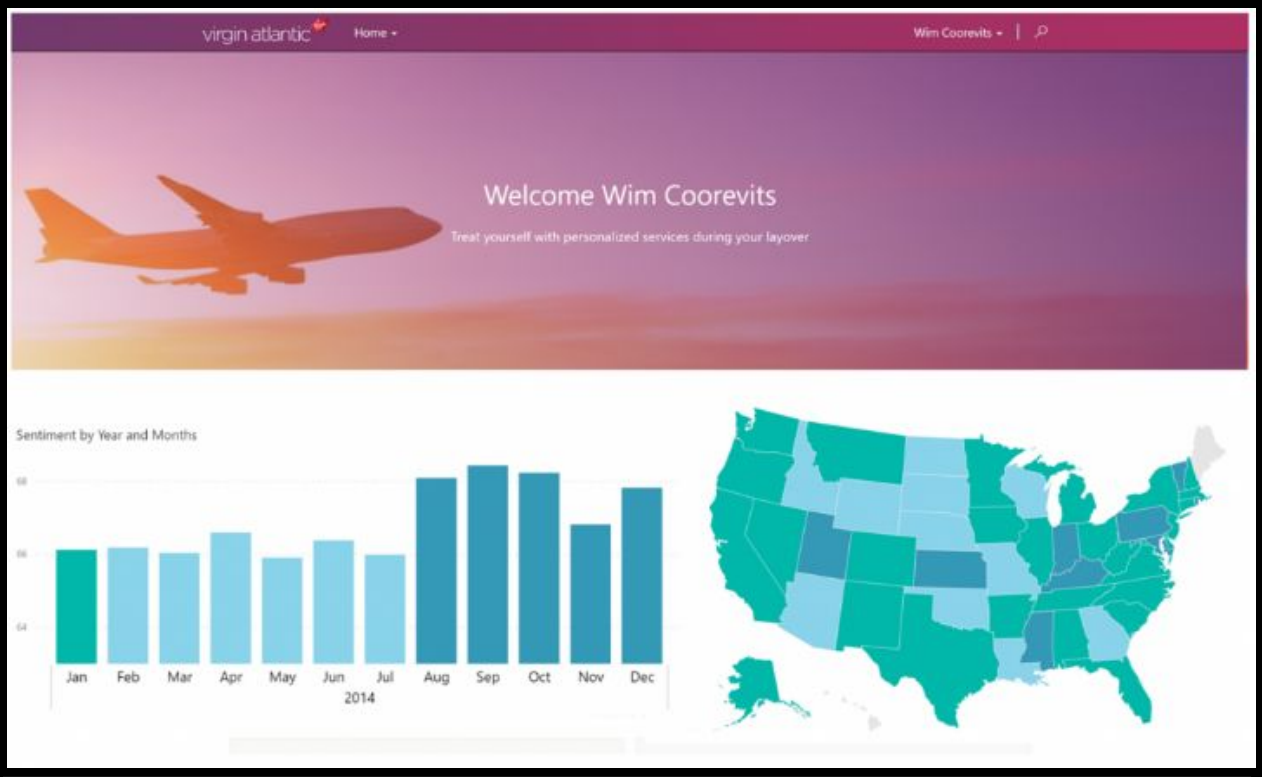
Model-driven apps build from data in the Common Data Service. Power Apps will build you a great looking, fully functional app to act upon and interact with this data. With model-driven apps, there is no need to worry about choosing the app size; it is responsive, meaning it works on mobile or tablet with no extra work by you. You define the relationships, forms, views, business rules, and more at the data layer, inside of the Common Data Service, giving you enough control to get your business result without writing all of the formulas yourself.

Below is an example of a fundraiser donations tracking Model-driven app.



Portals

Portals bring the power of no-code solutions to building externally facing websites. Through the Power Apps interface, you can build an anonymous or authenticated website that allows users to interact with data held in Common Data Service. The same drag and drop experience you enjoy when building apps is available to build these rich, interactive websites.



Add artificial intelligence to your app with no code

Prior to Power Apps, adding functionality such as image recognition or prediction models required advanced computer knowledge. There was code to write, data models to design and train, and a whole lot of complicated logic. Power Apps has "democratized" artificial intelligence by providing a wizard-based interface for building and training your model. This unlocks the power of Azure Machine Learning and Cognitive services without writing a single line of code or creating complex machine learning models.

A ready to use AI component is the Business card reader. This component reviews an uploaded photo or picture taken to determine if it is a business card and subsequently extracts the relevant information. No configuration required.

The business card reader is simply one ready-made example which builds off of the underlying AI components which you can deploy in other ways by first building a model. From https://make.PowerApps.com, a wizard guides you through building and training the model. Currently, there are four available AI models in Power Apps:

- Prediction - This model predicts whether something will happen or not based on previous data history. More details in the following section.
- Form processor - This model extracts text from an image like the business card reader.
- Object detector - This model identifies objects from an uploaded image or taken photo and then provides a count of the number of objects present.
- Text classification - This model categorizes text by its meaning, making it is easier to analyze.

The prediction model

The AI Builder prediction model allows you to create a model that can predict yes or no outcomes based on historical data. You train the model by providing historical data that includes the yes/no outcome and then artificial intelligence does the rest.

You can build prediction models to solve business problems such as:

- Will a lead become a customer?
- Will a project be profitable?
- Will a customer churn based on activity?

As you can see, AI can help you answer powerful business questions without writing a single line of code.

Security and Administration

There are many tools for those in IT or otherwise responsible for governance. Power Apps has a multitude of security, governance, and reporting capabilities to let you manage Power Apps. Also, Power Apps doesn't circumvent security in any way. Users cannot build apps to bypass current access permissions. To manage security for Power Apps, you can access https://admin.powerplatform.microsoft.com/. Here you will find options for creating and managing environments, monitoring licenses, working with Data Loss Prevention policies and managing Common Data Service Data Integration projects. This allows you to manage the Power Apps throughout your tenant from one single place.

Power Apps also has its own set of PowerShell cmdlets for app creators, administrators, and developers that allow you to automate many of your administrative duties. A

common use case of the PowerShell cmdlets is to automate the discovery and permission management of all apps in your tenant, allowing you to better understand and manage apps as they are created and spread throughout your company.

In addition, in the Power Automate learn module, you will see that Power Automate has the ability to automate these tasks. You can download the Center of Excellence starter kit, a collection of components or tools that are designed to help get started with developing a strategy for adopting and supporting the Power Platform, with a focus on Power Apps and Power Automate.

Driving business value

By now you understand that Power Apps does not require traditional "code" like C#, making a low technology barrier to entry. If you know your business process, you can get started writing your first app. For many businesses, this means apps are not coming from IT but instead directly from frontline business users. The same way Finance builds their own Excel workbooks today, they can build their own Power Apps.

Power Apps allows even the most entry level users to reduce paperwork, increase process efficiency, and ensure a single source of truth by combining multiple data sources into one app. Users can build apps with ease, while staying within the guardrails set by IT. This allows for an unprecedented amount of innovation and digitization, eliminating the app backlog, cumbersome paper-based processes and more.

In the next unit, you learn about how Heathrow Airport has built 30 apps that have eliminated 75,000 pages of paperwork, reduced data entry by nearly 1,000 hours, and saved the airport hundreds of thousands of dollars.

Customer case study - Heathrow Airport

- 3 minutes

Heathrow Airport is one of the world's busiest airports and models a small city in complexity and traffic. With over 76,000 people working on a given day and 200,000 travelers passing through, there's a strong need to continually optimize operations. One way to accomplish this goal is through the digitization of processes. Part of that effort included the adoption of Microsoft Office 365 for all employees, and later use of Power Apps

There are easily thousands of processes that could be turned into apps. These processes would require hundreds of developers and years of work to even make a

dent. While Heathrow has a robust IT department, they did not have hundreds of extra developers to begin digitizing processes. Or did they?

At Heathrow Airport, they have attacked the process of digital transformation by empowering the front line workers to build apps and solve problems. This revolution began with Samit Saini who decided to show what could be accomplished with Power Apps. Saini's business change manager requested a further look into the value Office 365, which led Saini to Power Apps. Saini began developing an app that which held security information translated into dozens of different languages. Previously, this information was kept in a book located at security checkpoints, and Saini set out to improve this experience. From the success of that first app, there are now dozens of apps and an official process for creating and implementing new apps.

In the full case study of Heathrow Airport, you'll learn the tremendous business value created by enabling front line workers to build apps using Power Apps. You'll also read about the support structures that they have put into place. These structures will help more people build their own apps and avoid duplication of efforts. Finally, you'll get an overview of the tremendous reduction of work and cost savings these apps have led to.

https://docs.microsoft.com/en-us/learn/modules/introduction-power-apps/4-power-apps-action

Portals are used to build a web-based experience that can be either authenticated or anonymous.
With a Model-driven app, you start from your data model, choose things such as views and business process flows, and then Power Apps automatically generates you an app.
Power Apps allows you complete control of which data sources you use in your app. There are over 270 data connectors available, and the ability to build a customer connector if one does not exist.

Check your knowledge

1. You want to build a mobile app that allows you complete control over the user experience and design. What type of Power Apps app should you build?

- ○ A Model-driven app
- ○ A Portal app
- ◉ A Canvas app ✓

  Correct. With Canvas apps, you start with a blank canvas choosing your type of mobile or tablet. Then you can design the app to meet your "pixel perfect" design.

2. Your projects requirements call for the ability to take a picture of a part and identify which model it is. You are not a data scientist or developer, how can you add this to your app?

- ◉ Train and use an Object Detection model in your app with AI Builder. ✓

  Correct. Power Apps makes available to you the ability to use artificial intelligence without writing code. This allows you to do Form Processing, Object Detection, Prediction, and Text Classification all powered by Azure services.
- ○ This is not possible.
- ○ Add a camera control to your app.

3. How can data sources be used with Power Apps Canvas apps?

- ○ Power Apps can only use its internal data sources.
- ○ Power Apps uses only external data sources and does not have an internal data source.
- ◉ Power Apps can use multiple internal and external data sources in the same app. ✓

  Correct. Power Apps allows you complete control of which data sources you use in your app. There are over 270 data connectors available, and the ability to build a customer connector if one does not exist.

Key takeaways

Here are the four key takeaways:

1. Power Apps is a no-code/low-code platform that allows you to build apps with your business knowledge and existing skills.

2. Power Apps has different offerings to meet your needs. Canvas apps, Model-driven apps, and Portals each have their own unique properties.

3. Power Apps helps you build and deploy customized apps that work across web and mobile, embedded or standalone, on any device.

4. You can build apps that follow your business processes instead of making your business process follow the software.

Resources

Use these resources to discover more.

Tip

To open a resource link, right-click and select "Open in a new tab or window". That way, you can check out the resource and easily return to the module tab to unlock your achievement when done.

Power Apps

- Power Apps
- Power Apps Resources

Learn more about AI Builder

- AI Builder
- AI Builder Documentation

Learn more about Portals

- Portals
- Portals Documentation

https://docs.microsoft.com/en-us/learn/modules/build-app-solution/1-introduction

Learn basic elements

- 3 minutes

Power Apps has many different components to build solutions including screens, inputs, galleries, forms and more. Let's review some of the most common elements you will need to get started.

Power Apps Studio

Power Apps Studio is the name of the web interface you use to build your app. With Power Apps, there is no client to download or install for building apps. Everything is done from the browser by logging into https://make.PowerApps.com.

App format

The first step in creating your app is to choose the format of your app: Mobile or Tablet. While both formats can be used interchangeably on a mobile device, a tablet, or a computer, each has different defaults around sizing of the screens and controls. Once you choose the format for an app, you cannot change it.

Galleries

The Gallery control is used to display records from a table of data. The display of a record is then defined by a template, which you can customize to meet your needs. This

allows you to control which fields are shown and how they are formatted. Power Apps will then apply this template automatically to every record in your data.

Forms

Forms are focused on working with a specific record, often based on a selection from a gallery. In this experience, a user browses a gallery to find and select the desired record displaying the details on the form. Forms enable a user to not only view detailed information, but to save new records and edit existing ones. The various actions performed with forms are controlled by form modes allowing the form to serve many purposes.

Input Controls

To allow you maximum flexibility in customizing your apps, Power Apps has a large selection of Input controls. Text inputs, buttons, dropdowns, toggles, date pickers, and sliders are a few examples. You can add these controls to galleries, forms, and screens to build a functional and aesthetic experience for your app. All inputs have a multitude of settings for default data, formatting, and actions which allow you to build an app that has the right user experience for your business process.

Intelligent Controls

In addition to common inputs as covered above, Power Apps also provides a rich set of controls for more advanced operations. There are hardware-based controls which allow access to the camera, bar code scanner, GPS, and more hardware features. There are also service backed controls like the business card reader or object detector which allow you to add artificial intelligence to your app without writing code.

Functions

Functions are the glue that binds all these controls, inputs, and data sources together. You can use one or more functions to create formulas in your apps. These formulas are similar to the language you use in Excel and can be used for actions such as sending data to a data source, formatting information, creating animations, and more. No complicated code is necessary, simply powerful functions with straightforward inputs to enhance your app.

Now that you have an overview of some of the rich functionality you can achieve with Power Apps, the next section will walk you down the path of building your first app.

Get started with functions in Power Apps

- 3 minutes

When using Microsoft Power Apps, you don't have to write complicated application code the way that a traditional developer does. However, you must express logic in an app and control its navigation, filtering, sorting, and other functionality. This is where formulas come in.

If you've used Microsoft Excel functions, you'll be comfortable building apps in Power Apps. To create a formula, you will combine one or more formulas with the required and optional parameters. Here are some common functions and an explanation of what they do:

- Filter - This function is often used with galleries or tables of data to narrow down the records returned from your data source. You do this by specifying one or more columns in your data set to perform a logic test on, which will allow you to return data that falls in a certain date range, has a set value, or was created by the user for example.
- Match - This function allows you to check a value to see if it follows a given pattern. You can use this to check if the user entered a properly formatted email address in the input field and if they did not show them a warning that a valid email is required. This function serves well for conditional formatting.
- Distinct - This function allows you to return the unique values from a list of data, making it easier to build dynamic dropdowns that show users only the valid values for the given field.
- Math functions - Power Apps includes a range of math formulas for working with your data from the simple such as Sum or Average to the complex such as Atan and Sin to work with radians.

This is a small sampling of the large library of Power Apps functions that are available. Also, remember you can combine functions into one formula to solve complex problems. This is the power of the platform. You start with simple formulas and then as your comfort grows you learn to combine them.

Check your knowledge

1. Your sales team is in desperate need of a mobile application that can display records living in a SharePoint list. When displaying the records, multiple pieces of information about each record need to be visible to the user. Which of the following Power Apps tools would you use when designing your app to achieve this functionality?

- (●) Create a gallery to display the sales records. ✓

  Correct. Galleries are used to display tables of data from your data source. They are highly customizable, so you can show whatever information about a particular record that is necessary. You could then use a Form to edit individual records, if desired.

- ( ) Create a form to display the sales records.
- ( ) Create an input control to display the sales records.

2. You have been tasked with creating a Power App that can scan barcodes that will display the scanned item's information on the screen. Out of the following control options, which one would the barcode scanner fall under?

- ( ) Galleries
- (●) Media ✓

  Correct. The barcode scanner control is a part of the Media category of controls. This control can be used to scan barcodes of various types and pass information about the barcode to the Power App. Then, additional Functions can use that barcode information to perform additional tasks.

- ( ) Forms

3. You have a gallery control in your app for displaying all customer orders. Your manager says she would like to see orders from the last 90 days in addition. How would you reduce the amount of data she sees?

- ( ) You would modify the data source to purge out all orders older than 90 days.
- ( ) Use the data filtering wizard.
- (●) Create a formula for your gallery that uses the Filter function. ✓

  Correct. The Filter function allows you to apply logic tests to one or more columns in your data source. For this scenario you could filter out data where the Order Date is less than 90 days ago.

Key takeaways

Here are the three key takeaways:

1. Power Apps can reference elements within the app to create dynamic solutions and a friendly interface.

2. With endless customization options, Power Apps can be used to enhance any business process.

3. Creating and managing apps are simple and easy to learn, even for those without a computer programming background.

Model-driven apps

- 3 minutes

Model-driven app design is a component-focused approach to app development. Model-driven app design does not require code, and the apps you make can be simple or very complex. Unlike canvas app development, where the designer has complete control over app layout, much of the layout is determined for you with model-driven apps and largely designated by the components you add to the app.

The approach to making model-driven apps

Model-driven apps have three design phases:

1. Model your business data
2. Define your business processes
3. Build the app

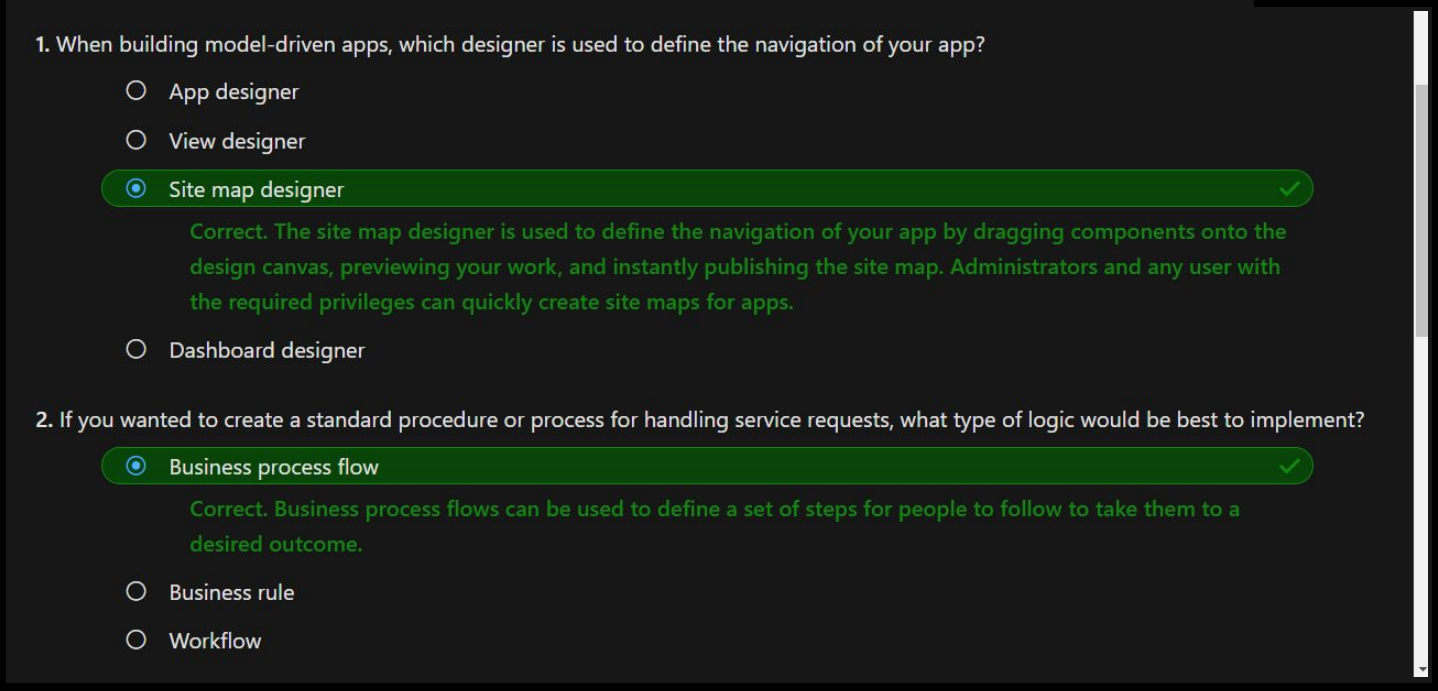
Model your business data

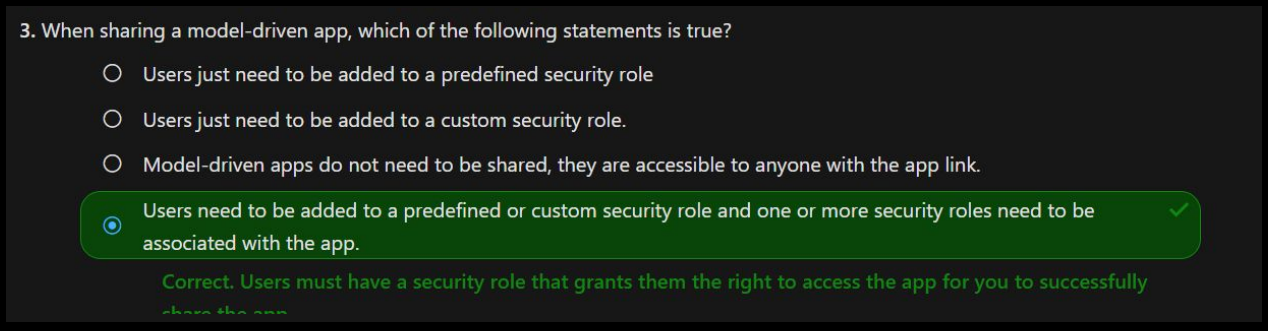
Model-driven design uses metadata-driven architecture so that designers can customize apps without writing code. To model business data, you determine what data the app will need and how that data will relate to other data. Metadata means *data about data* and defines the structure of the data stored in Common Data Service.

Define your business processes

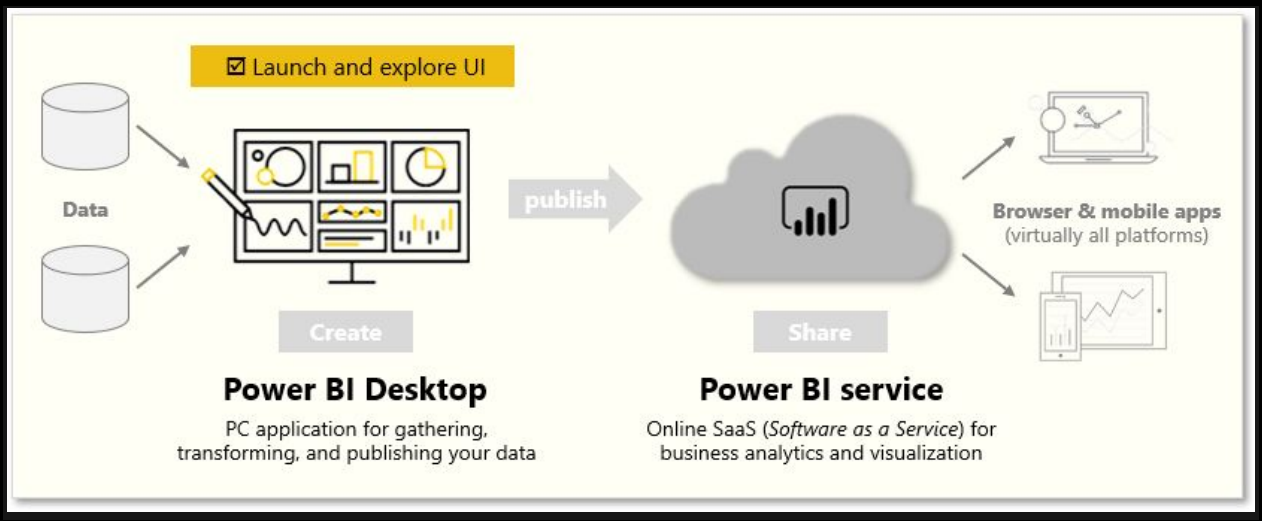
Defining and enforcing consistent business processes is a key aspect of model-driven app design. Consistent processes help ensure that your app users can focus on their work and not worry about having to remember to perform a set of manual steps. Processes can be simple or complex, and they often change over time.

Build the app





https://docs.microsoft.com/en-us/learn/modules/introduction-power-apps-portals/01-introduction-power-apps-portals



Answer the following questions to see what you've learned.

1. Where do you configure and customize your app?

- ○ In make.powerapps.com
- ○ In the Power Apps admin center
- ○ In Microsoft Dynamics 365
- ◉ In Power Apps Studio ✓

You can configure and customize your apps in Power Apps Studio.

2. Which of these statements about data sources is true?

- ○ In Power Apps, app data comes primarily from your local device.
- ○ Power Apps is not able to connect to external data. All data must be stored in the Power Apps app.
- ◉ Power Apps uses connectors to connect to Data Sources. If the data source supports it, Power Apps can read and write to the data source. ✓

You can use many external data sources, including Twitter, Facebook, Microsoft SQL Server, and Salesforce.

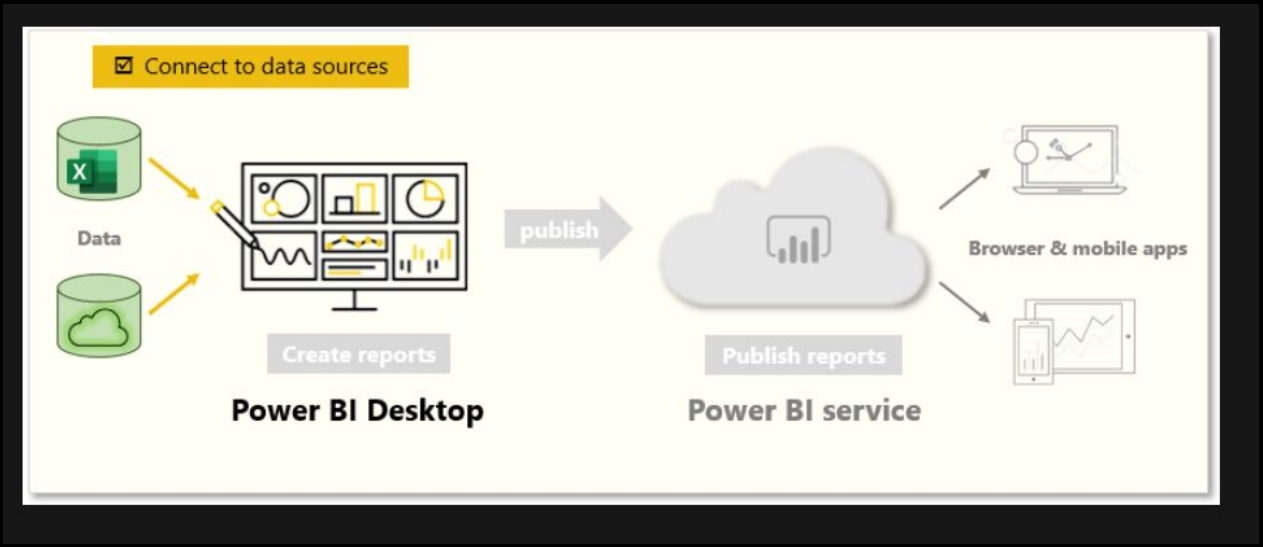
3. Power Apps automatically creates three screens for you when you build an app from a data source. Which of the following is not one of the screens created?

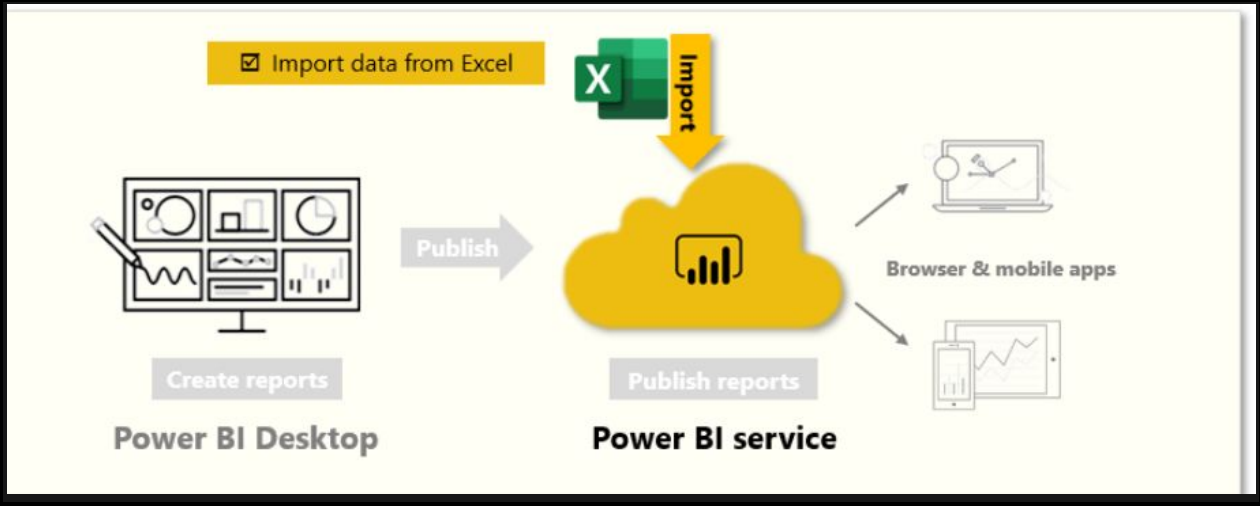
- ○ Browse Screen
- ◉ New Screen ✓

A Browse Screen, Edit Screen, and Details Screen are all created.

- ○ Edit Screen
- ○ Details Screen

https://docs.microsoft.com/en-us/learn/modules/get-data-power-bi/1-overview-power-bi-desktop





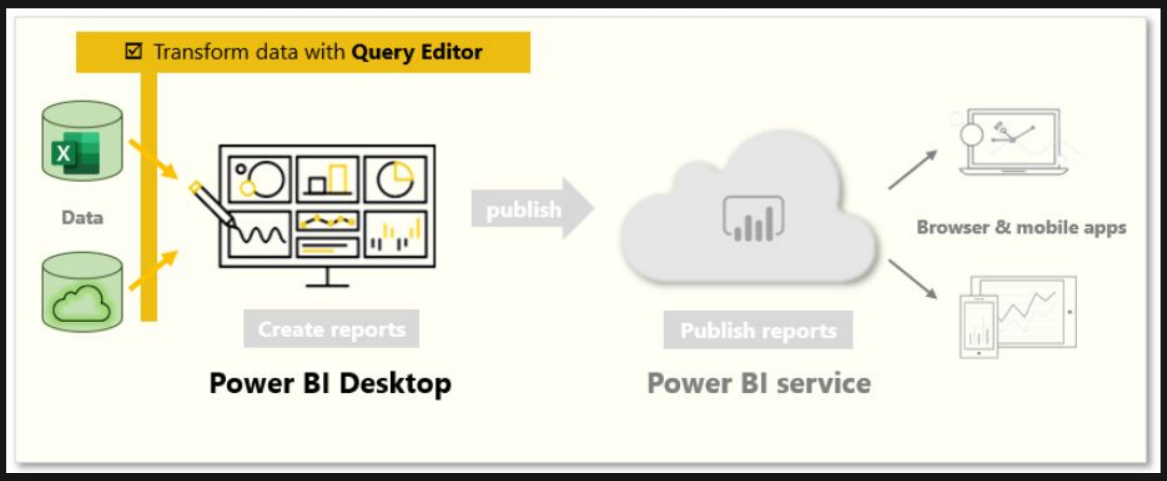
Transform data to include in a report

100 XP

10 minutes

Sometimes, your data might contain extra data or have data in the wrong format. Power BI Desktop includes the **Power Query Editor** tool, which can help you shape and transform data so that it's ready for your models and visualizations.

In this unit, you will transform data with Power Query Editor.



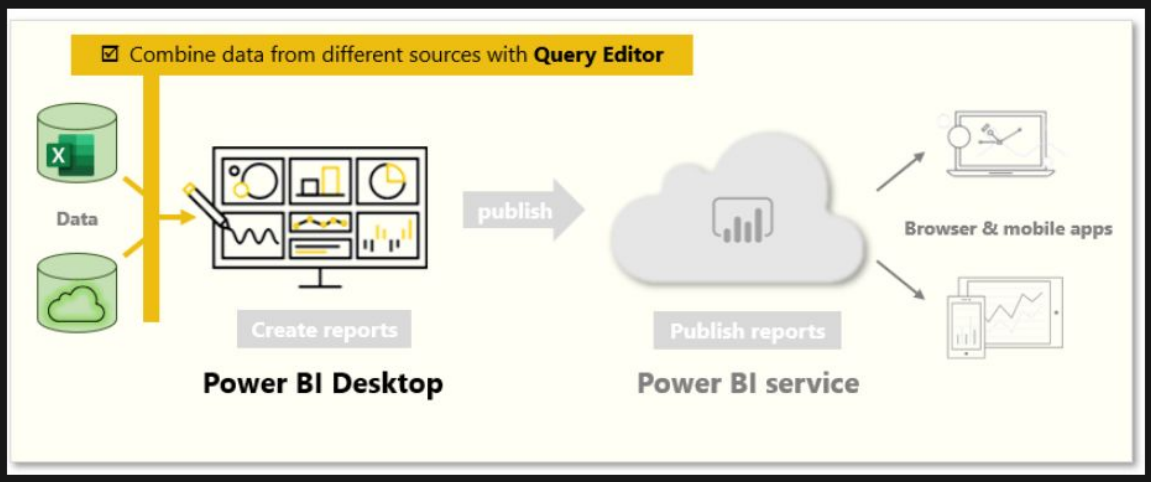
Combine data from multiple sources

100 XP

10 minutes

With Power BI Desktop, you can use the Power Query Editor tool to combine data from multiple sources into a single report.

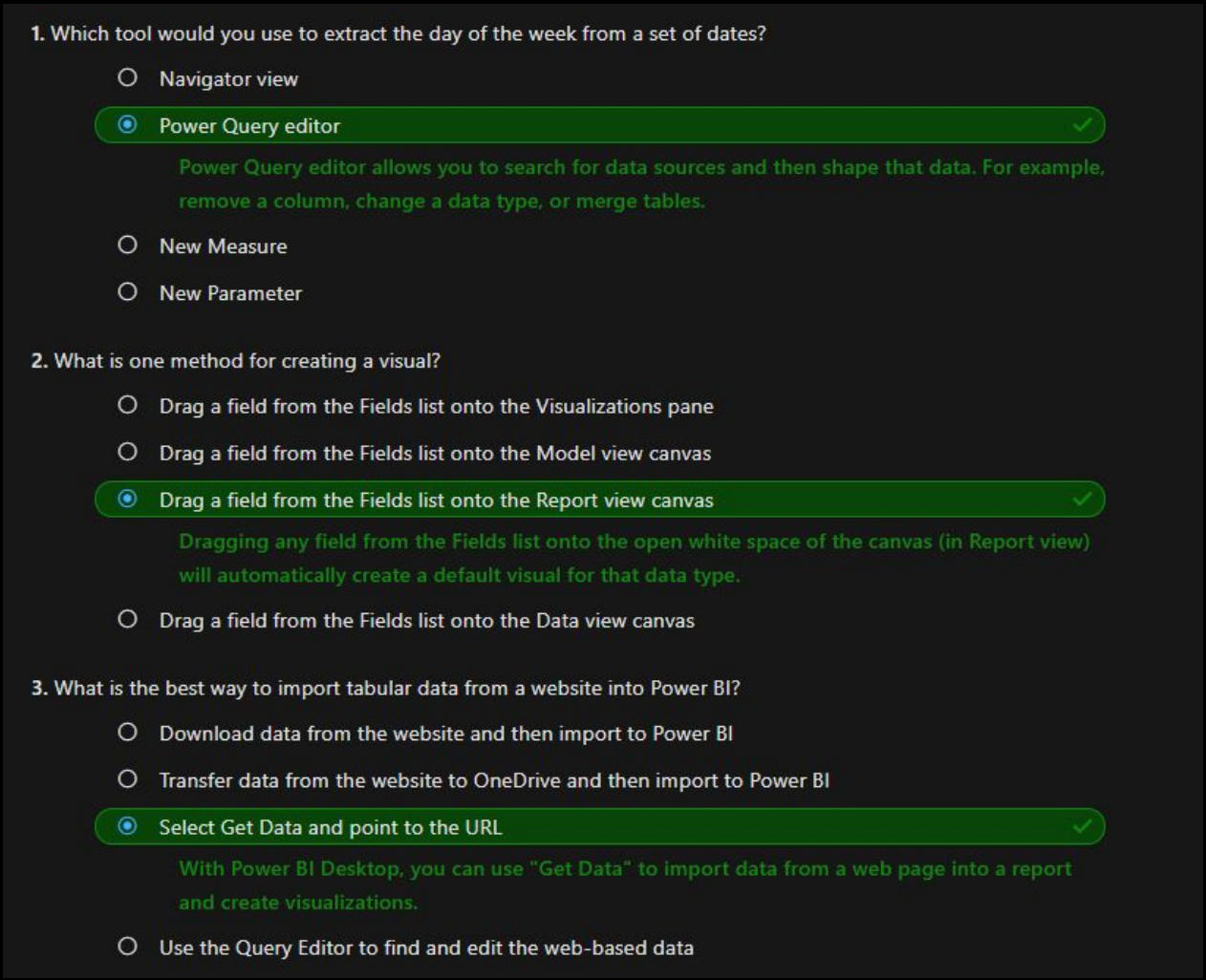
In this unit, you will combine data from different sources by using Query Editor.



Power Query editor allows you to search for data sources and then shape that data. For example, remove a column, change a data type, or merge tables.

Dragging any field from the Fields list onto the open white space of the canvas (in Report view) will automatically create a default visual for that data type.

With Power BI Desktop, you can use "Get Data" to import data from a web page into a report and create visualizations.



Knowledge check: Get started with Power Automate

1. How can you create a flow that reoccurs daily?

- ○ Create the flow and connect the flow to a calendar.
- ○ Create the flow and set the flow flag frequency to daily reoccurrence.
- ◉ Set the Schedule – Recurrence trigger frequency to daily and interval to 1. ✓

  This is how you can create a flow that reoccurs on a daily basis.
- ○ Create the flow and set the flow flag frequency to hourly and the value to 24.

2. Every flow has two main parts, a trigger, and one or more actions. Which of the following best describes an action?

- ◉ Actions are what you want to happen when a trigger is invoked. ✓

  Triggers invoke actions.
- ○ Actions will trigger the frequency of the flow reoccurrence.
- ○ Triggers are what you want to happen when an action is invoked.
- ○ Flow actions trigger the flow to occur as long as the defined value of the trigger is met.

3. You would like to be alerted to when a change is made to a SharePoint list, what are some ways in which Power Automate can help?

- ○ Power Automate can send up a flare near you to alert you of changes.
- ◉ You can write a flow with Power Automate to send push notifications, texts or emails for a variety of triggers. ✓

  Correct! Power Automate is good that way.
- ○ Sorry, Power Automate can't help with that.
- ○ Power Automate can send out a tweet whenever a list item is changed.

https://docs.microsoft.com/en-us/learn/modules/build-more-flows/1-flow-introduction

Check your knowledge

1. To secure and administer Power Automate you can use what resources?

- ○ Download the Power Automate administrator application.
- ○ Power Automate is only administered via the web portal.
- ◉ Power Automate can be administered on the web, via PowerShell, or even by building flows. ✓

  Correct. Power Automate provides **https://admin.powerplatform.microsoft.com**, PowerShell cmdlets, and flow actions for managing Power Automate. This give you complete flexibility for your Power Platform administration.

2. Where do you find flow templates?

- ○ You have to build and save templates yourself; there are no pre-built templates.
- ◉ Directly from the Power Automate builder website. ✓

  Correct. When you create a flow, you can choose to start from a template. There you can browse for templates or use search to find just the right template to begin your project.

- ○ You need to download the template pack from the Microsoft Learn website.

3. How can data sources be used with Power Automate?

- ○ Power Automate can only connect to Microsoft data sources such as Office 365 and Azure.
- ○ Power Automate requires you to build custom connectors to access external data.
- ◉ Power Automate can connect to data sources using one of the 260 plus prebuilt connectors or by building your own custom connector. ✓

  Correct. Power Automate allows you to automate business processes where your data lives.

Check your knowledge

1. One of your colleagues would like to set up a Flow to write an item to a SharePoint list when they receive an email from their boss. Which of the following would get them started?

- (●) See if any templates are available for this scenario. If one exists, use that, otherwise create a blank Flow with a "When an email is received" trigger. ✓

  Correct. Don't discount the template store in Power Automate; more often than not, someone has already solved this problem before. However, if they haven't, you can just create a new Flow and use the appropriate Trigger to achieve your goal.

- ( ) Create an "Instant" Flow and set the trigger to be "When an email is received", then add an action for creating the SharePoint item.
- ( ) Create an "Automated" Flow with only one action: create an item in SharePoint.

2. Your department would like to set up an approval system for items that are being written to a SharePoint list. You have been assigned the task of creating the system. Using Flow, what's the best approach?

- ( ) Create an Automated Flow from Blank and add a trigger for "When a new item is created" in SharePoint, and actions for the approvals.
- (●) Generate an Approval Flow from the built-in SharePoint Approvals functionality on your SharePoint list. ✓

  Correct. Why reinvent the wheel? SharePoint's built-in Approvals Flows are great, and using the information in this module you can set them up quickly and easily.

- ( ) Set up a blank Flow to send approval emails to the department manager when a new item is created in the SharePoint List.

3. Your organization has started using lots of Approval Flows, and users are complaining that they have too many approvals to keep track of. They don't know what is approved and what is still pending their approval. What is the easiest way for them to check?

- (●) Have the users navigate to the Approval Center in Power Automate. ✓

  Correct. In the Approval Center, users will be able to view their outstanding approvals as well as their approval history in a much easier-to-read platform. All of the approvals that require that user's interaction will be available through the Approval Center.

- ( ) Have the users set up Outlook rules to catch approvals so they are more visible.
- ( ) Send the users to the SharePoint list, which holds the records being approved to review which ones are approved and which ones are not.

4. One of your colleagues would like to post information about SharePoint items that have been approved to Twitter. Is this possible? If so, how would he do it?

- ( ) No, it's not possible. SharePoint lists don't connect to Twitter directly, so there's no way to Tweet about approved items.
- ( ) Yes, it's possible. He can create a Flow to monitor his SharePoint list for approved items, then notify him so he can Tweet the details manually.
- (●) Yes, it's possible. He can create a Flow to monitor his SharePoint list for approved items, then Tweet the details. ✓

  Correct. This was covered in this module, so you can refer back to earlier in the module to see how he could achieve it.

Introduction

- 3 minutes

Power Virtual Agents are adaptable Artificial Intelligence (AI) chatbots, at your service. They can solve common customer and internal-facing issues automatically, freeing up staff to focus on complex requests and high-value interactions. You can easily create your own virtual agents, powerful chatbots, without the need for developers or data scientists, by using a guided, no-code graphical interface. Integrate Power Virtual Agents with the products and services you use every day using hundreds of prebuilt connectors, by building custom workflows using Power Automate, or creating complex scenarios with Microsoft Bot Framework. Monitor and continuously improve chatbot performance using AI- and data-driven insights available in an easy-to-read dashboard.

In this module, you will:

- Learn how to describe the business value and features of Power Virtual Agents

- Learn essential components that make up Power Virtual Agents and chatbots

1. When building model-driven apps, which designer is used to define the navigation of your app?

- ○ App designer
- ○ View designer
- ◉ Site map designer ✓

  The site map is used to define the navigation of your app.
- ○ Dashboard designer

2. If you wanted to create a standard procedure or process for handling service requests what type of logic would be best to implement?

- ◉ Business process flow ✓

  Business process flow logic should be used.
- ○ Business rule
- ○ Workflow
- ○ Flow

3. When sharing a model-driven app, which of the following statements is true?

- ○ Users just need to be added to a predefined security role.
- ○ Users just need to be added to a custom security role.
- ○ Model-driven apps do not need to be shared, they are accessible to anyone with the app link.
- ◉ Users need to be added to a predefined or custom security role and then you also need to assign one or more security roles to the app. ✓

  To share a model-driven app, you need to add a user to a security role that is assigned to the app.