Power Platform is comprised of four key products: Power Apps, Power Automate, Power BI and Power Virtual Agents.

# Data Connectors

1. Data Sources- The two types of data sources are tabular and function-based.

* **Tabular data** - A tabular data source is one that returns data in a structured table format. Power Apps can directly read and display these tables through galleries, forms, and other controls. Additionally, if the data source supports it, Power Apps can create, edit, and delete data from these data sources. Examples include Microsoft Dataverse, SharePoint, and SQL Server.
* **Function-based data** - A function-based data source is one that uses functions to interact with the data source. These functions can be used to return a table of data but offer more extensive action such as the ability to send an email, update permissions, or create a calendar event. Examples include Office 365 Users, Project Online, and Azure Blob Storage.

1. Connectors- Connectors are divided into standard and premium.

* Some popular standard connectors are SharePoint, Outlook, and YouTube. Premium connectors require additional licensing for your app and/or users.
* Connectors can provide input and output between the data source and Power Platform, which can accelerate the delivery of Power Platform business solutions.

## Triggers and Action

* **Triggers** are only used in Power Automate and prompt a flow to begin. Triggers can be time based, such as a flow which begins every day at 8:00 am, or they could be based off of an action like creating a new row in a table or receiving an email. You will always need a trigger to tell your workflow when to run.
* **Actions**are used in Power Automate and Power Apps. Actions are prompted by the user or a trigger and allow interaction with your data source by some function. For example, an action would be sending an email in your workflow or app or writing a new line to a data source.

## Custom Connectors

* An advantage of building custom connectors is that they can be used in different platforms, such as Power Apps, Power Automate, and Azure Logic Apps.

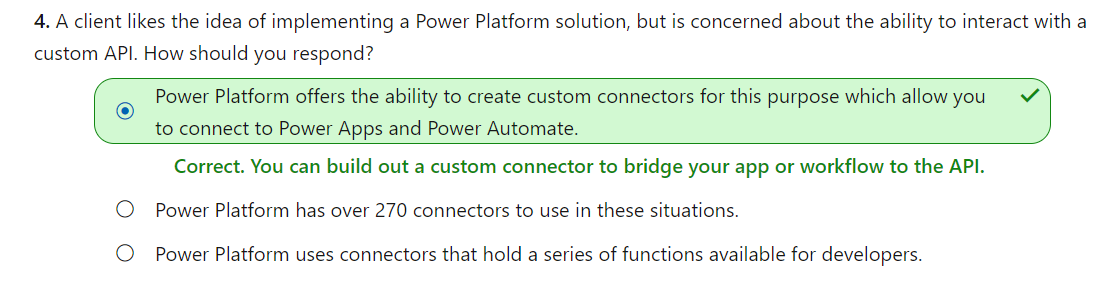
# Data loss prevention, compliance, privacy, and accessibility

## Data loss prevention policies

* You can create data loss prevention (DLP) policies that can act as guardrails to help prevent users from unintentionally exposing organizational data. DLP policies can be scoped at the environment level or tenant level
* For tenant-level policies, you can define the scope to be all environments, selected environments, or all environments except ones you specifically exclude.
* Connectors can be classified as either **Business** or **Non-Business** in the context of your organization. Connectors that host business-use data should be classified as **Business** and connectors that host personal-use data should be classified as **Non-Business**. Any connectors that you want to restrict usage of across one or more environments should be classified as **Blocked**. When a new policy is created, all connectors are defaulted to the **Non-Business** group. From there they can be moved to **Business** or **Blocked** based on your preference. You can manage connectors when you create or modify the properties of a DLP policy from the [Power Platform admin center](https://admin.powerplatform.microsoft.com/). These affect Power Platform canvas apps and Power Automate flows. To create a DLP policy, you need to be a tenant admin or have the Environment Admin role.

## Accessibility in Power Platform

* You can use the [Accessibility Checker](https://docs.microsoft.com/en-us/powerapps/maker/canvas-apps/accessibility-checker) to help review potential accessibility issues in your app.



# Overview of Dataverse

1. Microsoft Dataverse is a very powerful cloud-based solution for storing and working with your business data.

* **Security**: Dataverse 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 column level and provides rich auditing capabilities.
* **Logic**: Dataverse 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**: Dataverse 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**: Dataverse 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**: Dataverse connects in different ways to support your business needs. APIs, webhooks, eventing, and data exports give you flexibility to get data in and out.

## 2. Common Data Model vs. Microsoft Dataverse

The standard table design in a Microsoft Dataverse 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 tables and relationships that Microsoft and its partners have published in an industry-wide initiative called the Open Data Initiative. This collection of predefined tables, columns, semantic metadata, and relationships form the basis of the Common Data Model.

# Tables

1. A table is a logical structure containing rows and columns that represents a set of data.

2. The two types of tables are:

* **Standard** - The base set of tables that are created for every instance of a Microsoft Dataverse database. You can add more columns to any table, but you can only delete columns from a custom table.
* **Complex** - Tables that contain complex, server-side business logic, including real-time workflows or plug-ins. Some of the tables that are used in Dynamics 365 applications are complex.

3. Columns have data types, meaning that you can store data of a certain type in a column that matches that data type.

4. Tables that relate to one another have a relational connection. Relationships between tables exist in many forms, but the two most common are one-to-many and many-to-many, both of which are supported by Microsoft Dataverse.

* One-to-many relationships are also known as parent-child relationships. In the previous invoice example, the invoice table would be the parent and the line items would be a child table.
* A column that only allows unique values, such as invoice number, is used to identify the parent row. This unique column is called a key. The same value (the parent key) is stored in the related child rows. This column is called a foreign key when the child row is used to store the parent key value.

# Environments in Dataverse

1. Environments are used to store, manage, and share your organization's business data, apps, and flows in Power Platform. Each environment allows you to provision one Microsoft Dataverse database for use within that environment.

2. 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.

3. 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.

# Business Rules

1. In Microsoft Dataverse 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 Microsoft Dataverse, they are in effect regardless of where you interact with the data.

2. Business rules are usually defined for a table 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.

3. The following business rule actions can be used by canvas and model-driven apps:

* Set column values
* Clear column values
* Validate data and show error messages

4. Model-driven apps can also use business rules to:

* Show or hide columns (model-driven apps only)
* Enable or disable columns (model-driven apps only)
* Create business recommendations based on business intelligence (model-driven apps only)

5. The rule is tied to the data, not the app.

# Administer

1. Power Platform admin center

The Power Platform admin center lets you manage the tasks of setting up users, permissions, and many other important features and capabilities of Microsoft Dataverse.

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

# Power Apps

# Power Apps can create three types of apps: canvas, model-driven, and portals.

# 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.

# Model-driven apps build from data in Microsoft Dataverse. 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 Dataverse, giving you enough control to get your business result without writing all of the formulas yourself.

1. 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 Dataverse.

## Add artificial intelligence to your app with no code. 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. 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.
* 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.
* AI Builder entity extraction models recognize specific data in text that you target based on your business needs. The model identifies key elements in the text and then classifies them into predefined categories. This can help you transform unstructured data into structured data that's machine-readable. You can then apply processing to retrieve information, extract facts, and answer questions.

### 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?

# How to build a canvas app

1. Power Apps Studio is the name of the web interface you use to build your app.
2. **App format.** The first step in creating your app is to choose the format of your app: Mobile or Tablet. Once you choose the format for an app, you cannot change it.

## **Galleries.** The Gallery control is used to display rows from a table of 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 row to view the details on the form. Forms enable a user to not only view detailed information, but to save new records and edit existing ones.

## **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.

## **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.

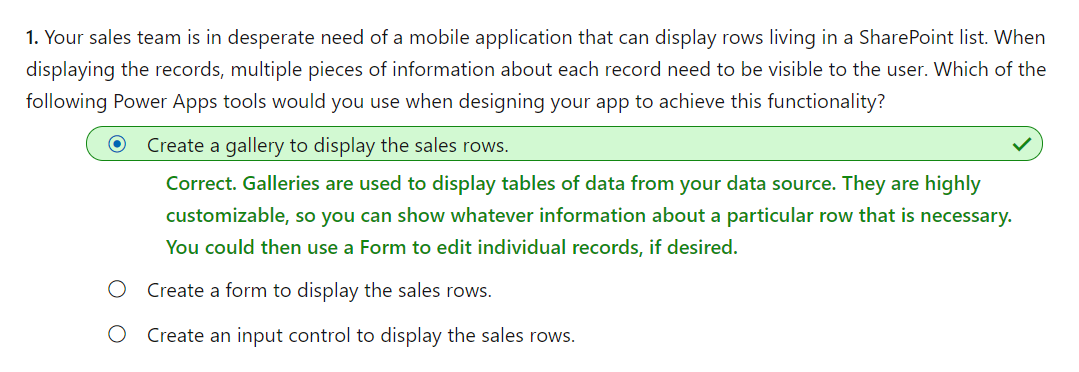
## **Functions.** Functions are the glue that binds all these controls, inputs, and data sources together. You can use one or more functions to create formulas in your apps.

## **Prepare to share an app.** By default, the user receives the User permission. If you want the user to also be able to edit the app, then select the co-owner check box. The following is a description of both permissions:

* **Co-owner** - Users can use, edit, and share the app, but can't delete or change the owner.
* **User** - Users can view and use the app, but they can't change it.

1. **Consider security groups.**

* If you share an app with a security group, existing members of that group and anyone who joins it will have the permission that you specify for that group. Anyone who leaves the group loses that permission unless they belong to a different group that has access or if you give them permission as an individual.
* Every member of a security group has the same permission for an app as the overall group does. However, you can specify greater permissions for one or more members of that group to allow them greater access. For example, you can give Security Group A permission to run an app, but you can also give User B, who belongs to that group, Co-owner permission. Every member of the security group can run the app, but only User B can edit it. If you give Security Group A Co-owner permission and User B permission to run the app, User B can still edit the app.



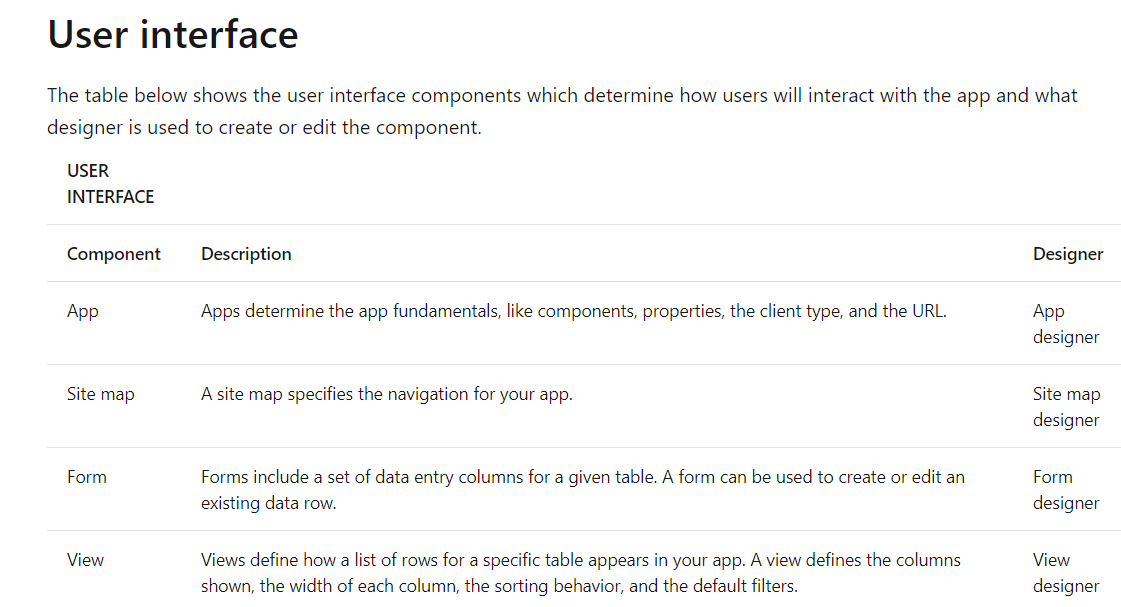
# Introduction to model-driven apps

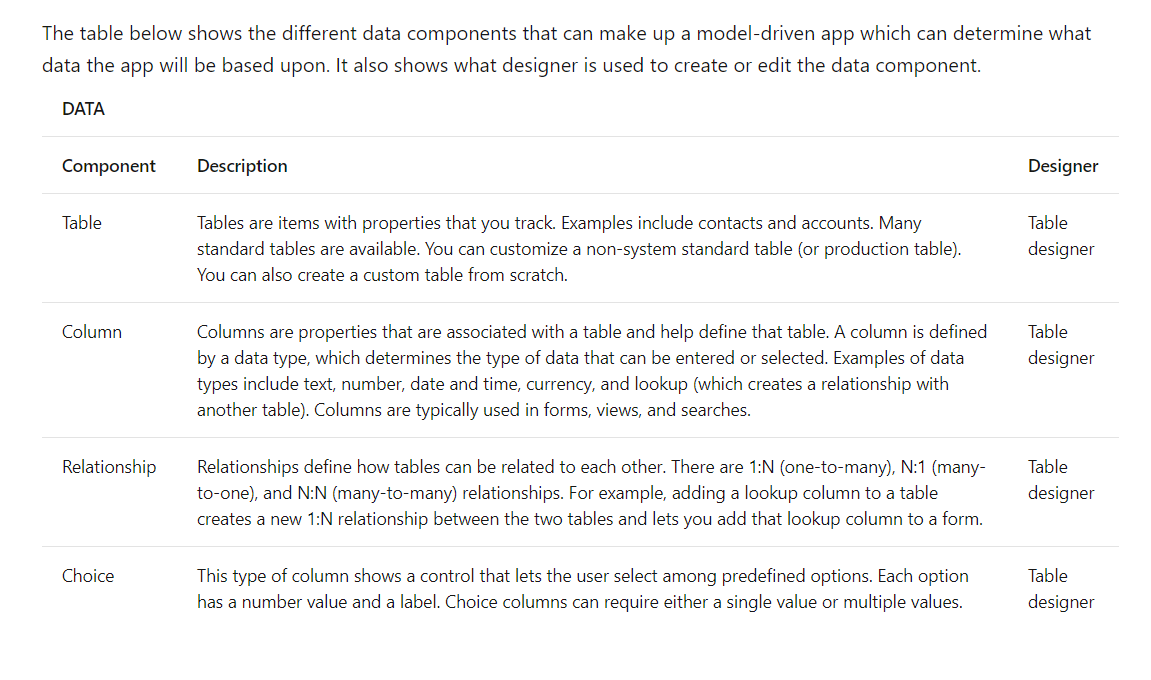
1. Model-driven app design is an approach that focuses on adding dashboards, forms, views, and charts to your apps. In canvas apps, the app maker has total control over the app layout. In model-driven apps, on the other hand, much of the layout is determined by the components you add. The emphasis is more on quickly viewing your business data and making decisions instead of on intricate app design.

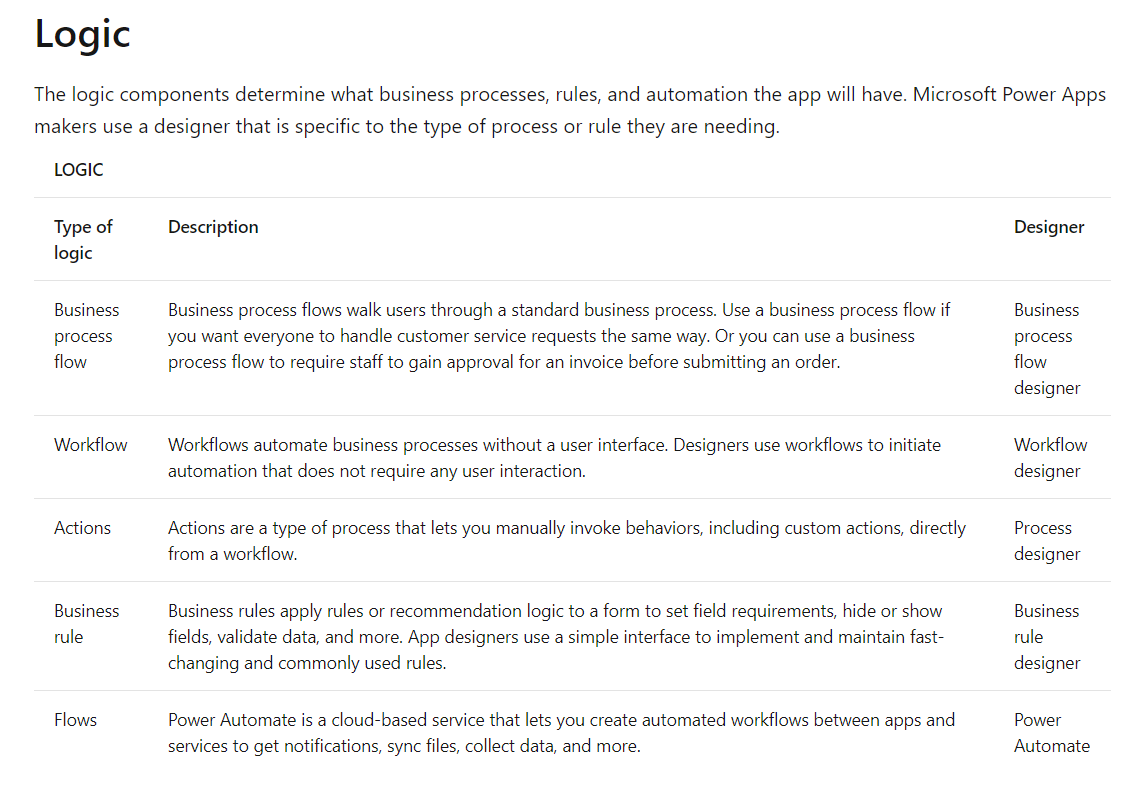
## The approach to making model-driven apps

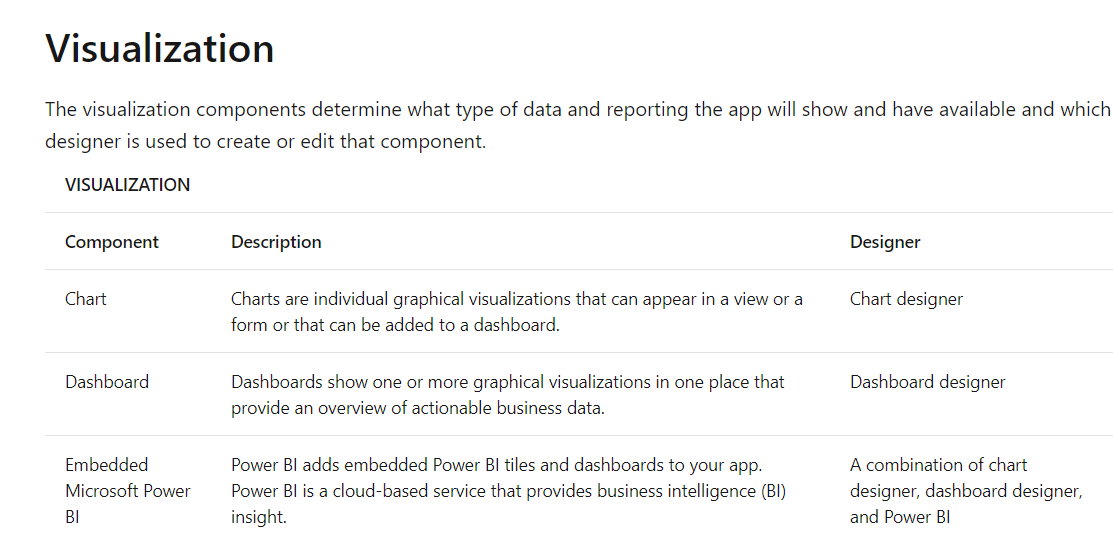
Model-driven apps have three design phases:

* Model your business data
* Define your business processes
* Build the app









### **Data model.** model-driven applications use a metadata-driven architecture. This means a large portion of the model-driven app is based on how your data is modeled, and there is no need to write custom code to alter the app design.

### **Business Logic.** When incorporating business logic in your app, there are two primary options available. You can set **Business Rules** on your Microsoft Dataverse tables or you can build **Business Process Flows**.

* With **Business Rules,** you will define behaviors at the data layer. This is great for setting conditions for when a field is required, setting a default value, or even showing or hiding a field based on criteria. An example could be a table for tracking expenses. You could have a column for type of travel and then build a business rule that dictates that if a user chooses automobile then the mileage field is required, else it is optional. This gives you the power to make sure you maintain data consistency in all scenarios.
* **Business process flows** are used to guide users through using your app. These workflows can provide visuals on next steps based on the status of the data and facilitate other actions that you want to occur as the user uses the app. Business Process Flows let you bring automation to your app and make it more of a guided experience than just a place to enter data.

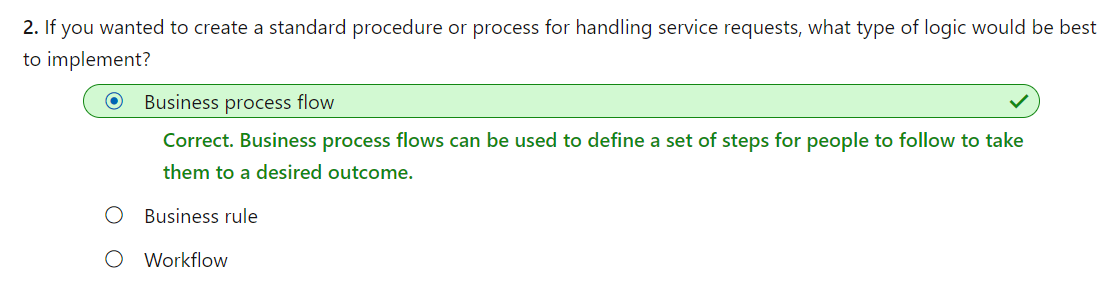
### Control security and share model-driven apps

* To share an app, you must have the Environment Admin or System Admin role.
* Create or set up a security role. Apps can be based on a custom table. Because the table is custom, privileges must be explicitly specified before users can work in it. To do this, you can use either of the following approaches:
* Expand an existing predefined security role so that it includes privileges for rows that are in the custom table.
* Create a custom security role to manage privileges for users of the app.

## 

## Share the link to your app

Sharing a model-driven app involves two primary steps. First, associate one or more security role(s) with the app, then assign the security role(s) to users. Unlike sharing canvas apps, sharing model-driven apps does not currently send an email with a link to the app.



# Power Apps portals

* 1. Power Apps portals give internal and external users secure access to your data either anonymously or through commercial authentication providers like LinkedIn, Microsoft, Facebook, and Google, or enterprise providers such as Azure AD B2C and Okta. Portals also allow you to set authentication requirements, customize data for each user, and allow users to submit their information privately with straightforward admin controls.
  2. The following are additional capabilities that Power Apps portals can deliver:

**Provide self-service support** - When your business is growing, rather than having to employ extra staff in call centers, you could use Power Apps portals to add self-service capabilities to your website so that your customers can search knowledge articles, engage with other customers, find answers, and create support cases when needed (that go directly into Dynamics 365 Customer Service), all without a single interaction from your resources.

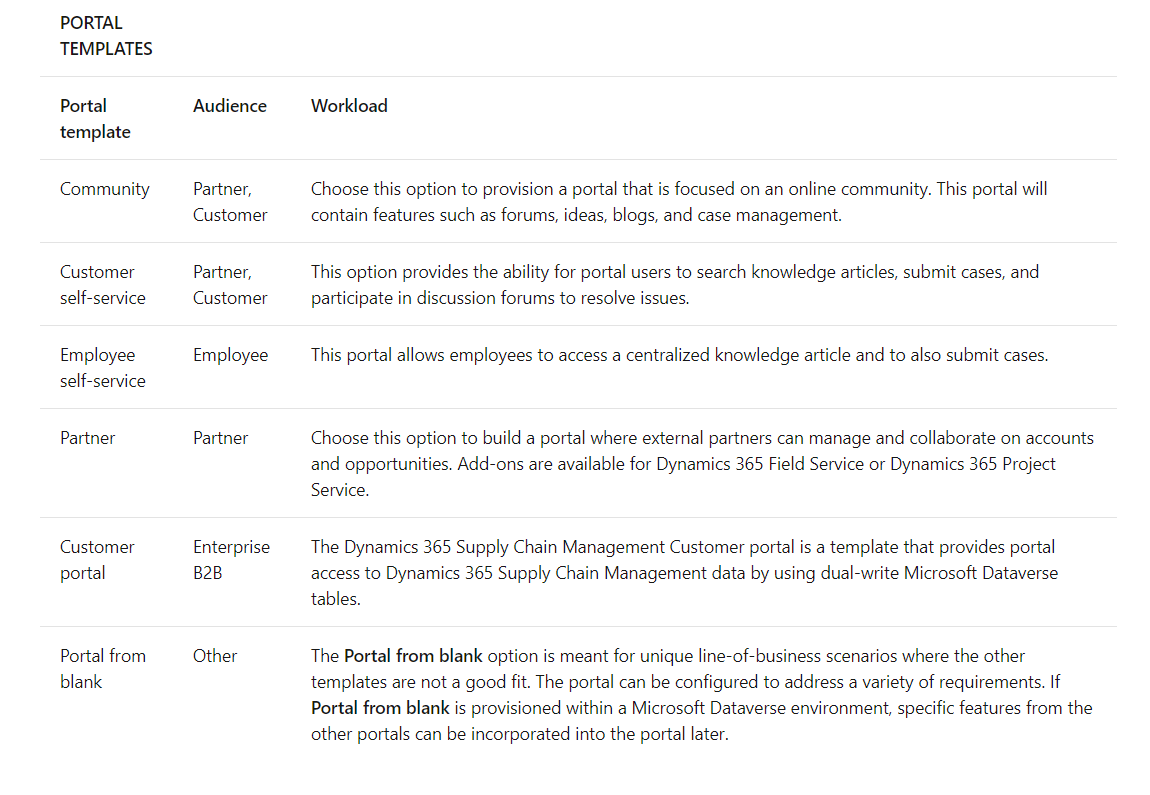
**Build a sales pipeline** - When a lead fills out a **Contact Us** form on your company website, this information is recorded in Dynamics 365 Sales where the record can become part of your sales pipeline automatically.

**Empower employees** - When an employee needs a new computer, they can fill out an online form, where the information will be recorded in Dataverse so the helpdesk staff can immediately access and process this information.

**Engage mobile workforce** - Empower agents on any device, wherever they work. Field technicians can process and complete work orders in the field, instantly updating Dynamics 365 Field Service.

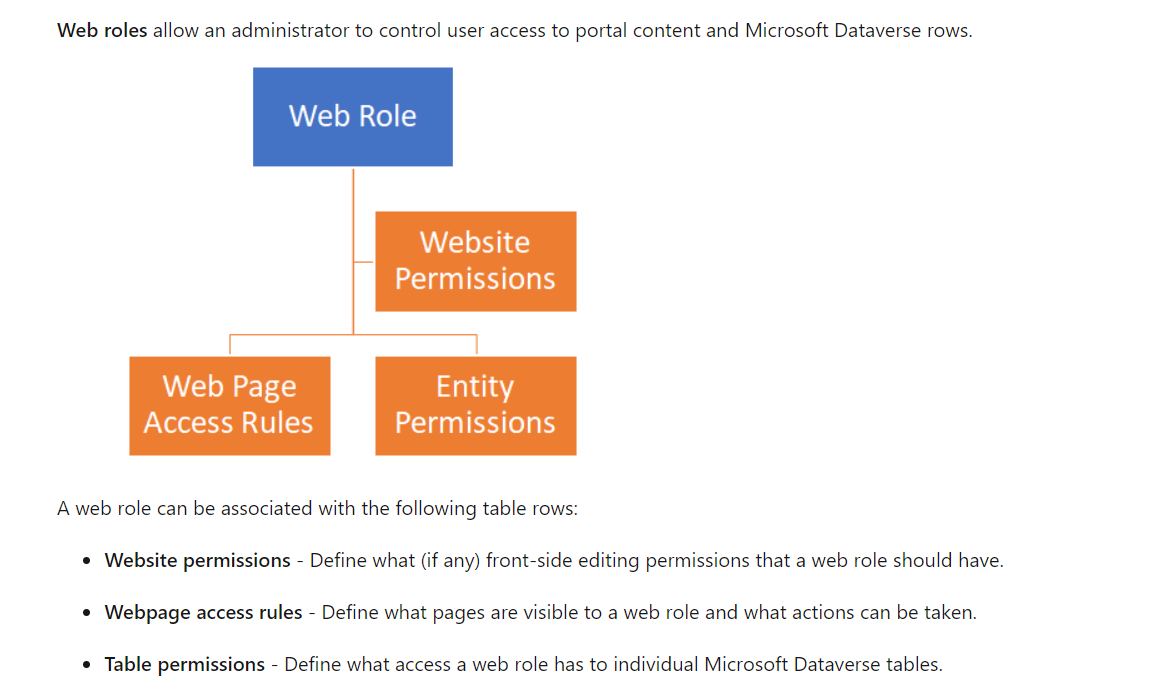
* 1. Power Apps portals are built on top of Microsoft Dataverse. This architecture comes with a major benefit. All the differentiating features of model-driven Power Apps are the features of Power Apps portals as well, including:
* Centralized management
* Common Data Model
* Roles and permissions
* Forms and views
* Business rules
* Declarative workflows and actions
* Plug-in architecture
* Integration with other services
* Microsoft Dataverse extensibility
* Audit

4 . A Power Apps portal is not automatically provisioned when a new Dataverse environment is created. You will need to provision a Power Apps portal and determine the name, default URL, language, and template. To provision a portal, you must be assigned the System Administrator role of the Microsoft Dataverse environment that is selected for the portal.

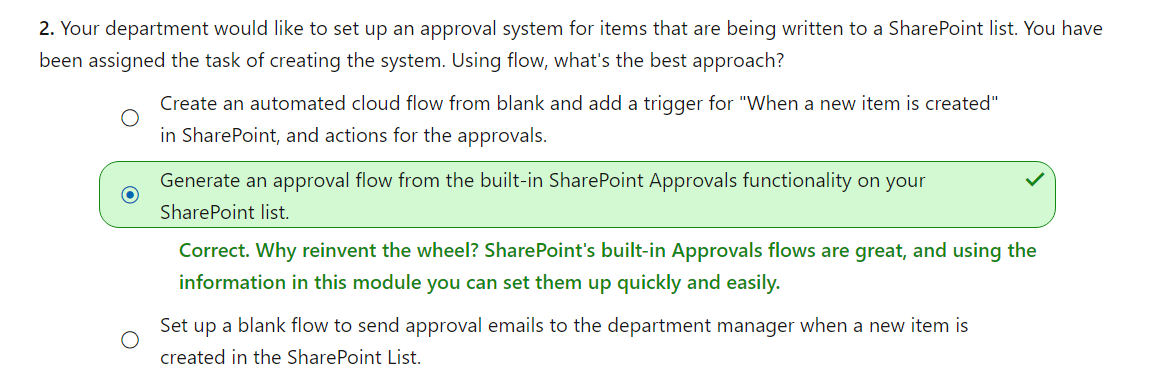


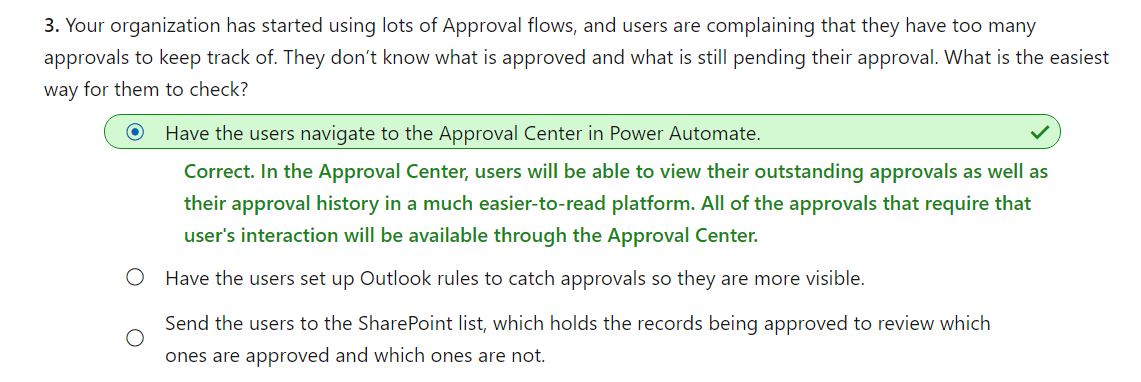
1. Only one Power Apps portals can be provisioned for each Microsoft Dataverse environment.

## Authorization



One of the web roles in the portal can be marked as **Anonymous** and all of the others are **Authenticated**. These roles allow you to apply permissions and access rules to all portal users based on whether they are signed in. If a user is not signed in, they will view the portal with the Anonymous web role permissions, which should be the most restrictive permissions.





# Power BI

1. Power BI consists of a Microsoft Windows desktop application called Power BI Desktop, an online SaaS (Software as a Service) service called the Power BI service, and mobile Power BI apps that are available on phones and tablets.

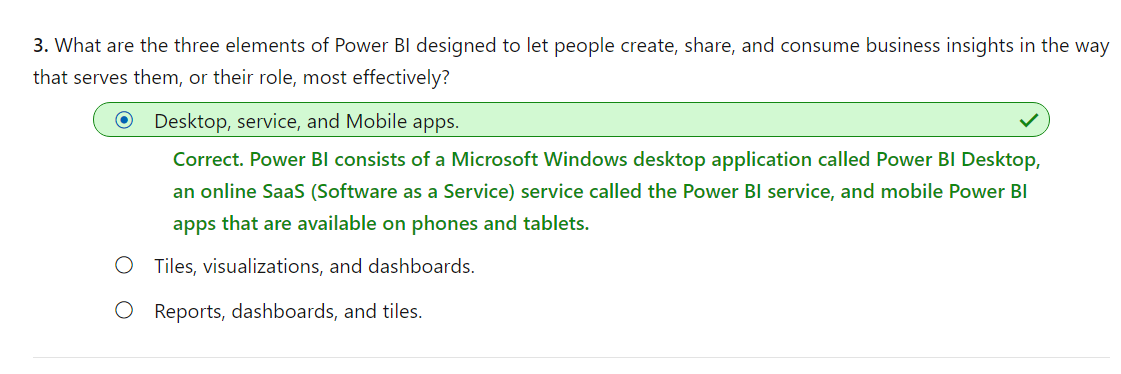
2. The major building blocks of Power BI are: datasets, reports, and dashboards. They are all organized into workspaces, and they are created on capacities.

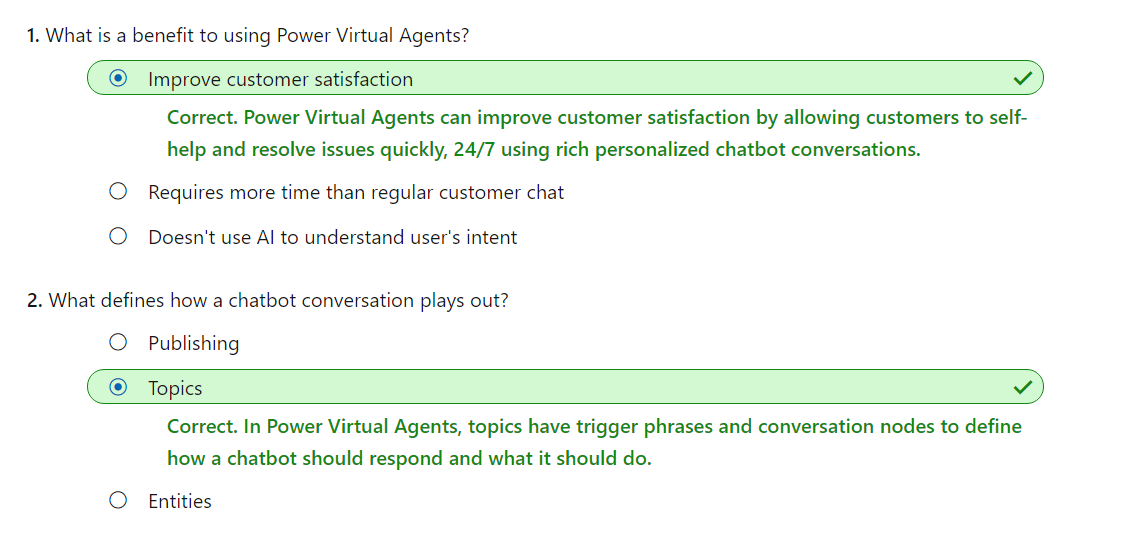
3. Capacities are a core Power BI concept representing a set of resources used to host and deliver your Power BI content. Capacities are either shared or dedicated. A shared capacity is shared with other Microsoft customers, while a dedicated capacity is fully committed to a single customer. Dedicated capacities require a subscription. By default, workspaces are created on a shared capacity.

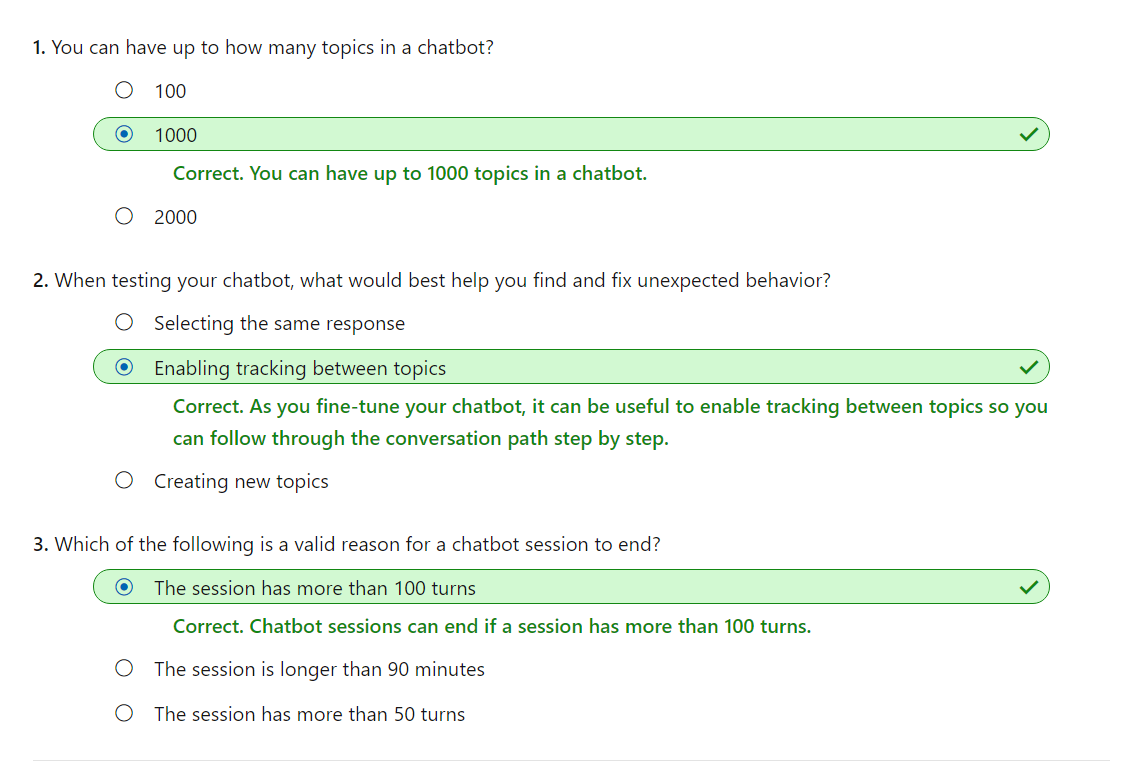
4. Workspaces are containers for dashboards, reports, datasets, and dataflows in Power BI. There are two types of workspaces: My workspace and workspaces.

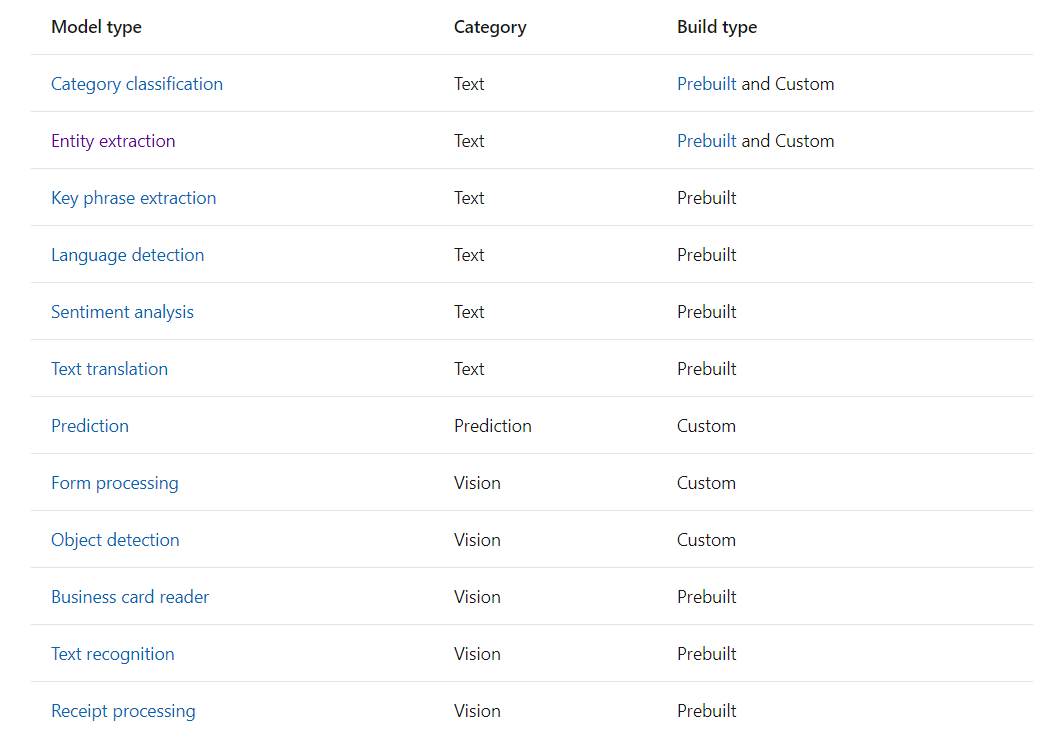
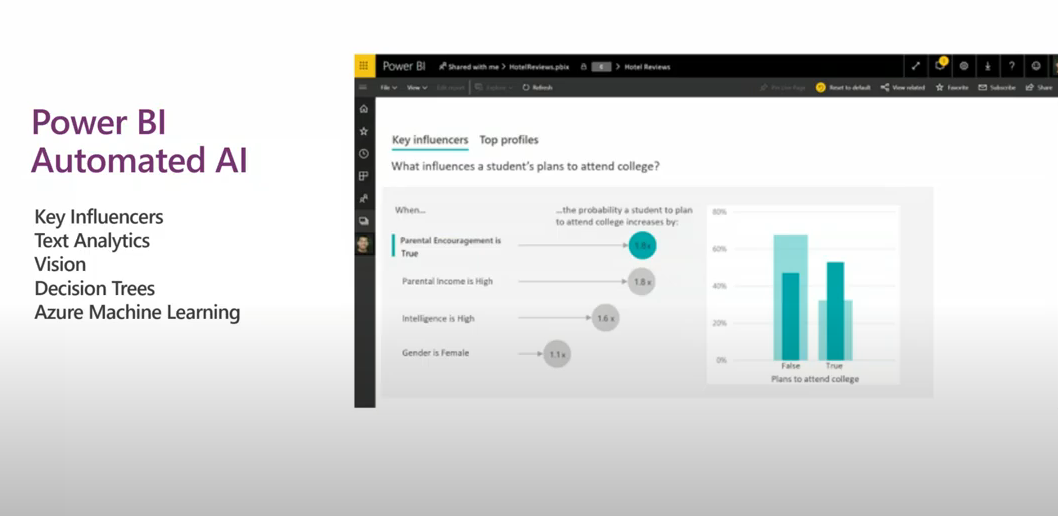
* **My workspace** is the personal workspace for any Power BI customer to work with your own content. Only you have access to your My workspace. You can share dashboards and reports from your My Workspace. If you want to collaborate on dashboards and reports, or create an app, then you want to work in a workspace.
* **Workspaces** are used to collaborate and share content with colleagues. You can add colleagues to your workspaces and collaborate on dashboards, reports, and datasets. With one exception, all workspace members need Power BI Pro licenses.

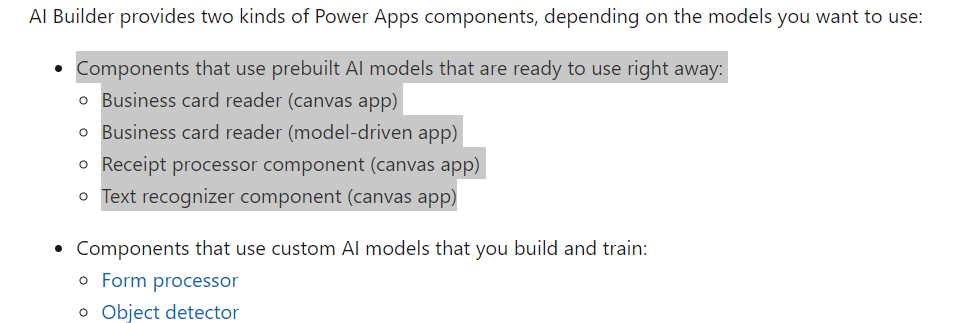
5. A **dataset** is a collection of data that you import or connect to. Power BI lets you connect to and import all sorts of datasets and bring all of it together in one place. Datasets can also source data from dataflows. Datasets added by one workspace member are available to the other workspace members with an admin, member, or contributor role.











## Common business scenarios

