## Understanding Low Code as a Traditional Developer

In canvas-based Power Apps, the low code scripting language used is called Power Fx.

Link to Creating Power Apps canvas app using Power Fx - <https://learn.microsoft.com/en-us/training/modules/understanding-low-code-as-a-traditional-developer/4-create-your-first-app-as-traditional-developer>

## Get started with Power Automate

Cloud flows – runs on cloud with supported connectors to work with different software/services. Three types of cloud flows:

* [Automated](https://learn.microsoft.com/en-us/training/modules/get-started-flows/flow-monitor-incoming-emails) cloud flows
* Instant cloud flows
* [Scheduled](https://learn.microsoft.com/en-us/training/modules/get-started-flows/6-flow-scheduled-flows) cloud flows

Every flow has two main parts: a trigger, and one or more actions.

Triggers - You can think of the trigger as the starting action for the flow. The trigger can be something like a new email arriving in your inbox or a new item being added to a list in Microsoft Lists.

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

Every flow has one trigger and one or more actions.

Share flow - extend the potential of Power Automate to groups of people. All owners of a shared flow can view its history, manage its properties, edit it, add and remove owners, and delete it. To create a shared flow, you must have a paid Power Automate plan. Also, to add more owners to a shared flow or remove owners from it, you must be the creator or an owner.

Important: If you remove an owner whose credentials are used to access Power Automate services, be sure to update the credentials for those connections, so that the flow continues to work correctly.

Embedded Connection - these connections are used in the flow.

Limits and Caveats – each Power App account can have up to:

* 600 flows
* 50 custom connectors
* 20 connections per application programming interface (API) and 100 connections total.

Tips: You can install a gateway only in the default environment.

## Share a cloud flow with Power Automate

Only the person who created a cloud flow can view, edit and run, but can add as co-owner to allow making changes. Co-owner can remove all other co-owners of the flow, but not the creator. Co-owner can delete the flow. A connection that is used by a flow cannot be used again outside of that flow. That connection used the credentials of the user who created it. When collaborating with others on a flow, you might want to designate one user to manage the connections so that they're all in the same context.

Below describes scenarios to determine whether need to share by using co-ownership or not.

|  |  |
| --- | --- |
| **Scenario** | **Options** |
| I want to allow another user to edit my flow. | Co-ownership is the best option. |
| I have a flow that triggers when someone adds a list item to SharePoint or creates a row in a Dataverse table. Do I need to add every user who adds items as a co-owner of my flow? | No, this flow is an automated flow that's triggered by an event. As long as the flow connection user has access to the list or table, the flow runs properly without you needing to add all users as owners or run-only. |
| I have an instant flow, and I want others to run the flow. | Co-ownership isn't required; you should set up the users as run-only. |
| I have a scheduled flow, and I want others to run it on demand in addition to the scheduled times. | Co-ownership is the best option. However, the users have more permissions than only running the flow. |

Tips: Run-only sharing is only available for instant cloud flows. Instant cloud flows are triggered manually from one of the supported applications.

Connection Options – “Run only” option will have two different connection options:

1. Provided by run-only user (default) – when a new user first uses the flow, they're prompted to create a connection (with own credential) for the connectors (e.g. O365) that use that option. User will require to sign-in first.
2. Use this connection – it uses connection created by the flow creator using his/her credentials. This approach allows the flow to take actions that the implementing user might not have access to do. For example, it could post to a Teams channel that the user isn't a member of, because now the flow is using the creator’s credentials, not the one who runs it. No sign-in require using it.

Note: This option can be updated anytime but it will impact all users. You can't have some users with different options.

Send a Copy – sharing this way, users receive their own copy. If the flow uses connectors, the recipient needs to create their own connections; therefore, they need to have relevant permissions. Best approach to sharing when you have a cloud flow that would help others but needs personalization (as a starter).

Important: If you send a copy of a flow to a user from an environment that they don't have access to, they get an error when they try to create the flow. Make sure both parties have access to same environment.

Solution & Sharing - When a Power Automate cloud flow is created in a solution or added to a solution, an extra layer of security is enabled, and it changes how you share a cloud flow. You can add Power Automate cloud flows as a solution component along with other resources, such as Power Apps and Dataverse tables. Then, solutions act as a container for your changes and allow you to transport them from one Microsoft Power Platform environment to another. Additionally, you can export solutions and store them in source control as part of your ALM strategy.

Solution Flow - You can only create solutions in a Microsoft Power Platform environment with Microsoft Dataverse set up. Cloud flows that are associated with a solution are stored in a Dataverse table named Process. The ability of a user, other than the creator of the cloud flow, to run or modify the flow depends on the user's Dataverse security privileges. The newly added user needs to have a security role assigned that gives them at least user-level privileges to the Processes table. Can use built-in [Environment Maker role](https://learn.microsoft.com/en-us/power-platform/admin/database-security#environments-with-a-dataverse-database).

Sharing Solution with Group of Users - You can share a solution flow with a group of users, but it's done by using [Dataverse Teams](https://learn.microsoft.com/en-us/power-platform/admin/manage-teams/) instead of user groups. You can still use Microsoft Entra security groups or office groups, but you need to first associate the group with a Dataverse team. Before you can share flows with the team, you need to associate it with a security role. After you finish, you can share a solution cloud flow with the team, which will give owner access to the cloud flow to all members of the group.

Tips: To share as “Run-only”, the security role that grants them privileges to the Dataverse Processes table must be limited to their own processes. That way, the users can't modify the processes that they didn't create. A good built-in role for run-only use would be the [Basic User role](https://learn.microsoft.com/en-us/power-platform/admin/database-security#environments-with-a-dataverse-database).

## Manage solutions in Power Apps and Power Automate

Categories of Connectors – three types: Standard, Premium and Preview. Custom connectors and connectors for Microsoft Azure Logic Apps fall under the Premium category.

Environment Variables - After creating environment variables in Dataverse for apps, you can consume them by retrieving data from the Environment Variable Definition and Environment Variable Value entities.

You can add environment variables in a solution, and you can add an existing environment variable in a solution or create a new one that automatically becomes part of that solution.

Managed/Unmanaged Solutions - you can't export a managed solution, and when you uninstall a managed solution, it takes all the customizations and extras with it. When you delete an unmanaged solution, it only removes the solution container but keeps all your customizations intact, safely stored in the default solution. The best practice in Application Lifecycle Management (ALM) is to generate managed solutions by exporting an unmanaged solution and treating it like a building block. However, you can only make changes to unmanaged solutions. If you ever need to tweak a managed solution, you'll have to include it in an unmanaged solution. This creates a connection between your unmanaged customizations and the managed solution. When there's a connection, you can't uninstall the managed solution until you remove the link.

Continue lesson here - <https://learn.microsoft.com/en-us/training/modules/manage-solutions-power-automate/2-add-remove-flow>