



Introduction to Power Automate

Sample manual - first two chapters



Wise Owl
Training

TABLE OF CONTENTS (1 of 5)

| 1 | GETTING STARTED | Page |
|-----|--|------|
| 1.1 | What is Power Automate? | 7 |
| 1.2 | Power Automate vs Power Automate Desktop | 8 |
| 1.3 | Licensing Requirements | 9 |

| 2 | WORKING WITH FLOWS | Page |
|-----|--|------|
| 2.1 | Getting Started | 10 |
| | <i>Logging on to Power Automate</i> | 10 |
| | <i>Choosing an Environment</i> | 11 |
| | <i>Listing Flows</i> | 11 |
| 2.2 | Creating Flows | 12 |
| | <i>Types of Flow</i> | 12 |
| | <i>Adding Steps</i> | 13 |
| | <i>Adding Dynamic Content</i> | 14 |
| | <i>Finishing your Flow</i> | 14 |
| 2.3 | Testing and Running Flows | 15 |
| | <i>Testing Flows</i> | 15 |
| | <i>Re-testing Flows</i> | 16 |
| | <i>Running Flows</i> | 16 |
| 2.4 | Viewing Flows | 17 |
| 2.5 | Improving Flow Readability | 18 |
| | <i>Renaming Actions</i> | 18 |
| | <i>Adding Notes</i> | 19 |
| | <i>Adding Comments</i> | 20 |
| 2.6 | Working with Flows | 21 |
| | <i>Checking Flows</i> | 21 |
| | <i>Peeking at Flow Syntax</i> | 21 |
| | <i>Disabling Flows (but not Actions)</i> | 22 |
| | <i>Copying Actions</i> | 22 |
| | <i>Copying Flows</i> | 23 |
| | <i>Deleting Flows</i> | 23 |

| 3 | TRIGGERS AND TEMPLATES | Page |
|-----|--|------|
| 3.1 | Our Example Automated (Triggered) Flow | 24 |
| 3.2 | Creating Automated Flows from Scratch | 25 |
| 3.3 | Creating Automated Flows using Templates | 26 |
| | <i>Finding the Right Template</i> | 26 |
| | <i>Tweaking your Template</i> | 27 |

| 4 | TESTING AND DEBUGGING | Page |
|-----|---|------|
| 4.1 | Testing Flows | 28 |
| 4.2 | Viewing Outputs from Actions | 29 |
| | <i>Viewing Outputs</i> | 29 |
| | <i>Capturing Outputs - the Compose Action</i> | 30 |
| | <i>Seeing Inputs and Outputs</i> | 30 |
| | <i>Raw Inputs and Outputs</i> | 31 |
| 4.3 | Interpreting Outputs (JSON) | 32 |
| 4.4 | Writing to Text Files | 33 |
| 4.5 | Dealing with Flow Failures | 34 |

| 5 | PROGRAMMING CONCEPTS | Page |
|-----|--|------|
| 5.1 | Overview | 35 |
| | <i>Our Example</i> | 35 |
| 5.2 | Simple Variables | 36 |
| | <i>Initialising Variables</i> | 36 |
| | <i>The Types of Variables</i> | 37 |
| | <i>Variable Actions</i> | 37 |
| | <i>Debugging Variables</i> | 38 |
| 5.3 | Array Variables | 39 |
| | <i>Getting our List of Files - OneDrive</i> | 39 |
| | <i>Getting our List of Files – SharePoint</i> | 40 |
| 5.4 | Using Scope to Group Actions | 41 |
| | <i>Adding a Scope Action</i> | 41 |
| | <i>Tidying Things Up</i> | 42 |
| 5.5 | Parallel Branches | 43 |
| | <i>Creating Parallel Branches</i> | 43 |
| | <i>Adding Steps to Parallel Branches</i> | 44 |
| 5.6 | Basic Expressions | 45 |
| | <i>Basic Expressions and Referencing Variables</i> | 45 |
| | <i>Commenting Expressions</i> | 46 |
| | <i>Enabling Experimental Features</i> | 46 |
| | <i>Editing Expressions using Experimental Features</i> | 46 |
| 5.7 | Single Conditions - If | 47 |
| | <i>Adding Conditions</i> | 47 |
| | <i>Testing Conditions</i> | 48 |
| | <i>Why you Need to Terminate Flows Early</i> | 48 |
| | <i>The Terminate Action</i> | 49 |
| 5.8 | Multiple Conditions – Switch | 50 |
| 5.9 | Do Until Loops | 51 |
| | <i>Initialising Variables</i> | 51 |
| | <i>Adding a Do Until Loop</i> | 52 |
| | <i>Setting Limits to Do Until Loops</i> | 52 |

TABLE OF CONTENTS (2 of 5)

| 6 | WORKING WITH ARRAYS | Page |
|-----|---|------|
| 6.1 | Creating Arrays | 53 |
| | <i>Setting Array Variables Manually</i> | 53 |
| | <i>Creating Arrays with Named Properties</i> | 54 |
| | <i>Creating Arrays using Expressions</i> | 54 |
| | <i>Nested Arrays</i> | 55 |
| 6.2 | Referring to Items within an Array | 56 |
| | <i>An Example using a List of Files</i> | 56 |
| | <i>Getting Values by their Key</i> | 57 |
| 6.3 | Mapping or Reshaping Arrays using Select | 58 |
| | <i>Collapsing Arrays to One Value</i> | 58 |
| | <i>Returning Multiple Columns with Renaming</i> | 59 |
| 6.4 | Intersection and Union | 60 |
| | <i>Overview of Final Answer</i> | 60 |
| | <i>The Union and Intersect Expressions</i> | 61 |

| 7 | GETTING DATA | Page |
|-----|---|------|
| 7.1 | Sources Included in this Chapter | 62 |
| 7.2 | Getting Data from Excel | 63 |
| | <i>The Need for a Table</i> | 63 |
| | <i>Creating a Table</i> | 63 |
| | <i>Getting Data from an Excel Table</i> | 64 |
| | <i>Bonus: Listing Worksheets and Tables</i> | 65 |
| 7.3 | Getting Data from a SharePoint List | 66 |
| | <i>Getting SharePoint List Rows</i> | 66 |
| | <i>Showing the Correct Column Names</i> | 67 |
| 7.4 | Listing Files/Folders from OneDrive | 68 |
| 7.5 | Listing Emails from Outlook | 69 |

| 8 | ODATA | Page |
|-----|---|------|
| 8.1 | What is OData? | 70 |
| 8.2 | The Need for OData | 71 |
| 8.3 | Using OData | 72 |
| | <i>Applying OData when Getting Data</i> | 72 |
| | <i>Sorting using OData</i> | 72 |
| | <i>Filtering using OData</i> | 73 |
| | <i>Combining Filters</i> | 74 |
| | <i>Testing for Null Values</i> | 74 |
| | <i>Filtering Dates</i> | 74 |
| 8.4 | Issues with Different Data Sources | 75 |
| | <i>Excel: Spaces in Column Names</i> | 75 |
| | <i>SharePoint: Lookup Columns</i> | 75 |
| 8.5 | Dynamic Filters | 76 |
| 8.6 | OData Filters using Experimental Features | 77 |
| 8.7 | Avoiding OData – the FilterArray Action | 78 |
| | <i>Suppressing the GetItems Warning</i> | 78 |

| 9 | HTML TABLES | Page |
|-----|------------------------|------|
| 9.1 | Overview | 79 |
| 9.2 | Basic HTML Tables | 80 |
| 9.3 | Custom HTML Tables | 81 |
| 9.4 | Formatting HTML Tables | 82 |

| 10 | APPLY TO EACH | Page |
|------|---|------|
| 10.1 | Our Example | 83 |
| 10.2 | Adding Apply to Each Manually | 84 |
| | <i>Getting an Array to Loop Over</i> | 84 |
| | <i>Creating an Apply to Each Action</i> | 84 |
| | <i>The Current Item</i> | 85 |
| | <i>Testing Apply to Each Loops</i> | 85 |
| 10.3 | Adding Apply to Each Automatically | 86 |
| 10.4 | Referring to the Current Item's Contents | 87 |
| | <i>Dynamic Content (Visible Fields)</i> | 87 |
| | <i>Copying Visible Fields to Expressions</i> | 87 |
| | <i>Making Pasted Expressions Valid</i> | 88 |
| | <i>Visible Fields and their Underlying Expressions Compared</i> | 88 |
| | <i>Using the Simpler Item() Function</i> | 89 |
| 10.5 | Applying Conditions within Loops | 90 |
| | <i>The Outline of our Flow</i> | 90 |
| | <i>Our Condition</i> | 91 |

TABLE OF CONTENTS (3 of 5)

| 11 | EXPRESSIONS | Page |
|------|---|------|
| 11.1 | Overview | 92 |
| 11.2 | Text Functions | 93 |
| | <i>The Main Text Functions</i> | 93 |
| | <i>Example Flow using Text Functions - Introduction</i> | 94 |
| | <i>Example Flow - Apply to Each Action</i> | 95 |
| | <i>Example Flow – Showing the Results</i> | 95 |
| 11.3 | Number Functions | 96 |
| | <i>Basic Mathematical Operator Functions</i> | 96 |
| | <i>Other Mathematical Functions</i> | 96 |
| | <i>Conversion Functions</i> | 96 |
| 11.4 | Date Functions | 97 |
| | <i>Date Formats and Timestamps</i> | 97 |
| | <i>Getting Dates</i> | 97 |
| | <i>Manipulating Dates</i> | 98 |
| | <i>Getting Parts of Dates</i> | 98 |
| 11.5 | Collection/Array Functions | 99 |
| | <i>Inspection Functions</i> | 99 |
| | <i>Joining and Splitting</i> | 99 |
| | <i>Picking Items</i> | 100 |
| | <i>Ordering Arrays</i> | 100 |
| | <i>Set Operations</i> | 100 |
| 11.6 | Comparison Functions | 101 |
| | <i>Conditional IF Function</i> | 101 |
| | <i>Testing for Equality or Inequality</i> | 101 |
| | <i>Combining Conditions</i> | 101 |

| 12 | INPUTS AND TRIGGERS | Page |
|------|--|------|
| 12.1 | Overview | 102 |
| | <i>Our Example</i> | 102 |
| 12.2 | Creating a Flow with Inputs | 103 |
| | <i>Dropdown Inputs</i> | 103 |
| | <i>How Power Automate Refers to Inputs (TriggerBody)</i> | 104 |

| 13 | ERROR HANDLING | Page |
|------|---|------|
| 13.1 | Try / Catch / Finally Blocks | 105 |
| 13.2 | Configuring Run After Actions | 106 |
| 13.3 | Implementing Error Handling – a Case Study | 107 |
| | <i>Our Example Error</i> | 107 |
| | <i>Summary of Proposed Flow</i> | 107 |
| | <i>Step 1 – Getting the Results</i> | 108 |
| | <i>Step 2 – Filtering the Results Array</i> | 109 |
| | <i>Step 3 – Picking out the Columns of Interest</i> | 109 |
| | <i>Step 4 – Presenting an HTML Table</i> | 109 |
| | <i>Step 5 – Sending a Notification Email</i> | 110 |
| | <i>Step 6 – Creating a Link back to our Flow</i> | 110 |
| 13.4 | Warning: Flows Run Successfully with Error-Handling | 111 |

| 14 | APPROVALS | Page |
|------|---|------|
| 14.1 | Our Example | 112 |
| 14.2 | Building a Flow Diagram | 113 |
| 14.3 | How the Approval Process Works | 114 |
| 14.4 | Building the Approval Flow | 115 |
| | <i>Step 1 – Creating the Trigger</i> | 115 |
| | <i>Step 2 – Request Approval</i> | 115 |
| | <i>Step 3 – Adding a Condition</i> | 116 |
| | <i>Step 4 – Processing Rejections</i> | 116 |
| | <i>Step 5 – Processing Approvals (Part 1)</i> | 117 |
| | <i>Step 6 – Processing Approvals (Part 2)</i> | 118 |
| 14.5 | Managing Approvals | 119 |
| 14.6 | Custom Responses | 120 |
| | <i>Choosing to Set Custom Responses</i> | 120 |
| | <i>Testing the Different Responses</i> | 121 |

TABLE OF CONTENTS (4 of 5)

| 15 | FORMS | Page |
|------|--|------|
| 15.1 | Overview of Forms | 122 |
| | <i>Creating Forms</i> | 123 |
| 15.2 | Adding Questions | 124 |
| | <i>The Types of Question</i> | 124 |
| | <i>Sections</i> | 125 |
| | <i>Images and Videos</i> | 126 |
| | <i>Branching</i> | 126 |
| 15.3 | Testing and Distributing your Form | 127 |
| | <i>Previewing a Form</i> | 127 |
| | <i>Changing Form Settings</i> | 127 |
| | <i>Saving your Form</i> | 128 |
| | <i>Distributing your Form via a Web Link</i> | 128 |
| | <i>Generating QR Codes and Embed Codes</i> | 128 |
| 15.4 | Viewing Responses | 129 |
| 15.5 | Other Things you can do with Forms | 130 |
| | <i>Using Other Apps such as Excel</i> | 130 |
| | <i>Creating and Applying Themes</i> | 130 |
| | <i>Letting Users Upload Files</i> | 131 |
| | <i>Quizzes</i> | 132 |
| 15.6 | Responding to Form Events in a Flow | 133 |

| 16 | SHAREPOINT LISTS | Page |
|------|---|------|
| 16.1 | About SharePoint Lists | 134 |
| 16.2 | Creating a SharePoint Site | 135 |
| 16.3 | Working with SharePoint Lists | 136 |
| | <i>Creating a New List</i> | 136 |
| | <i>Renaming the Title Column</i> | 137 |
| | <i>Adding/Editing List Data</i> | 137 |
| | <i>Creating Choice and Lookup Columns</i> | 138 |
| 16.4 | Internal SharePoint List Column Names | 139 |
| | <i>Finding Column Names in Power Automate</i> | 139 |
| | <i>Finding Column Names in SharePoint</i> | 139 |
| 16.5 | Referencing Choice and Lookup Columns | 140 |

| 17 | POWER BI DESKTOP FLOWS | Page |
|------|--|------|
| 17.1 | Our Example | 141 |
| | <i>Some Limitations of Running Flows from Power BI Desktop</i> | 141 |
| 17.2 | Creating a Power Automate Button | 142 |
| | <i>Step 1 – Creating the Button</i> | 142 |
| | <i>Step 2 – Assigning Data Fields</i> | 142 |
| | <i>Step 3 – Start Editing your Flow</i> | 143 |
| | <i>Step 4 – Add Actions (and Data)</i> | 143 |
| | <i>Step 5 – Saving and Applying your Flow</i> | 144 |
| | <i>Step 6 – Formatting your Button</i> | 144 |
| | <i>Step 7 – Test your Button</i> | 144 |
| 17.3 | Editing Power BI Flows in Power Automate | 145 |

| 18 | POWER APPS FLOWS | Page |
|------|---|------|
| 18.1 | Our Example | 146 |
| 18.2 | Designing an Information Flow | 147 |
| 18.3 | Creating the Flow | 148 |
| | <i>Step 1 – Creating the Flow with a Trigger</i> | 148 |
| | <i>Step 2 – Creating Parameters with Good Names</i> | 148 |
| | <i>Step 3 – Getting the Number of Excel Films</i> | 149 |
| | <i>Step 4 – Adding the Film to Excel</i> | 149 |
| | <i>Step 5 – Returning Values to Power Apps</i> | 150 |
| 18.4 | Creating the App | 151 |
| | <i>Step 1 - Creating the App Framework</i> | 151 |
| | <i>Step 2 – Adding the Flow</i> | 151 |
| | <i>Step 3 – Calling the Flow</i> | 152 |
| 18.5 | Possible Problems | 153 |

| 19 | TEAMS | Page |
|------|---|------|
| 19.1 | Teams Triggers and Actions | 154 |
| 19.2 | Our Example – Listening for Keywords | 155 |
| 19.3 | Creating our Example Flow | 156 |
| | <i>Step 1 - Setting the Trigger</i> | 156 |
| | <i>Step 2 - Starting to Post our Message</i> | 156 |
| | <i>Step 3 - Getting the Keyword Message's Details</i> | 157 |
| | <i>Step 4 – Posting the Notification Message</i> | 158 |
| | <i>Step 5 – Testing your Flow</i> | 158 |

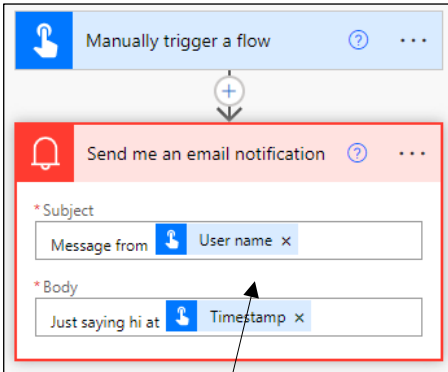
TABLE OF CONTENTS (5 of 5)

| 20 | PLANNER | Page |
|------|---|------|
| 20.1 | Triggers and Actions | 159 |
| 20.2 | Simple Example – Listing Tasks | 160 |
| 20.3 | Harder Example – Getting Assigned Users | 161 |
| | <i>Step 1 – Prepare the Way</i> | 161 |
| | <i>Step 2 – Create the Nested Loops</i> | 162 |
| | <i>Step 3 – Configure the Inner Loop over Users</i> | 162 |
| | <i>Step 4 – Add this Task and List of Users</i> | 163 |
| | <i>Step 5 – Show the Results</i> | 163 |

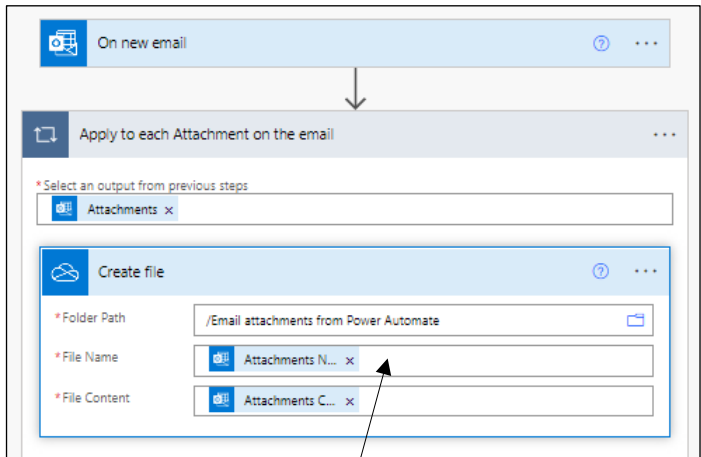
CHAPTER 1 - GETTING STARTED

1.1 What is Power Automate?

Power Automate (PA) used to be called *Microsoft Flow* (Microsoft renamed it in 2019). PA allows you to create flows (series of saved instructions) online:



A simple manual flow to send an email greeting to yourself.



A flow using a trigger (when you receive an email, PA will save its attachments to a specified folder in OneDrive).

Flows can connect to literally hundreds of cloud-based services – here is a small sample:



Just a few of the over 370 connectors that you can use to link to online services.



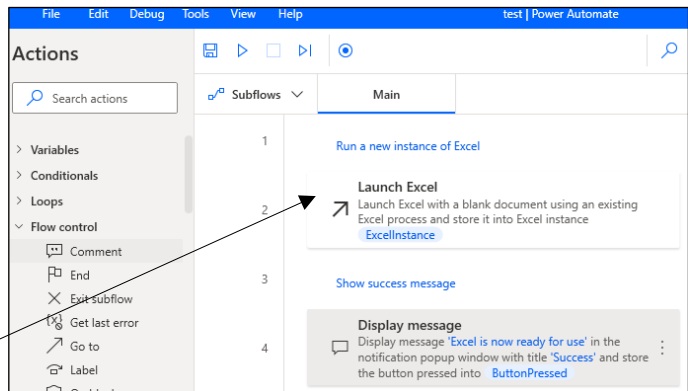
If you commit to using Power Automate, you are trusting your data, business processes and probably career to Microsoft and the Internet: if your web connection should happen to go down or Microsoft's servers fail, there won't be much that you can do!

1.2 Power Automate vs Power Automate Desktop

Microsoft (confusingly, if typically) market two software products whose names include **Power Automate**:


| Software | Runs in | What it does | Typical use cases |
|-------------------------------|--------------|---|--|
| <i>Power Automate</i> | The cloud | Automates a series (or <i>flow</i>) of instructions to run on Microsoft's servers, under your account. | Collating the results of an online survey, emailing notifications when someone posts a comment in Teams, capturing social media updates in an online database. |
| <i>Power Automate Desktop</i> | Your desktop | Automates a series (or <i>flow</i>) of instructions to manipulate data on your own computer. | Collating data in Excel workbooks, logging attachments sent in emails, submitting website forms, running backup routines. |

Here's what a *Power Automate Desktop* flow looks like:



This flow starts an Excel application and then announces that it's ready for use.

Power Automate and Power Automate Desktop were written by different companies (Microsoft bought *SoftMotive*, authors of Power Automate Desktop, in May 2020) and are different products.



Confusingly, while Power Automate Desktop runs on your desktop it too stores its flows in the cloud (in your online Microsoft account). You can run Power Automate Desktop flows from Power Automate, but the converse isn't true.

1.3 Licensing Requirements

Most users of Power Automate will begin with an Office 365 licence. This allows you to do the following:

| Capability | With an Office 365 licence you can ... |
|--|--|
| <i>Run flows</i> | ... create and run automated, scheduled and button flows (that is, most of the flows you are likely to want to run). |
| <i>Use standard connectors</i> | Use any of the connectors described as <i>Standard</i> by Microsoft. These include Azure Active Directory, Dropbox, Excel online, GitHub, Gmail, Google apps, LinkedIn, Microsoft Forms, Office 365, Outlook, Pinterest, Power Apps and Power BI, Slack, SharePoint, Teams, WordPress and YouTube. |
| <i>2,000 platform requests per day</i> | During an initial transition period (lasting 6 months or more), you will be allowed up to 10,000 requests per day. The limit is applied on a rolling 24-hour basis. If you exceed your limit you will need to go to another plan (see below). |

Here are some of the main things NOT included in this licence:

| What | Notes |
|-----------------------------------|---|
| <i>Premium connectors</i> | To connect to connectors designated as premium by Microsoft, you will need any standalone plan (that is, anything except an Office 365 licence). Premium connectors include ArcGIS, most Azure services, most Dynamics service (for which you can buy a specialist licence), Map Pro, Microsoft Dataverse, MySQL, Oracle, Ordnance Survey Places, Salesforce, SAP, Stripe, Word Online and Zoom Meetings. |
| <i>Business process flows</i> | For example, you can create a business process flow to ensure that everyone in an organisation follows the same steps every time they handle a customer request. This courseware doesn't consider business process flows, and neither (Wise Owl suspect) will most customers. |
| <i>Custom connectors</i> | If you can't find a connector for your service, you can create your own with a custom API wrapper. |
| <i>On premises gateways</i> | These let you build a bridge between a Microsoft cloud service such as Power Automate and a data source in your organisation (such as a SQL Server database). |
| <i>Robotic process automation</i> | This allows you to build a flow to mimic the keyboard actions and mouse movements of a human user (for example, to fill in a website form), usually using Power Automate Desktop. |
| <i>AI builder capacity</i> | You can use AI to build and train AI models to make predictions based on your data. The AI builder has its own licensing rules! |

To use premium or custom connectors or on premises gateways you will probably need to buy one or more separate per-flow or per-user Power Automate licence plans.



If there's not enough detail in this section, you can always search for the phrase "Microsoft Power Apps, Microsoft Power Automate and Microsoft Power Virtual Agents Licensing Guide" and download the 34-page PDF this leads to.


CHAPTER 2 - WORKING WITH FLOWS

2.1 Getting Started

This section shows how to access Power Automate, choose the environment in which to work and see flows you've created.

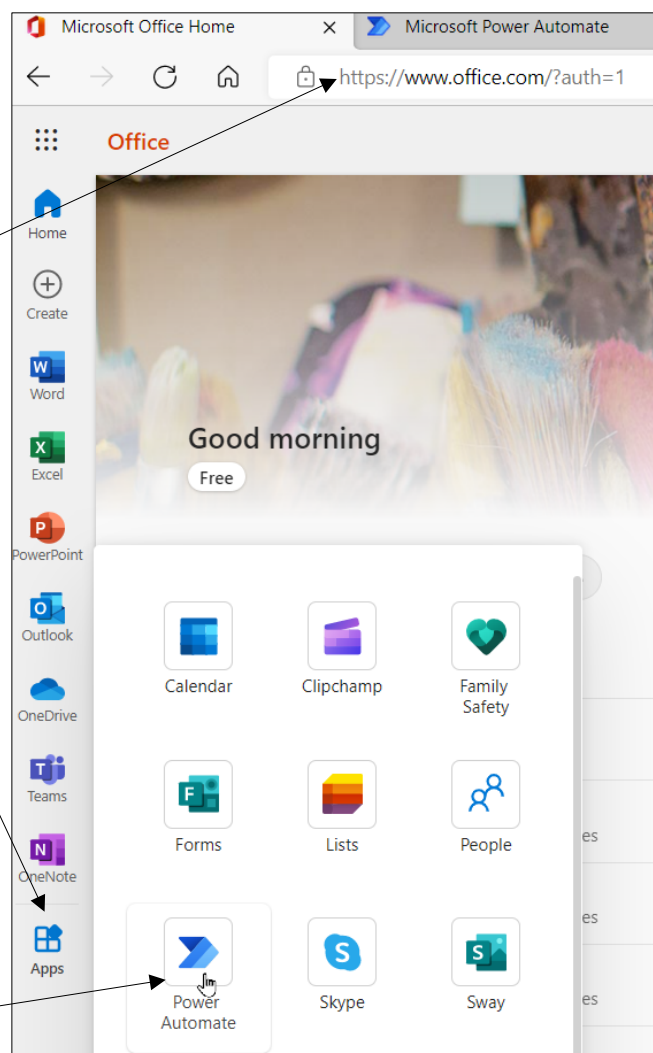
Logging on to Power Automate

If you are using an Office 365 licence to access Power Automate, one reliable way to log in is via the Office 365 home page:

a) Type in the URL **office.com** (it will morph into this longer address when you press ).

b) Choose to run an app (Power Automate isn't listed on the left-hand sidebar).

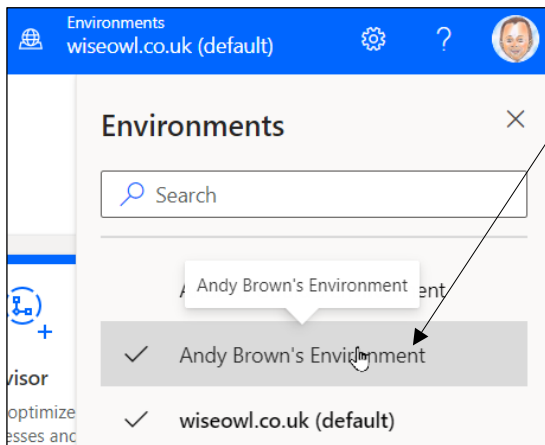
c) Choose to run Power Automate: it will open in a new window.



Alternatively, just visiting the website **powerautomate.com** also seems to work, although again after the website loads a longer URL may appear in your browser's address bar.

Choosing an Environment

You can think of *environments* as different folders in your little bit of the Microsoft cloud. Here's how to change your environment:



Often your organisation will have one environment for development and one production environment. Wise Owl have a shared company environment, plus a private one for each trainer.

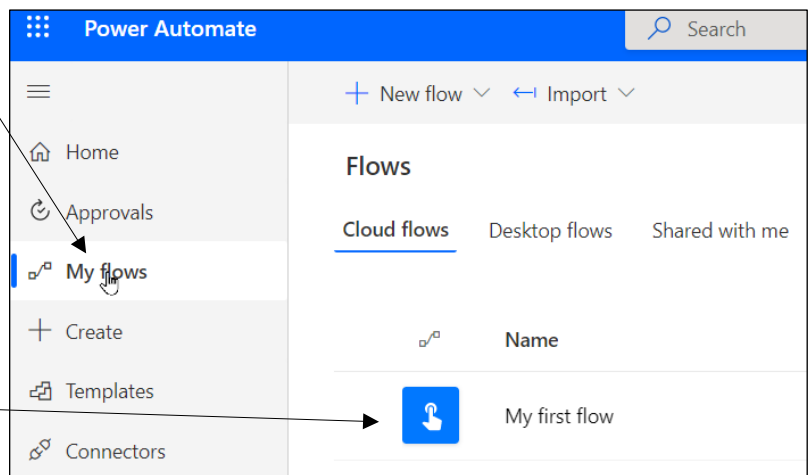


Each environment sits on its own tenant (think of this as a space you've leased from Microsoft on some central server). To move a flow that you've created from environment (tenant) to another is a non-trivial process: you have to export it from the first environment and then import it into the second.

Listing Flows

Once you've chosen your environment, you can see the flows it contains:

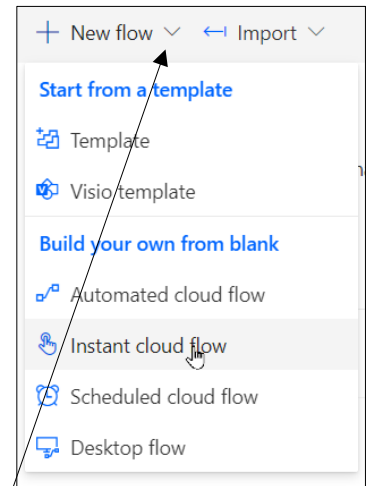
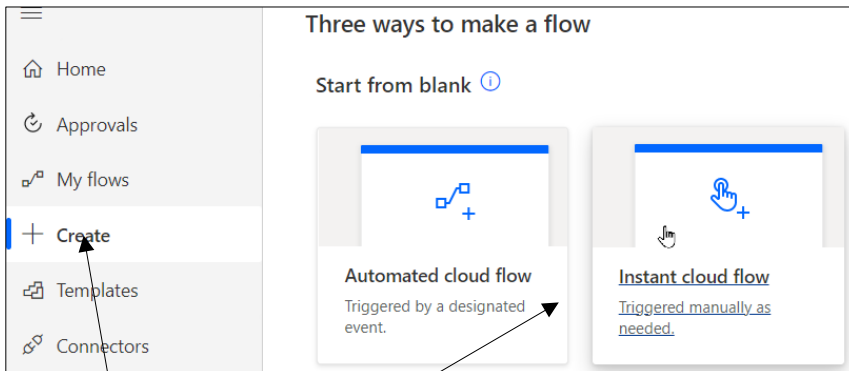
Click on this menu option on the left-hand side ...



... to see the cloud flows or desktop flows that you've created (or the flows colleagues have shared with you).

2.2 Creating Flows

Follow these steps to create a new flow:



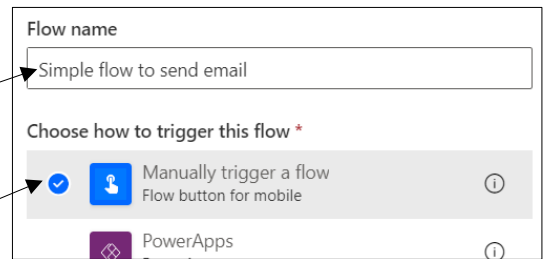
a) Click on this menu to create a new flow.

b) Choose the type of flow to create (see notes below for what each means).

Alternatively, click on the arrow next to the **New flow** button when listing your flows, and choose which one you want to create.

c) Give your flow a name. Avoid punctuation: although you can create flows whose names include characters like `[]`, they will crash when you try to run them!

d) For the example in this chapter the only way to run our flow will be manually.



Types of Flow

PA lists 5 types of flow, but the first two are the ones which most people will choose:

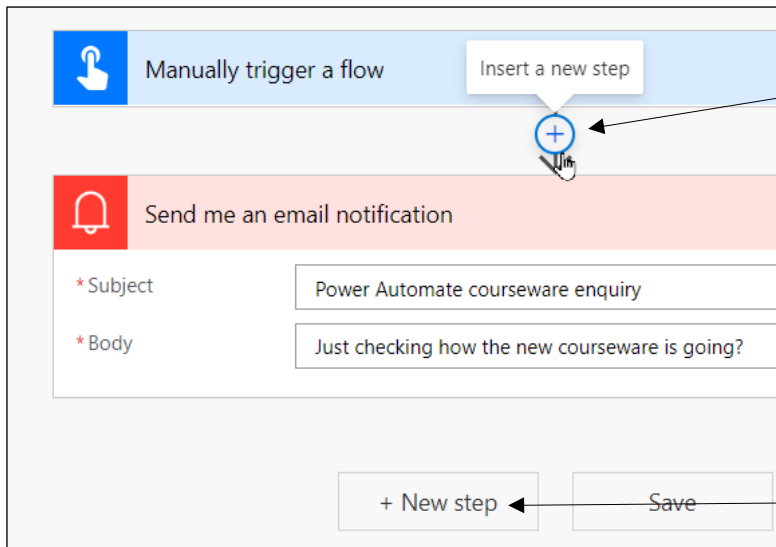
| Type of flow | Description |
|-----------------------------|---|
| <i>Automated cloud flow</i> | Any flow which begins when something happens. Triggers can include comments being added to Teams discussions, emails being received or items being added to SharePoint lists (to choose just 3 examples). |
| <i>Instant cloud flow</i> | When you're testing a flow (or learning PA) this is the best flow type to choose, since you can test it in isolation without having to trigger an event first. |
| <i>Scheduled cloud flow</i> | When you create a scheduled flow you'll be asked to say when it should run (every week at 10am, for example, or every 10 minutes throughout the day). |
| <i>Desktop flow</i> | This will launch Power Automate Desktop, the subject of a different Wise Owl training course (and manual!). |
| <i>Process advisor</i> | Business process flows are a separate topic which won't be of interest to most PA users, and are not covered in this courseware. |



*The easiest thing to do, Wise Owl have found, is to create either an automated or instant cloud flow, but then click on the **Skip** button to miss out the stage where you choose the initial trigger. This will bring up a blank flow, and you can then choose which trigger you want to choose more easily.*

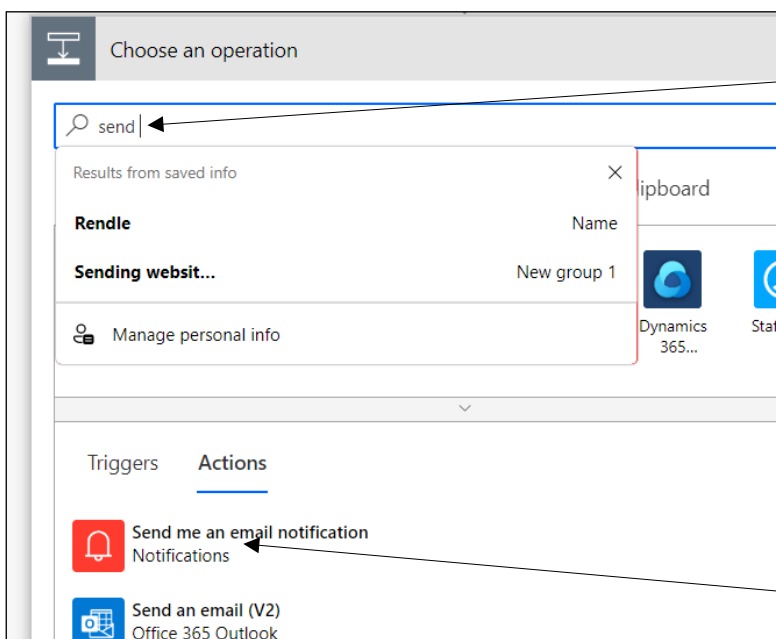
Adding Steps

You can add actions to your flow as follows:



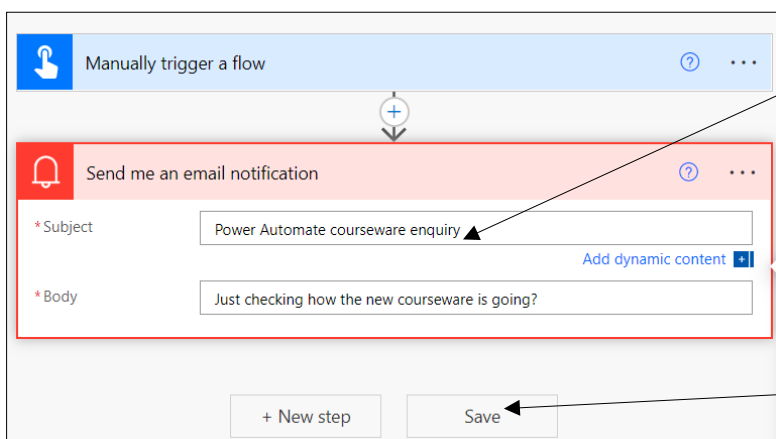
a) To add a new step either let your mouse button linger over the arrow joining two existing steps together, and click on the **+** sign which appears as shown here ...

b) ... or click on this button to add a new step to the bottom of your flow.



c) Type in what you want to do in part (this is by far the easiest way to choose what action you want to add).

d) Choose the action which best describes what you're trying to do.



e) You can now configure the action (here to say what the subject and body of the email you're sending will be).

f) Click on the **Save** button to avoid losing the change that you've just made.

Adding Dynamic Content

When adding a step, PA will always offer up any information that it knows about for inclusion – this is called *dynamic content*. To add this to any action:

a) Click where you want to insert something.

b) Click on what you want to add (here it's the current date). PA will add this in at your cursor position:

Finishing your Flow

When you've finished working with a flow, here are two ways to leave it:

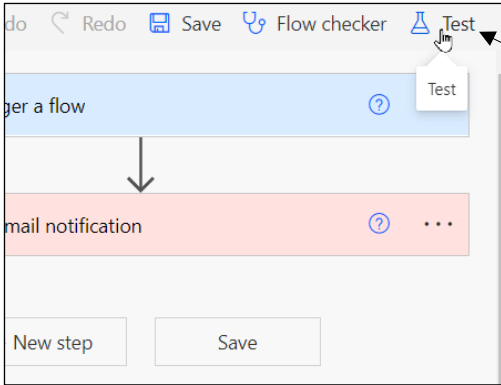
Click on this button to return to the previous screen ...

... or click on **My flows** to list out the flows that you have created (including the one you've just added and saved).

2.3 Testing and Running Flows

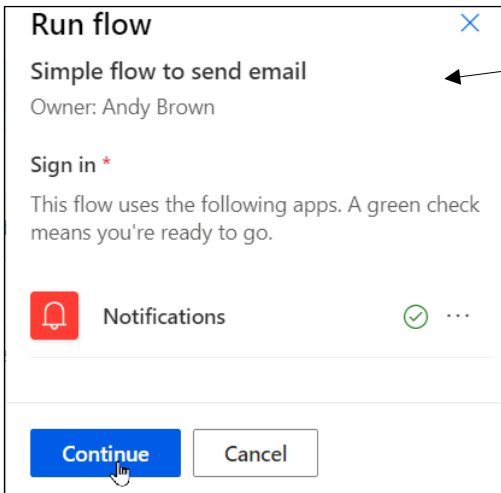
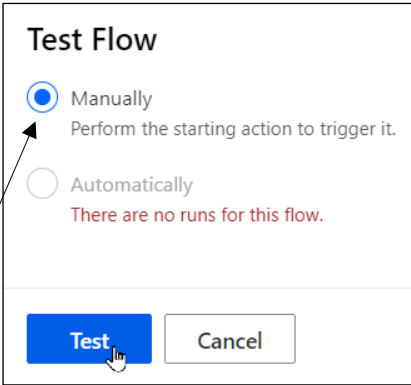
Testing Flows

One way to test a flow is to do whatever triggers it (whether this be sending an email, adding a comment or whatever). However, you can also test and run instant flows as follows:



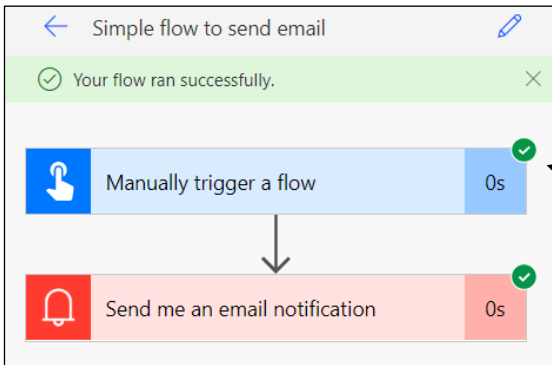
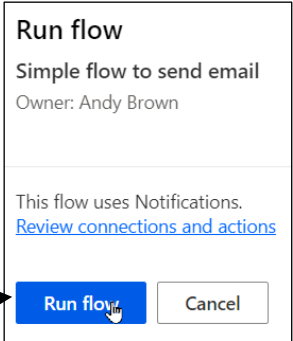
a) Click on this button to test the flow you're editing.

b) Choose to test your flow (see overleaf under re-testing flows for what **Automatically** means).

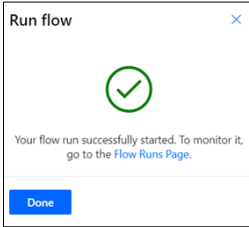


c) If your flow is using any connections to other services, you'll be asked to check that these are set up OK (here the green tick means we're good to go).

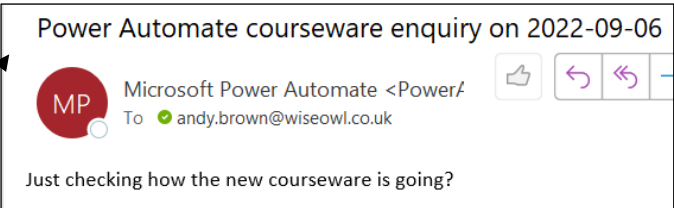
d) You can now choose to run this flow.



e) When you choose to confirm that you want to run the flow as below, you'll see green ticks appearing next to each completed action:

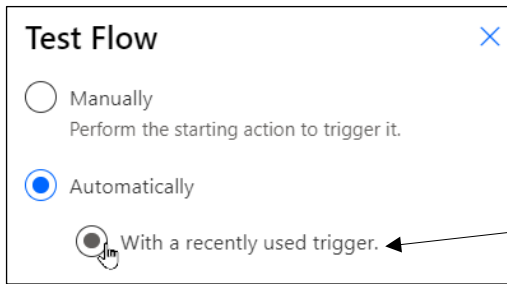


f) The result of running this flow: after a short while this email will appear in your inbox, together with the delivery date we added as dynamic content.



Re-testing Flows

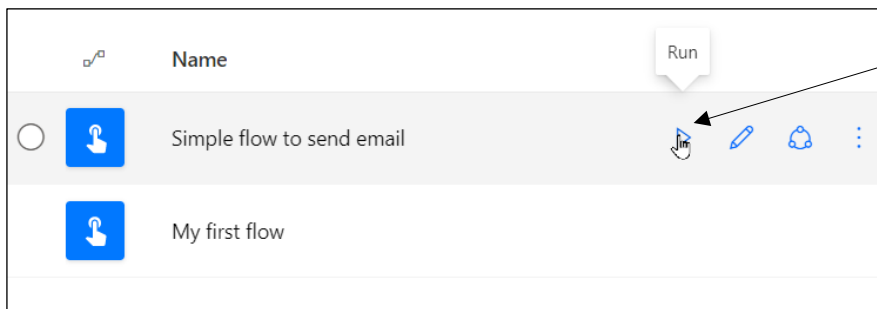
Once you've run a flow once, there's a different way to test it:



When you click on the button to test your flow you'll see a list of all the previous occasions on which it's run. You can choose any of these to re-run the flow using the same starting conditions.

Running Flows

Another way to run a flow is after you've saved it:

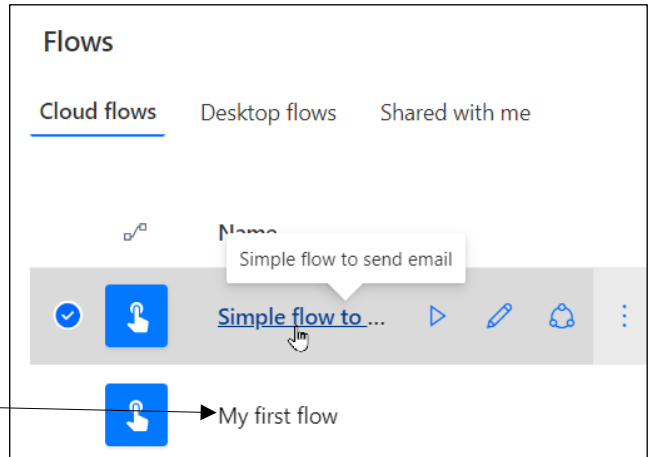


Click on this play button to run a flow from your list.

2.4 Viewing Flows

Click on the name of a flow to view it:

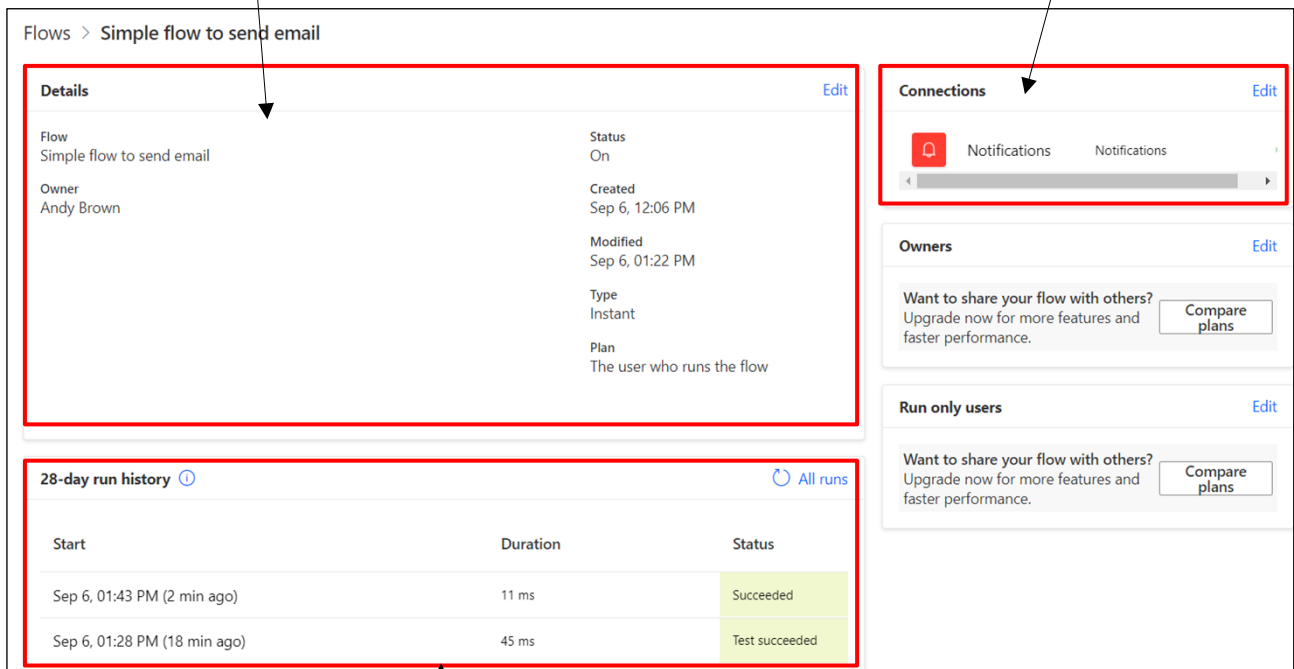
You can click on the pencil symbol to go directly to editing the flow, but if you click on the name of the flow you'll see the results below.



Here's what you then see:

Details about this flow (who created it and when, and what type of flow it is).

A list of the connections used by this flow.



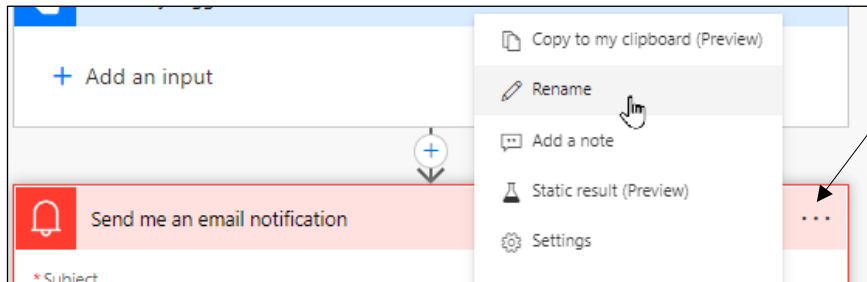
A history of when the flow has been run (twice in the last 20 minutes for this flow, both times being successful).

2.5 Improving Flow Readability

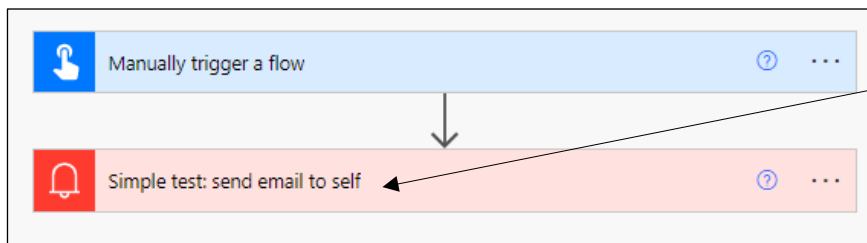
There are three ways to make your flows easier to read: renaming actions, adding *notes* and adding *comments*.

Renaming Actions

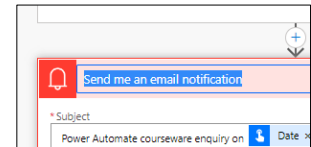
Renaming actions makes it easier to see what your flow is doing:



a) Click on these three dots in an action's title bar and choose the **Rename** menu option, as shown here.



b) Overtyping what's in the action's name:



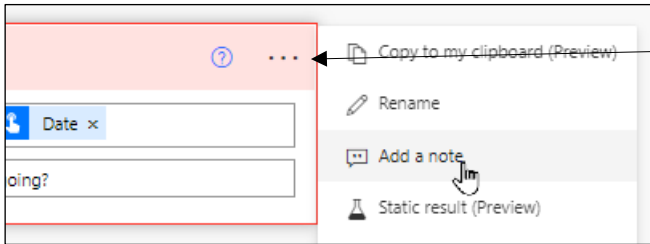
to get a more meaningful name such as this.



The downside of this? Many expressions that you create in PA refer to actions by name. The longer your action names are, the longer (and more cumbersome) your expression formulae will be!

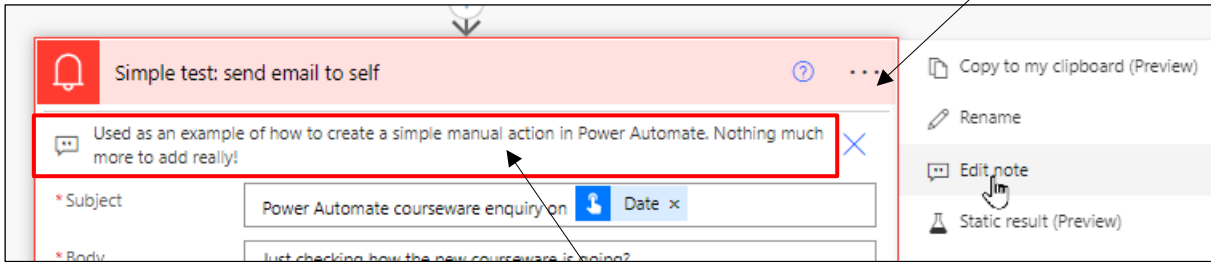
Adding Notes

Notes explain what actions are doing:



Click on the 3 dots in an action's title bar to add a note to it, choosing this menu option.

Once you've created a note, you can click on the same 3 dots and choose to edit it, as shown here.



There doesn't seem to be any way to add a carriage return within a note, but the text wraps round when what you type in takes up more than one line (as here).

Adding Comments

Different viewers of your flows can add *comments* to explain (or ask about!) their use:

You can click on this button to show any comments for a flow.

A number like this against an action shows that it has a comment against it.

When you click on an action, PA highlights (it seems) the first comment for it.

To add a comment to a flow:

a) Click on this button to work with comments.

b) Click on this button to create a new comment.

c) Type in the text of your comment and either click on this symbol to post it or press **Ctrl** + **↵**.



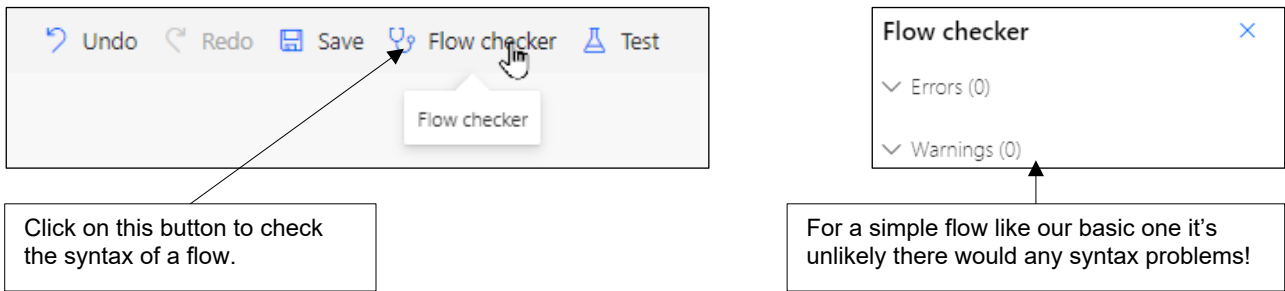
Because comments aren't permanently visible, you should perhaps use them to discuss the contents and purpose of a flow rather than to document it.

2.6 Working with Flows

This section shows some other ways in which you can work with flows.

Checking Flows

The **Flow checker** button at the top right of PA allows you to see if a flow makes syntactical sense:



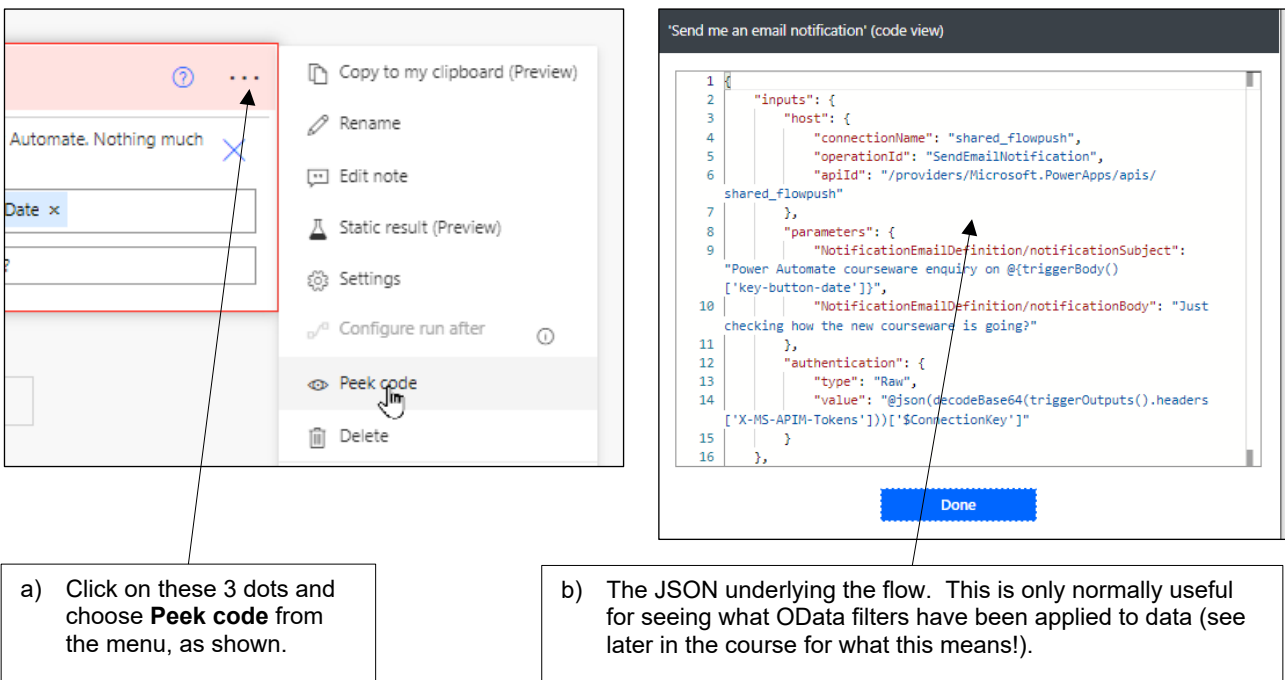
The image shows the Power Automate interface with the 'Flow checker' button highlighted in the top right corner. A callout box points to this button with the text: "Click on this button to check the syntax of a flow." To the right, the 'Flow checker' dialog box is open, showing 'Errors (0)' and 'Warnings (0)'. A callout box points to the dialog with the text: "For a simple flow like our basic one it's unlikely there would any syntax problems!"



When you run a flow, PA automatically invokes the flow checker anyway, so checking flows manually probably isn't that important.

Peeking at Flow Syntax

If you want to see what an action in your flow is really doing, *peek* at its code:



The image shows the 'Peek code' menu option in the Power Automate interface. A callout box points to the three dots menu with the text: "a) Click on these 3 dots and choose **Peek code** from the menu, as shown." To the right, the 'Send me an email notification' (code view) is shown, displaying the JSON code for the action. A callout box points to the JSON code with the text: "b) The JSON underlying the flow. This is only normally useful for seeing what OData filters have been applied to data (see later in the course for what this means!)."

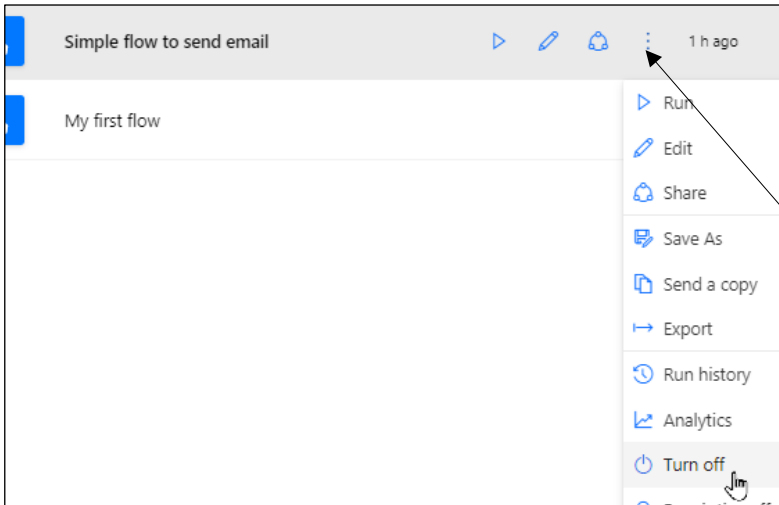
```

1 {
2   "inputs": {
3     "host": {
4       "connectionName": "shared_flowpush",
5       "operationId": "SendEmailNotification",
6       "apiId": "/providers/Microsoft.PowerApps/apis/
shared_flowpush"
7     },
8     "parameters": {
9       "NotificationEmailDefinition/notificationSubject":
"Power Automate courseware enquiry on @{triggerBody()
['key-button-date']}",
10      "NotificationEmailDefinition/notificationBody": "Just
checking how the new courseware is going?"
11    },
12    "authentication": {
13      "type": "Raw",
14      "value": "@json(decodeBase64(triggerOutputs().headers
['X-MS-APIM-Tokens']))['$ConnectionKey']"
15    }
16  },

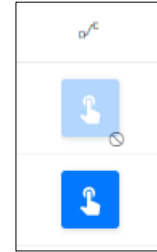
```

Disabling Flows (but not Actions)

If you don't want a flow to run, but want to keep it in your list, turn it off:



Click on the 3 dots next to a flow's name and choose **Turn off**. Disabled flows appear with a lighter background:



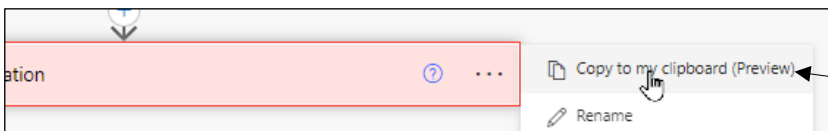
To turn a flow back on again choose the same menu option (which will now say **Turn on**).



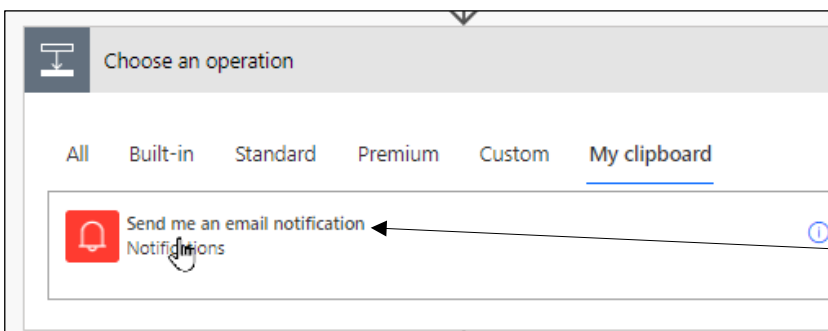
There is no way to disable a single action within a PA flow (more's the pity); only complicated workarounds.

Copying Actions

If you have an action which you may want to repeat, copy it to the PA clipboard:



a) While viewing an action, click on the 3 dots to the right of its name and choose to copy it to your clipboard.



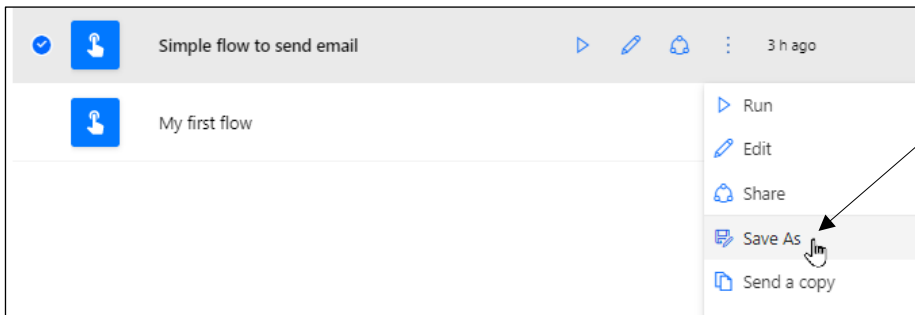
b) When creating another action, you can then retrieve the details of this action from your clipboard for further editing.



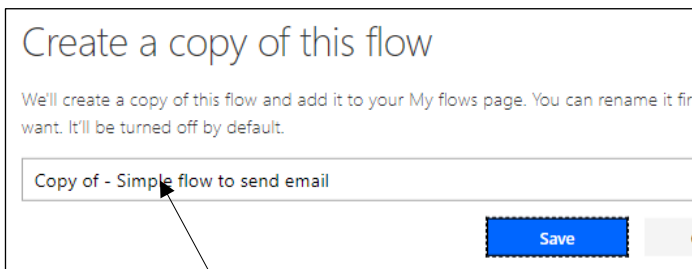
Bug alert! When you copy actions in PA it's possible to create objects (such as variables) with duplicate names. PA then not only prevents you saving your flow, but also makes it hard to edit or even delete the offending actions!

Copying Flows

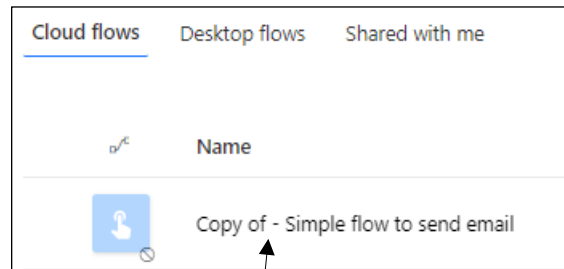
To copy a flow, choose to save it as another name:



a) Click on the 3 dots to the right of a flow's name and choose to save it as another name.



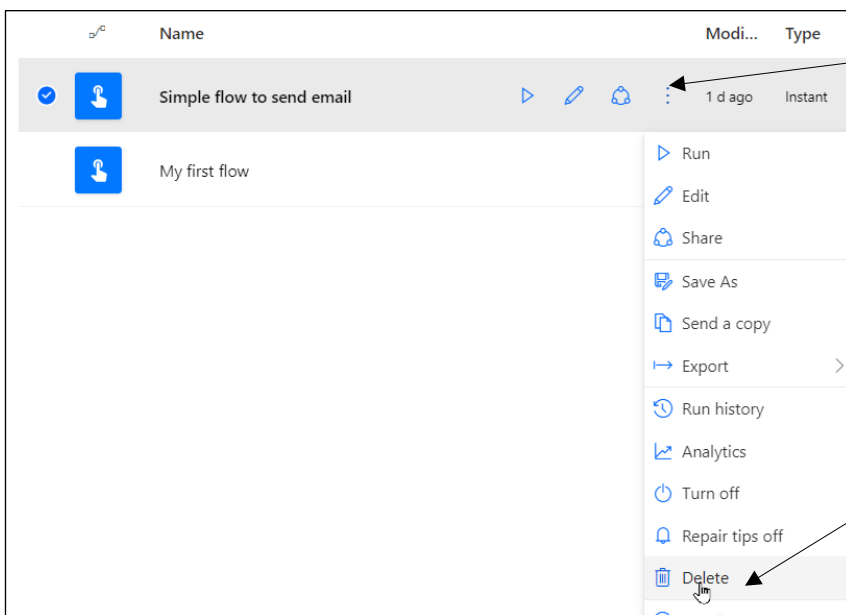
b) Type in a new name for your flow, then choose to save this.



c) PA creates the copy of your original flow, and sensibly ensures it is turned off by default.

Deleting Flows

Sadly you have to delete flows one at a time (there is no way to select more than one flow at the same time):







a) Click on the 3 dots to the right of a flow's name.

b) Choose this option from the menu which appears to delete this flow – you'll be asked to confirm this:

A screenshot of a confirmation dialog box titled 'Delete: Simple flow to send email?'. The text reads: 'Deleting this flow will remove it for all owners and uninstall it for all users. Previous flow instances will continue to run to avoid data loss. This action cannot be undone.' There are 'Delete' and 'Cancel' buttons at the bottom.

WHAT WE DO

| |  |  |  |  | |
|-------------------------|---|---|---|---|---|
| | ONLINE TRAINING | MANCHESTER OR LONDON | AT YOUR OFFICE | BESPOKE CONSULTANCY | |
| OFFICE 365 | Microsoft Excel | ✓ | ✓ | ✓ | ✓ |
| | VBA macros | ✓ | ✓ | ✓ | ✓ |
| | Office Scripts | ✓ | | ✓ | |
| | Microsoft Access | | | | ✓ |
| POWER PLATFORM | Power BI and DAX | ✓ | ✓ | ✓ | ✓ |
| | Power Apps | ✓ | | ✓ | |
| | Power Automate | ✓ | ✓ | ✓ | ✓ |
| SQL SERVER | Reporting Services | ✓ | ✓ | ✓ | ✓ |
| | Report Builder | ✓ | | ✓ | ✓ |
| | Integration Services | ✓ | ✓ | ✓ | ✓ |
| | Analysis Services | ✓ | | ✓ | |
| CODING LANGUAGES | SQL | ✓ | ✓ | ✓ | ✓ |
| | Visual C# | ✓ | ✓ | ✓ | ✓ |
| | Python | ✓ | ✓ | ✓ | ✓ |



