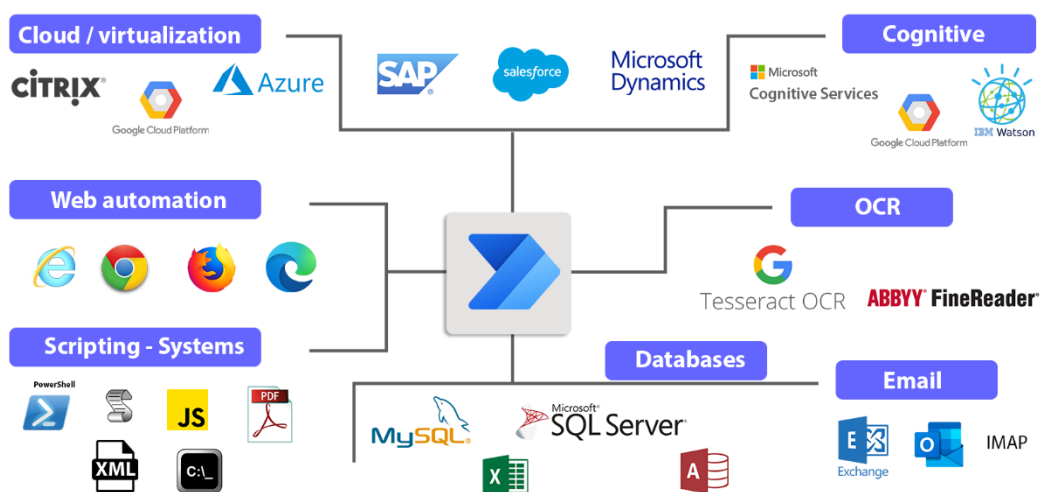
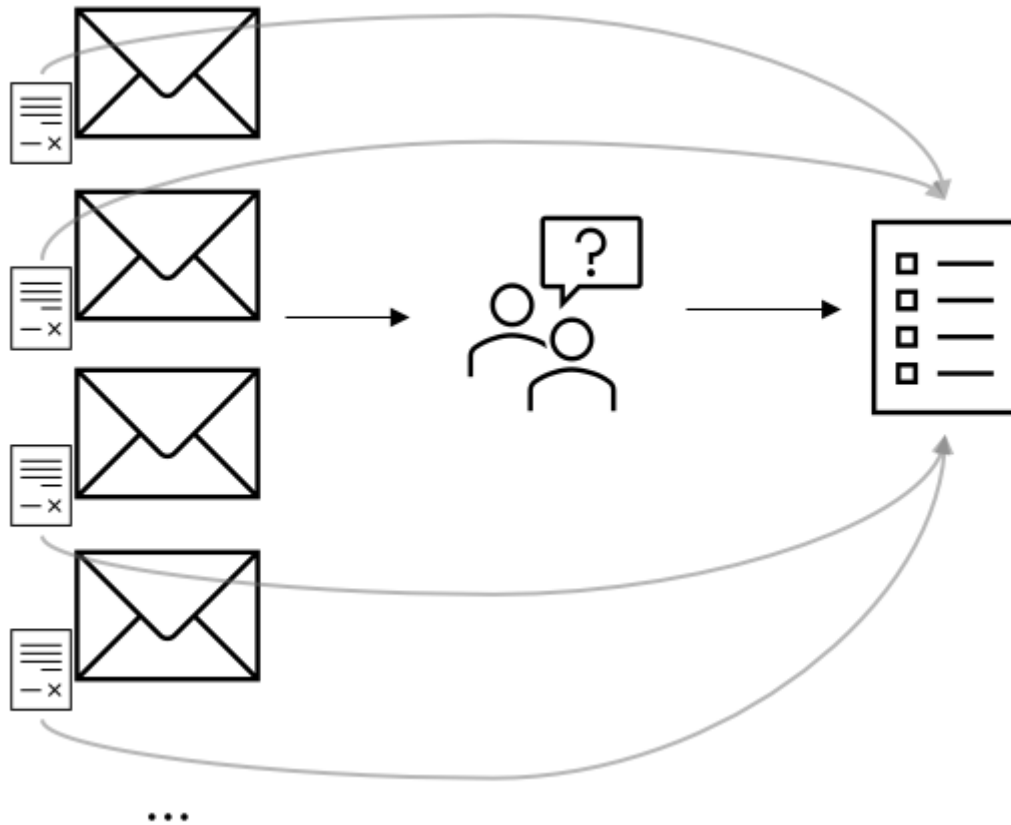


# Chapter 1: Getting Started with Power Automate Desktop





## Microsoft Power Platform

The low code platform that spans Microsoft 365, Azure, Dynamic 365, and standalone apps.



**Power Automate**  
Process automation



**Power Pages**  
External facing websites



**Power BI**  
Business analytics



**Power Virtual Agents**  
Intelligent virtual agents



**Power Apps**  
App development



**Data connectors**



**AI Builder**



**Dataverse**

AutoSave Off Parents Letter - Daniel Hunter.docx • Saved to this PC Peter Krause PK

File Home Insert Draw Design Layout References Mailings Review View Help

Undo Paste Font Paragraph Styles Editing

# Parents Association

Dear Parents,

Hereby I would like to inform you that we will soon have an extraordinary event at our school.  
Please let me know if you agree with the participation of

Name of the child: Daniel Hunter

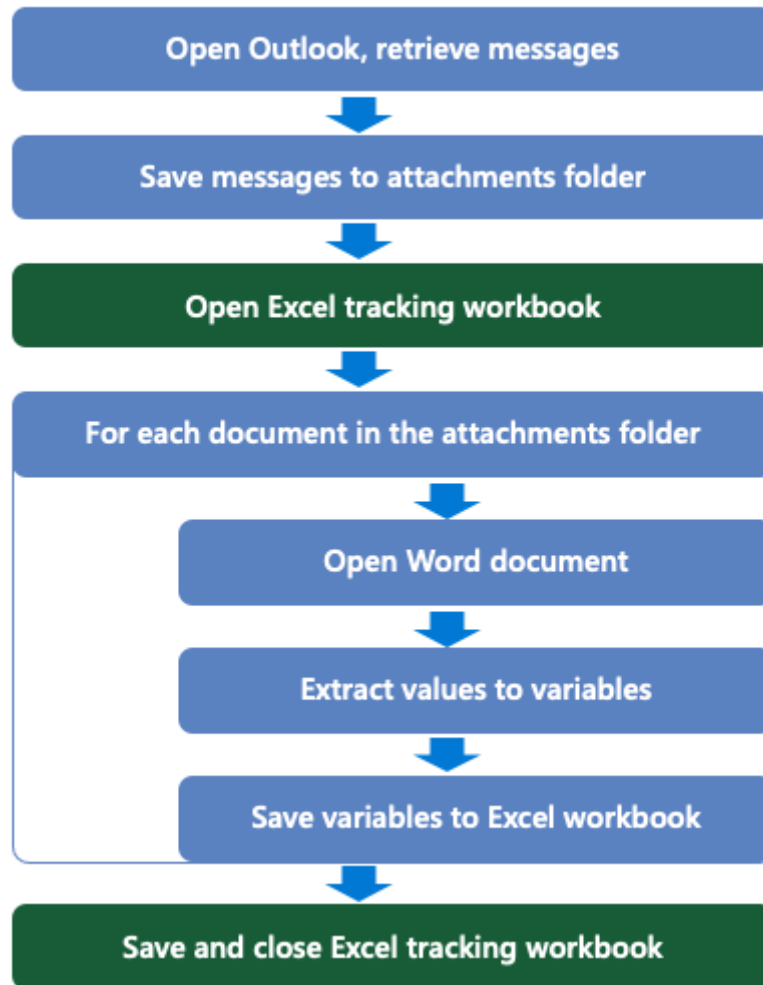
to participate:

☒ Yes, I agree  
☐ No, I don't agree

Please tick one of the boxes and return the document to us.

Sincerely,  
Yours parents' association

Page 1 of 1 66 words English (United States) Accessibility: Investigate Focus 100%



1  
2  
3  
4



#### Launch Outlook

Launch Outlook and store the instance created in `OutlookInstance`



#### Retrieve email messages from Outlook

Retrieve `Unread email messages only` from account `OutlookAccount` and store them in `RetrievedEmails`



#### Process email messages in Outlook

Move email messages `RetrievedEmails` to mail folder in account `OutlookAccount`



#### Close Outlook

Close the Outlook instance stored into `OutlookInstance`

5

**Launch Excel**

Launch Excel and open document `SaveAttachmentsFolder` '\Activity Tracker.xlsx' using an existing Excel process and store it into Excel instance `ExcelInstance`

6

**Get files in folder**

Retrieve the files in folder `SaveAttachmentsFolder` and its subfolders that match '\*.docx' and store them into `Files`

7

**Get current date and time**

Retrieve the current datetime and store it into `CurrentDateTime`

8



**For each** `CurrentItem` in `Files`

9

**Rename file(s)**

Rename the file(s) `CurrentItem` to 'work-in-progress' and store them into list `RenamedFiles`

10

**Run application**

Run application "'C:\Program Files\Microsoft Office\root\Office16\WINWORD.EXE'" with arguments '/t "' `SaveAttachmentsFolder` '\work-in-progress.docx'" and store its process ID into `AppProcessId`

11

**Get selected checkboxes in window**

Get checked state of `Check Box` and store it into `CheckBoxYes`

12

**Get selected checkboxes in window**

Get checked state of `Check Box 2` and store it into `CheckBoxNo`

13

**Extract data from window**

Extract all records from the UI element `NameOfChild` and store the result(s) into `NameOfChild`

14

**Get first free column/row from Excel worksheet**

Get the first free column/row in the active worksheet of the Excel document whose instance is stored into `ExcelInstance` and store them into `FirstFreeColumn` and `FirstFreeRow`

15

**Write to Excel worksheet**

Write the value `NameOfChild` into cell in column `1` and row `FirstFreeRow` of the Excel instance `ExcelInstance`

16

**Write to Excel worksheet**

Write the value `CheckBoxYes` into cell in column `2` and row `FirstFreeRow` of the Excel instance `ExcelInstance`

17

**Write to Excel worksheet**

Write the value `CheckBoxNo` into cell in column `3` and row `FirstFreeRow` of the Excel instance `ExcelInstance`

18

**Write to Excel worksheet**

Write the value `CurrentDateTime` into cell in column `4` and row `FirstFreeRow` of the Excel instance `ExcelInstance`

19

**Terminate process**

Terminate process with ID `AppProcessId`

20



**End**



21



#### Delete file(s)

Delete file(s) `SaveAttachmentsFolder` `"work-in-progress.docx"`

22



#### Save Excel

Save the Excel document stored into `ExcelInstance`

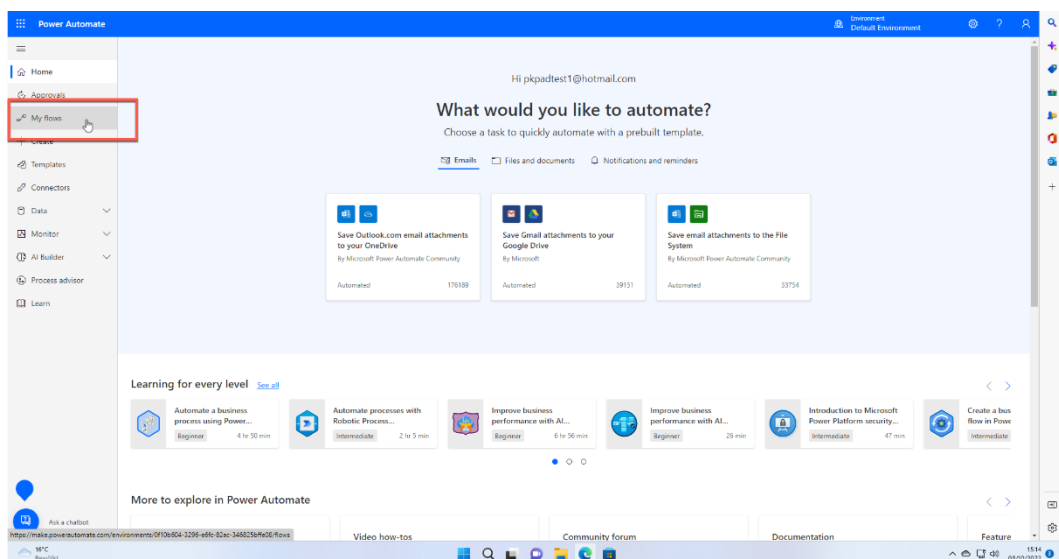
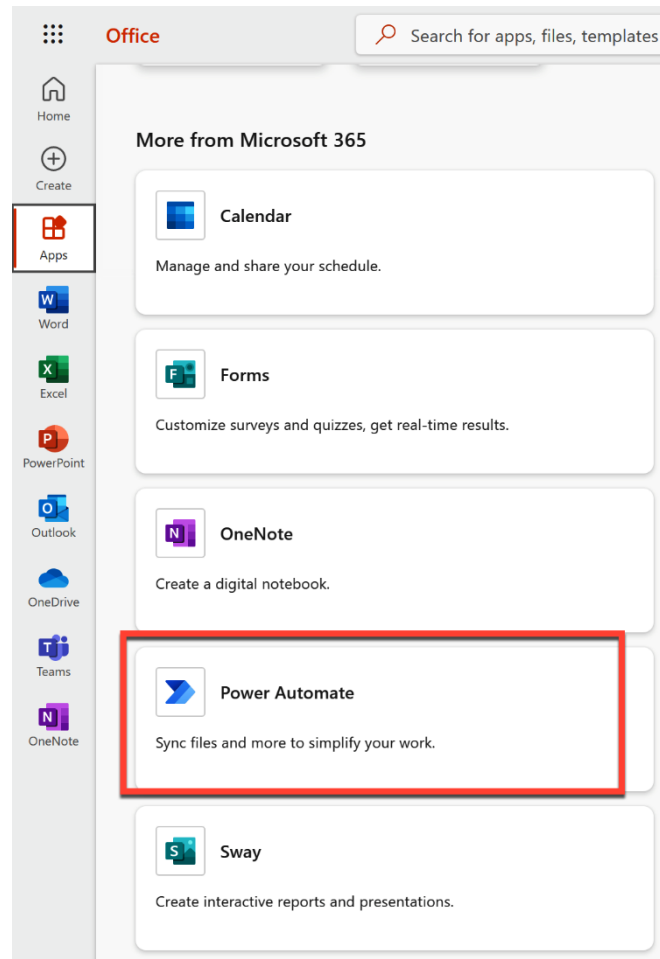
23

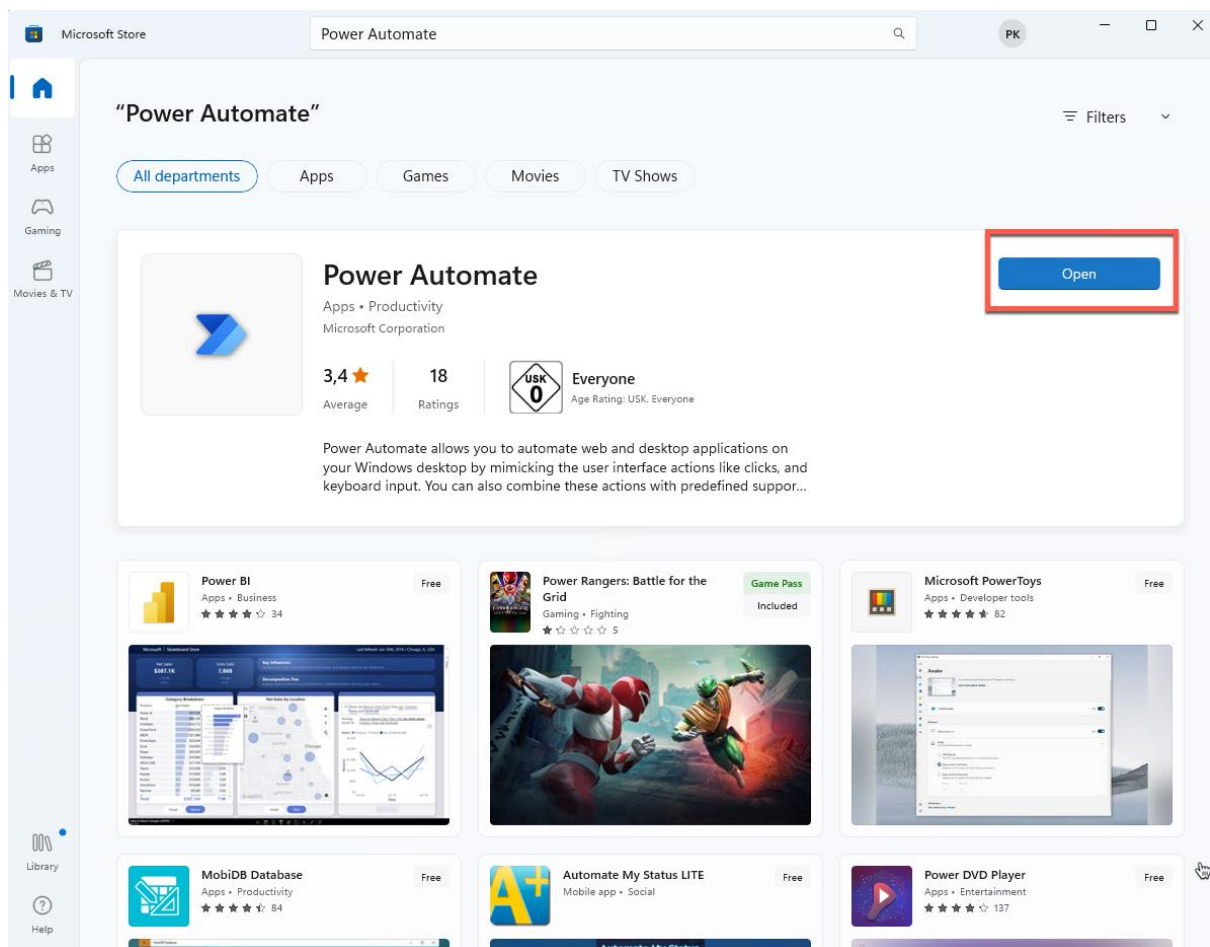
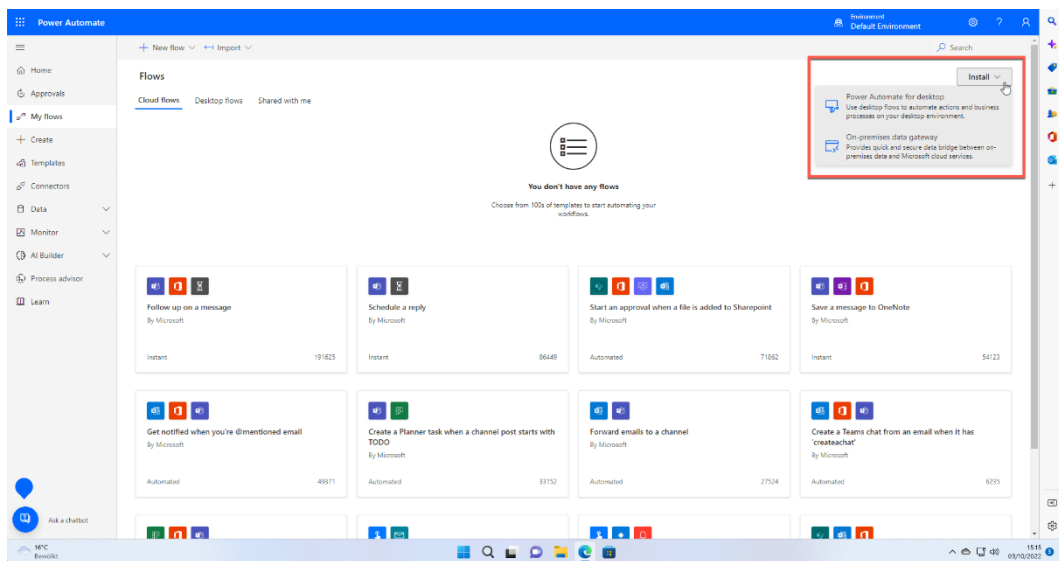


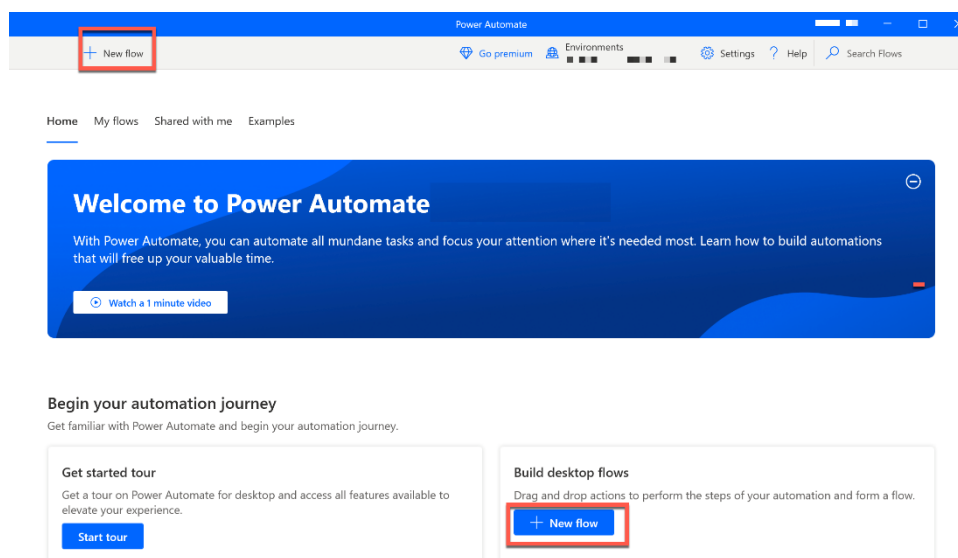
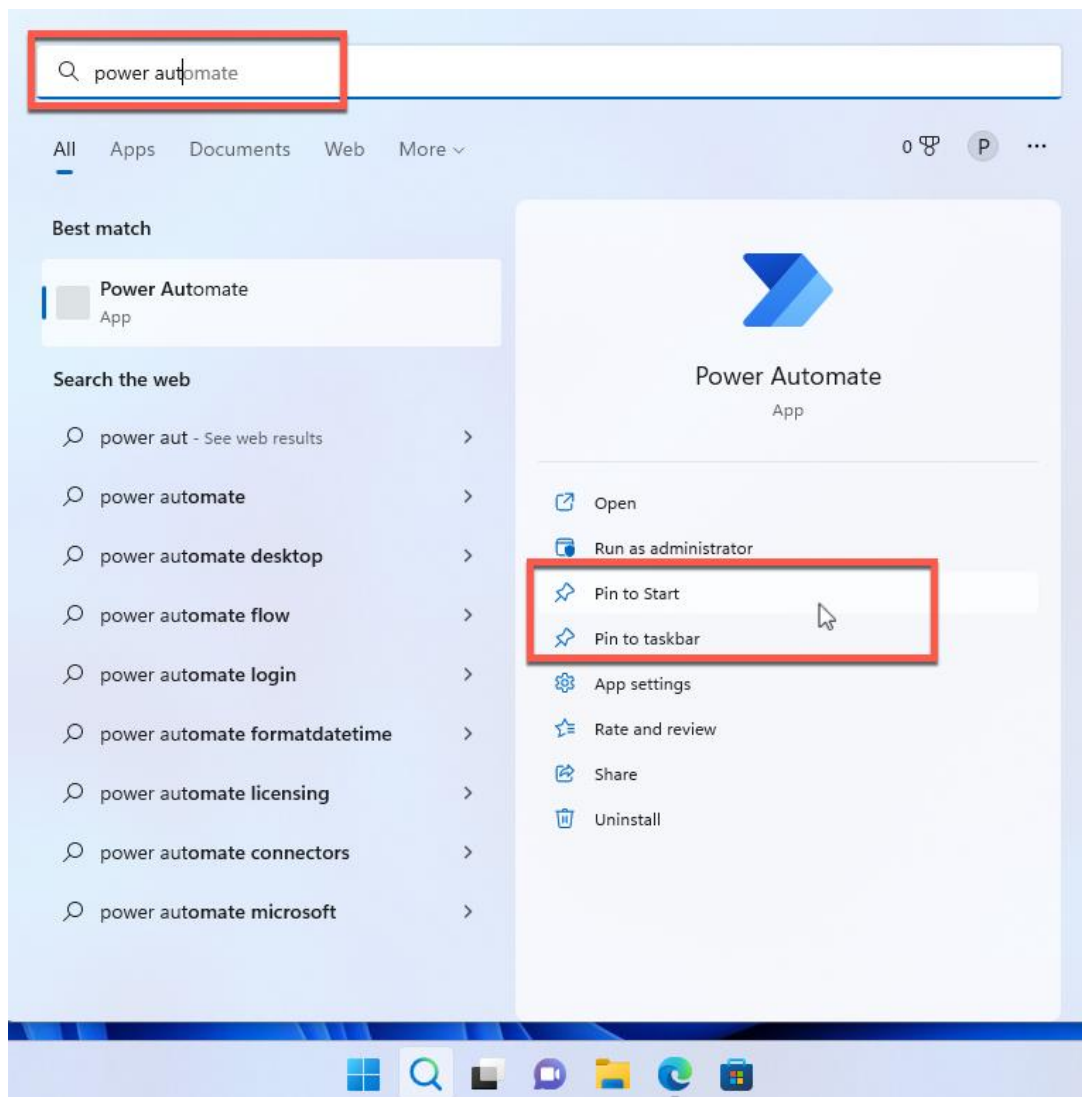
#### Close Excel

Save the Excel document and close the Excel instance `ExcelInstance`

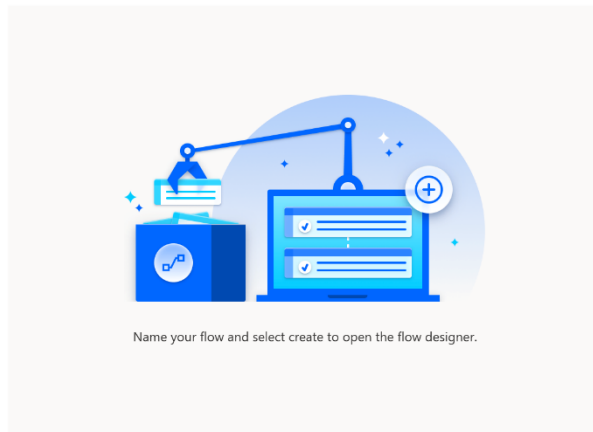
## Chapter 2: Using Power Automate Desktop and Creating First Flow







## Build a flow



Flow name

Welcome Flow

Create Cancel

File Edit Debug Tools View Help Welcome Flow | Power Automate

**Actions**

Search actions

- > Variables
- > Conditionals
- > Loops
- > Flow control
- > Run flow
- > System
- > Workstation
- > Scripting
- > File
- > Folder
- > Compression
- > UI automation
- > HTTP
- > Browser automation
- > Excel
- > Database
- > Email
- > Exchange
- > Outlook
- > **Message boxes**
- > Mouse and keyboard
- > Clipboard
- > Text

Subflows Main

You don't have any actions here yet

To get started, drag and drop actions to this pane or use the recorder to capture the actions

**Variables**

Search variables

Input / output variables 0

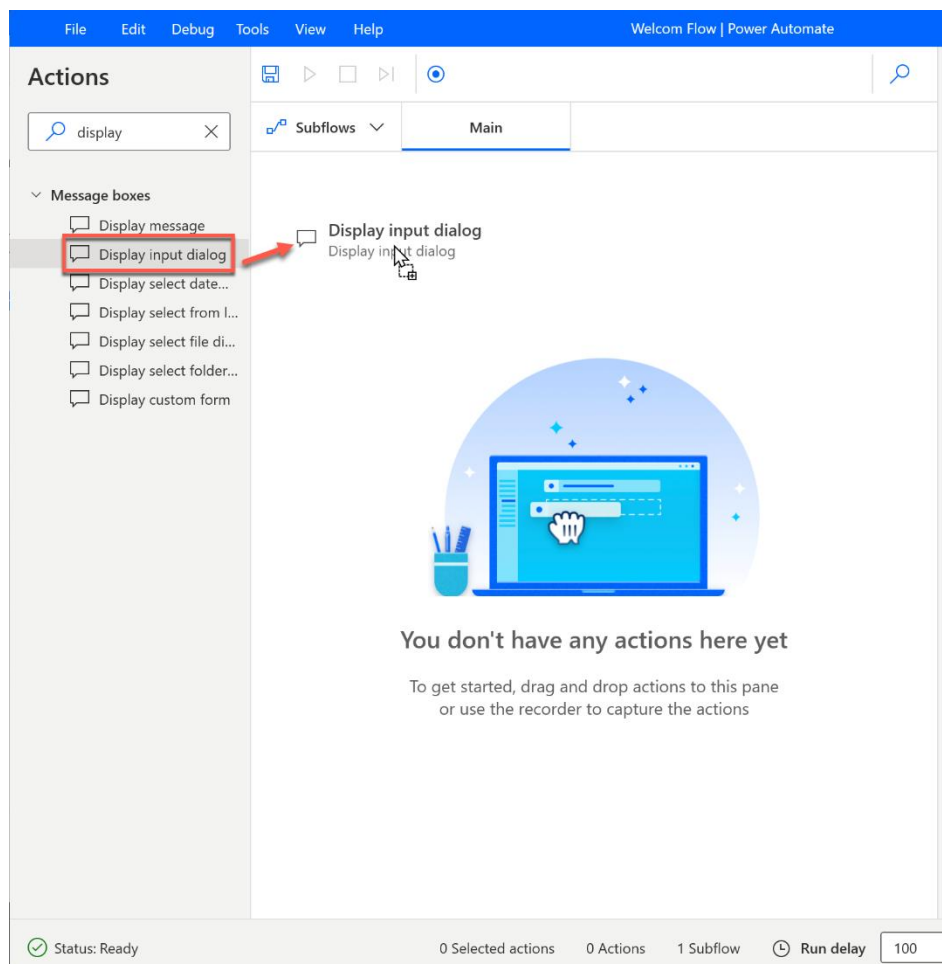
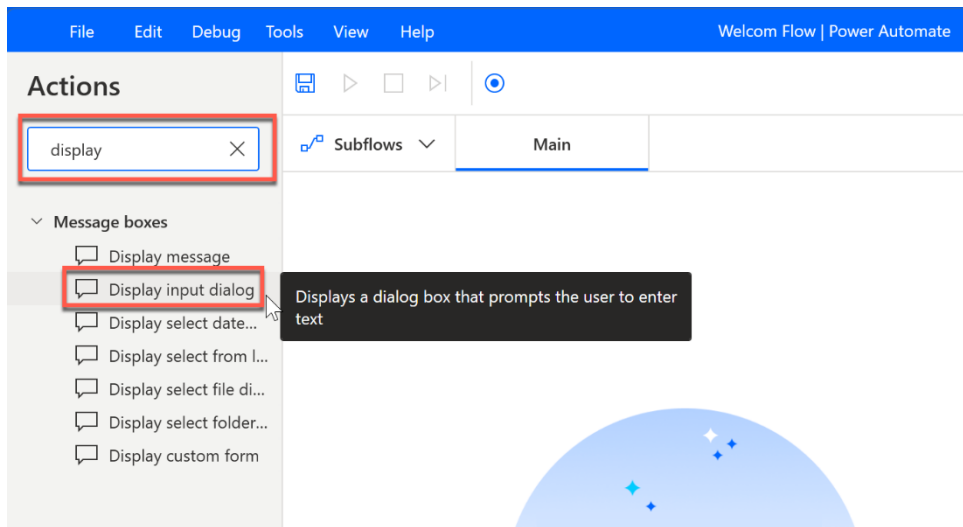
There aren't any input or output variables here yet

Flow variables 0

No variables to display

Interact with users and request input through message boxes

Status: Ready 0 Selected actions 0 Actions 1 Subflow Run delay 100 ms



Display input dialog

Displays a dialog box that prompts the user to enter text [More info](#)

Select parameters

1

Input dialog title:

What is your name?

{x}

i

2

Input dialog message:

Please tell me your name:

{x}

i

Default value:

{x}

i

Input type:

Single line

i

Keep input dialog always on top:

☐

i

> Variables produced

UserInput

ButtonPressed

3

On error

4

Save

Cancel

File Edit Debug Tools View HelpWelcome Flow | Power Automate

Actions

Search actions

> Scripting

> File

> Folder

> Compression

> UI automation

> HTTP

> Browser automation

> Excel

> Database

> Email

> Exchange

> Outlook

> Message boxes

Display message

Display input dialog

Display select date dialog

Display select from list dialog

Display select file dialog

Subflows Main

1

Display input dialog

Display input dialog with message 'Please tell me your name' in the notification popup window with title 'What is your name?' and store the user's input into UserInput and button pressed into...

Variables

Search variables

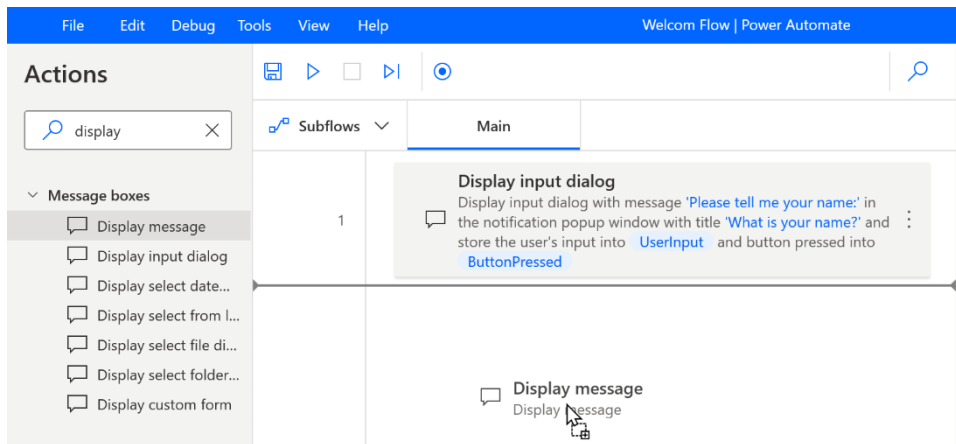
Input / output variables 0

There aren't any input or output variables here yet

Flow variables 2

ButtonPressed

UserInput



### Display message

Displays a message box [More info](#)

Select parameters

Message box title: **1** Hello and welcome to PAD!! {x} ⓘ

Message to display: **2** Hello {x} ⓘ **3**

Message box icon: Search variables ⓘ

Message box buttons: ⓘ

Default button: ⓘ

Keep message box always on top: **4** ⓘ

Close message box automatically: ⓘ

> Variables produced ButtonPressed ⓘ

**5** Select Cancel

**6** Save Cancel

Name	Type
Flow variables	
> ButtonPressed	Text value
> UserInput	Text value



## Display message

Displays a message box [More info](#)

### Select parameters

Message box title:  {x} ⓘ

Message to display:  {x} ⓘ

The screenshot shows the Power Automate interface with a flow named 'Welcom Flow'. The 'Actions' pane on the left lists various message box actions, with 'Display message' selected. The main workspace shows a flow with two steps: 'Display input dialog' (step 1) and 'Display message' (step 2). The 'Display message' step is highlighted, showing its configuration: 'Display message' with the message 'Hello \* UserInput' and title 'Hello and welcome to PAD!!'. The 'UserInput' variable is linked to the 'Display input dialog' step's output.

This screenshot shows the Power Automate interface with the 'Display message' action selected. A dialog box titled 'What is your name?' is displayed, prompting the user to enter their name. The dialog box has a text input field with the placeholder 'Please tell me your name:' and the name 'Peter' entered. The 'OK' button is highlighted. The background shows the flow configuration with the 'Display message' step selected, and the 'Variables' pane on the right showing the 'UserInput' variable.

File Edit Debug Tools View Help Welcome Flow | Power Automate

Subflows Main

1 Display input dialog  
Display input dialog with message 'Please tell me your name;' in the notification popup window with title 'What is your name?' and store the user's input into UserInput and button pressed into ButtonPressed

2 Display message  
Display message 'Hello ' UserInput in the notification popup window with title 'Hello and welcome to PAD!!' and store the button pressed into ButtonPressed2

Variables

Search variables

Input / output variables 0

There aren't any input or output variables here yet

Flow variables 3

- ButtonPressed OK
- ButtonPressed2
- UserInput Peter

Hello and we...  
Hello Peter  
OK

Power Automate

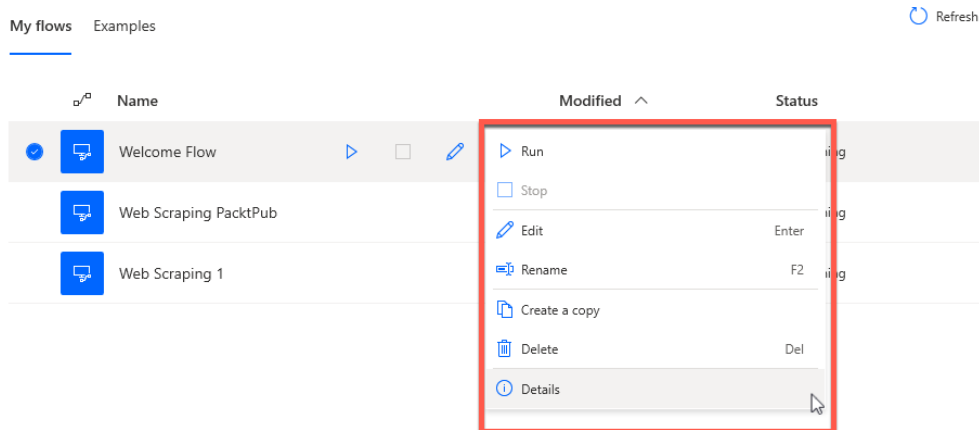
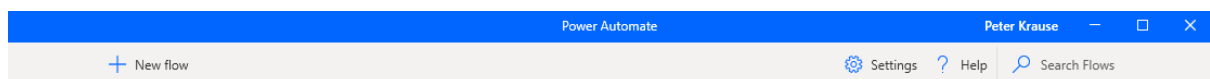
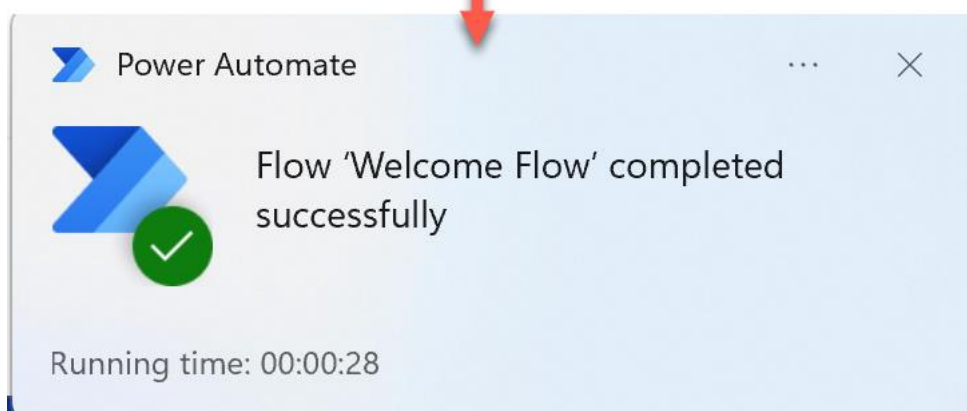
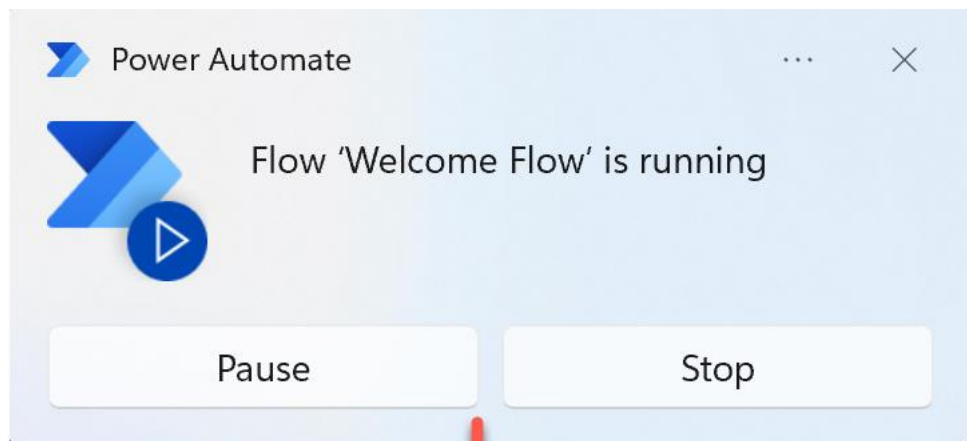
5 Peter Krause

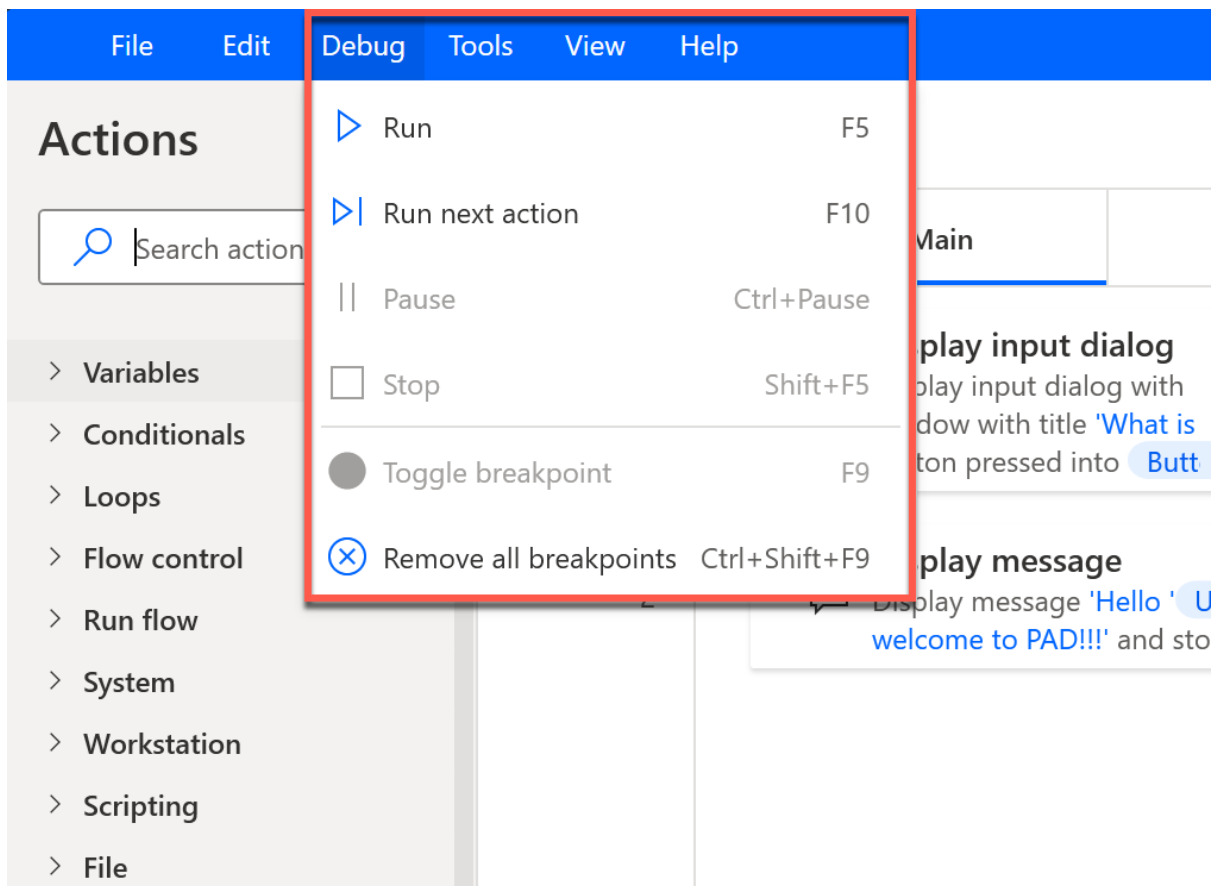
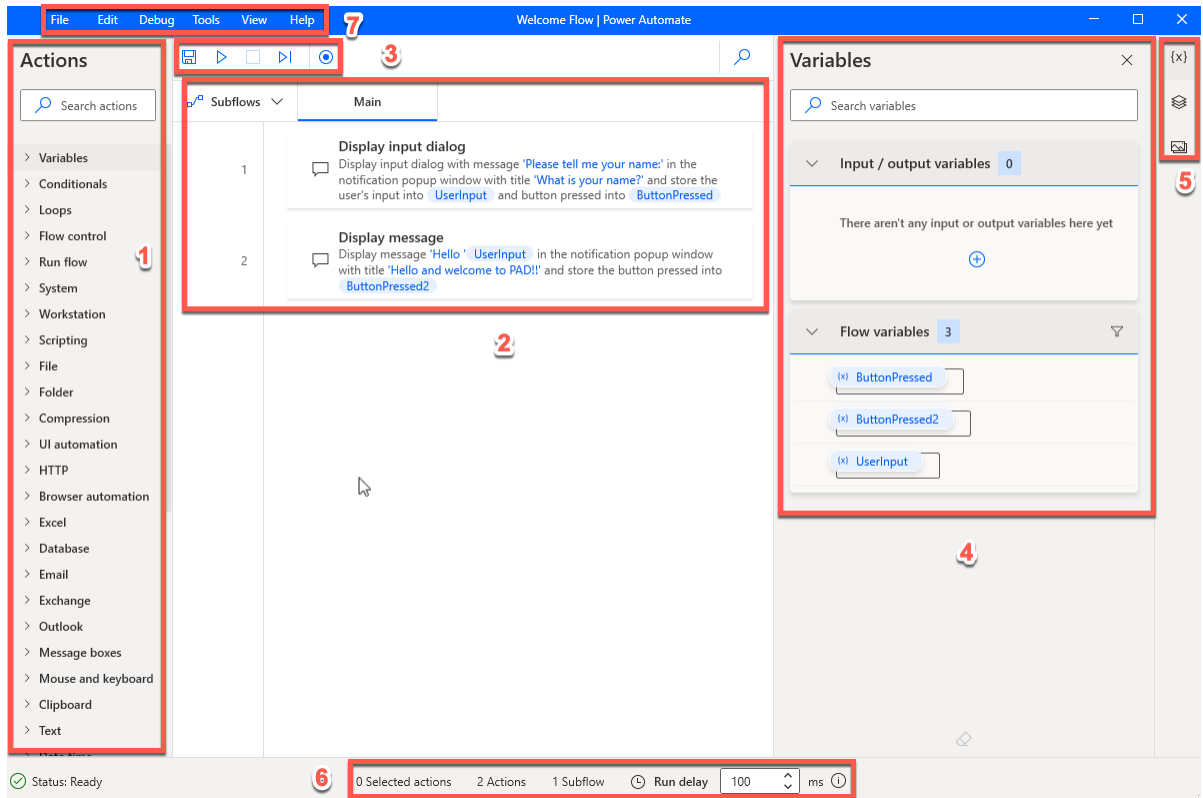
New flow Run Stop Edit ...

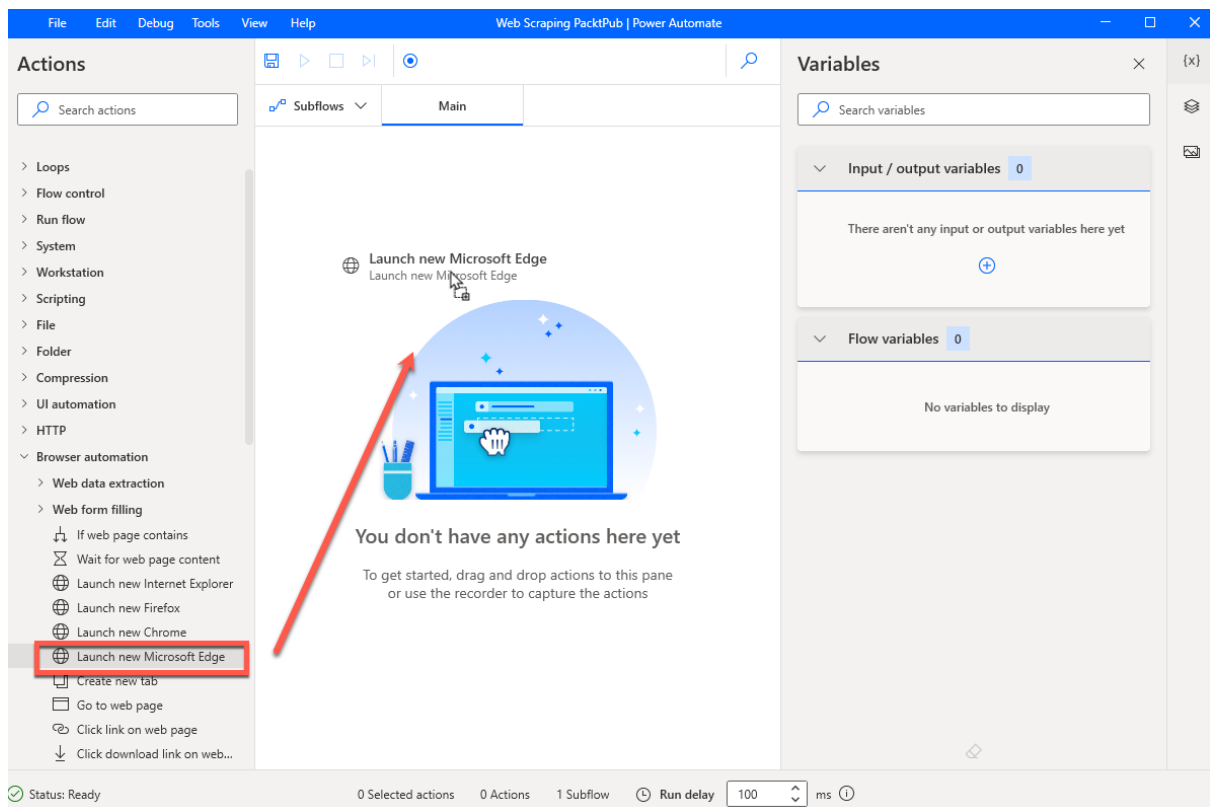
Settings ? Help Search Flows

Home My flows Examples Refresh

	Name	Modified	Status
	Images - PT1	2 months ago	Not running
	Welcome Flow	2 months ago	Not running
	Loops - For ludo game	2 months ago	Not running
1	UI Automation PT1	1 month ago	Not running
	UI Automation - Access 2 Contoso	1 month ago	Not running
	07 - Windows Services	1 month ago	Not running
	07 - Workstation Management	1 month ago	Not running
	07 - CMD session	1 month ago	Not running







**You need the Power Automate extension.**



1. To use web automation in your flows, select 'Get Extension' to download and install to the selected browser.
2. After installing, be sure to enable the extension in the browser.

[Get Extension](#)

[Cancel](#)

## Launch new Microsoft Edge



Launch a new instance of Microsoft Edge for automating web sites and web applications [More info](#)

### Select parameters

Launch mode:  ⓘ

Initial URL:  {x} ⓘ

Window state:  ⓘ

#### > Advanced

Variables produced



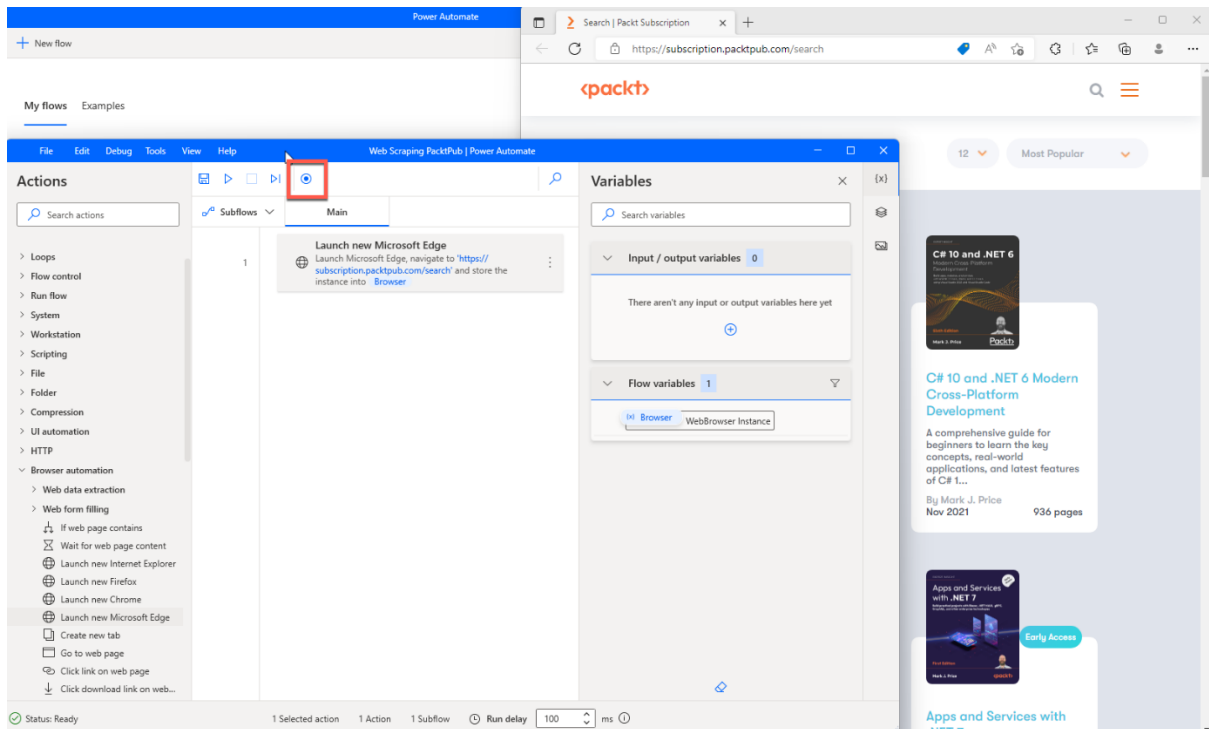
Browser {x}

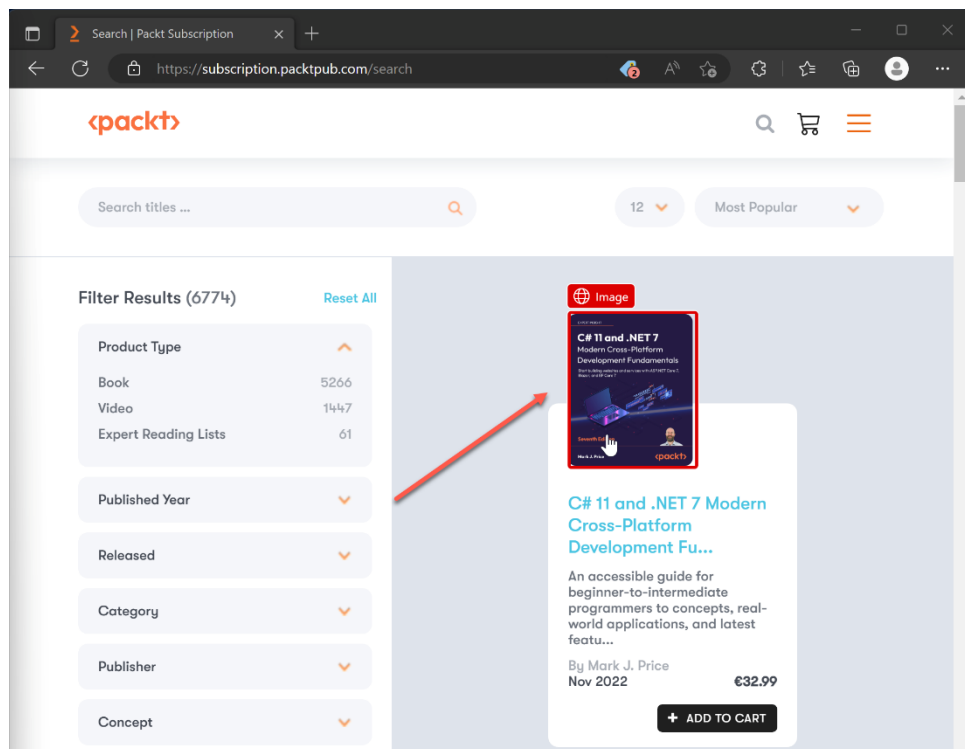
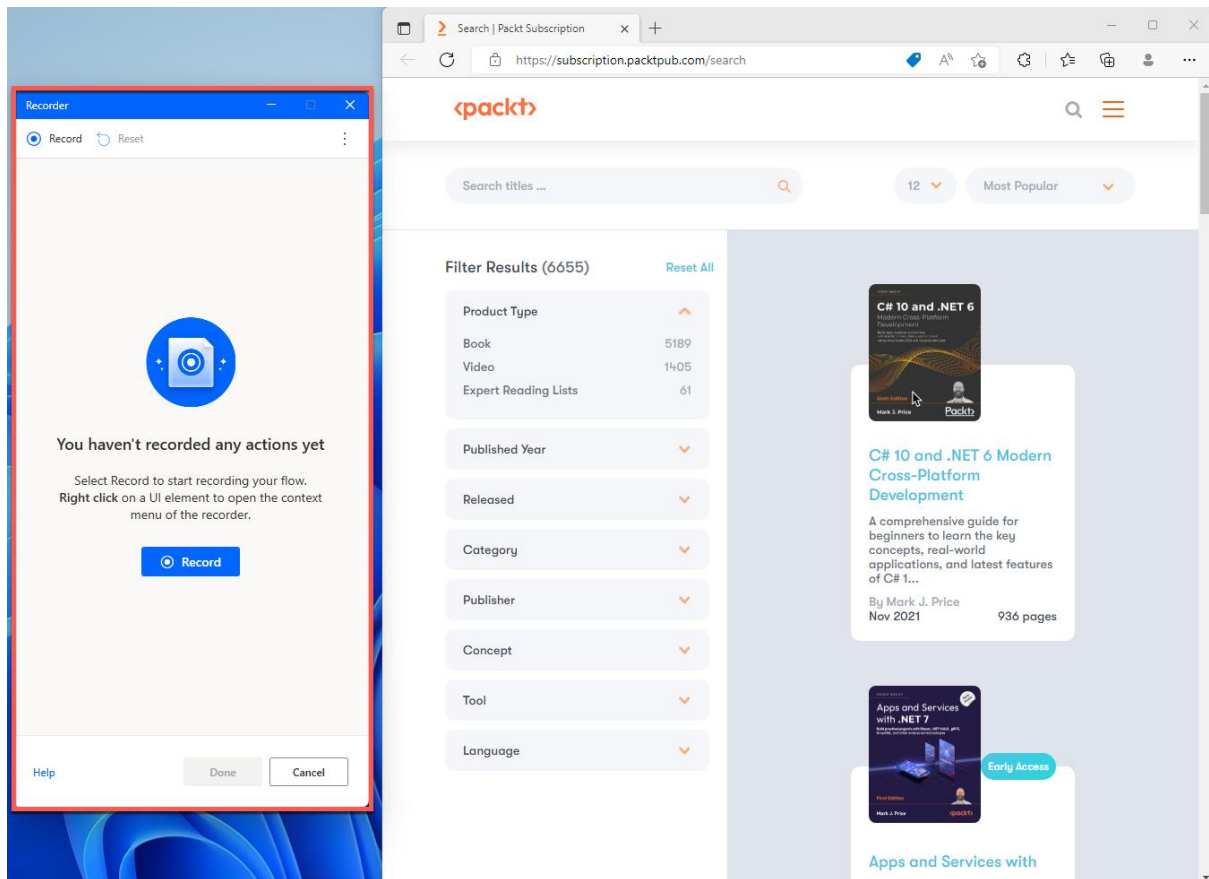
The Microsoft Edge instance to use with web automation actions

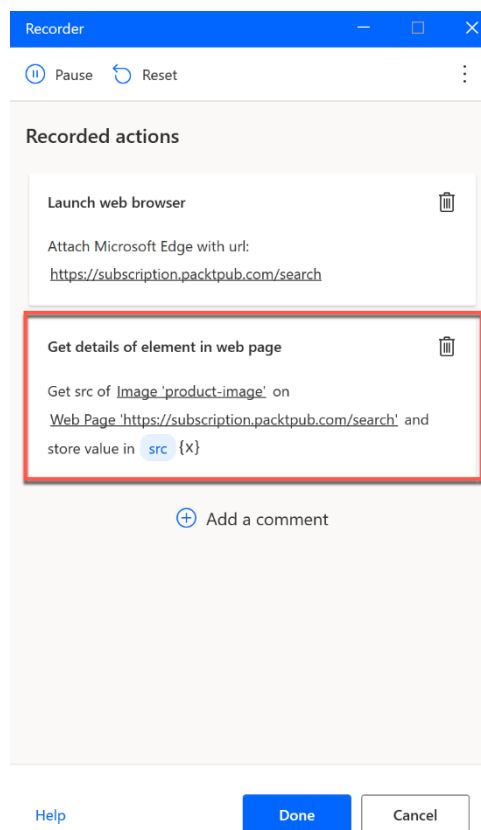
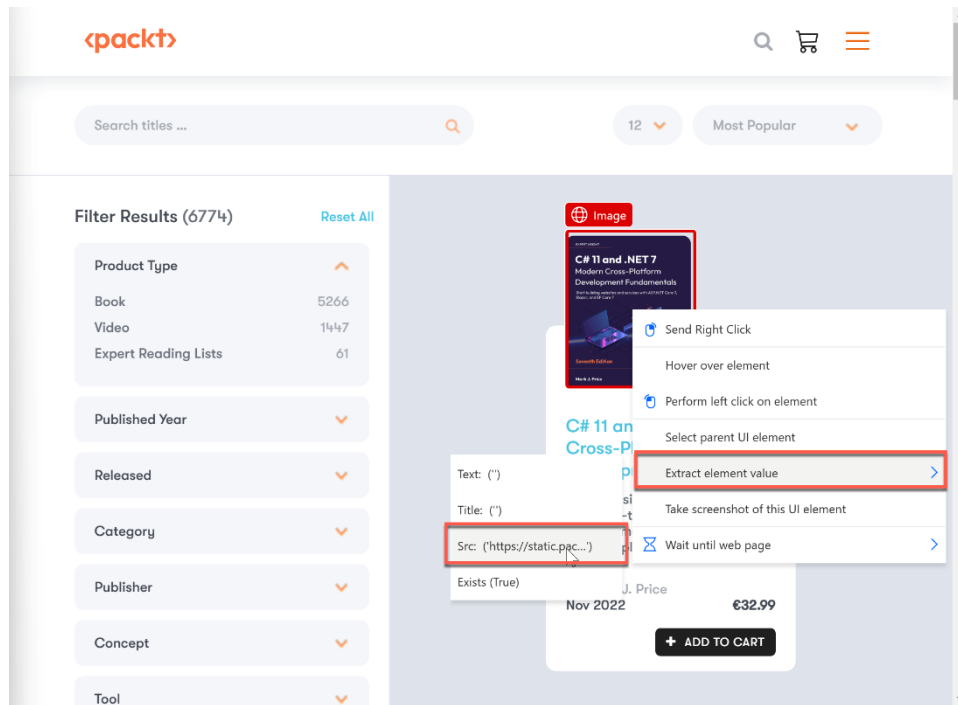
On error

Save

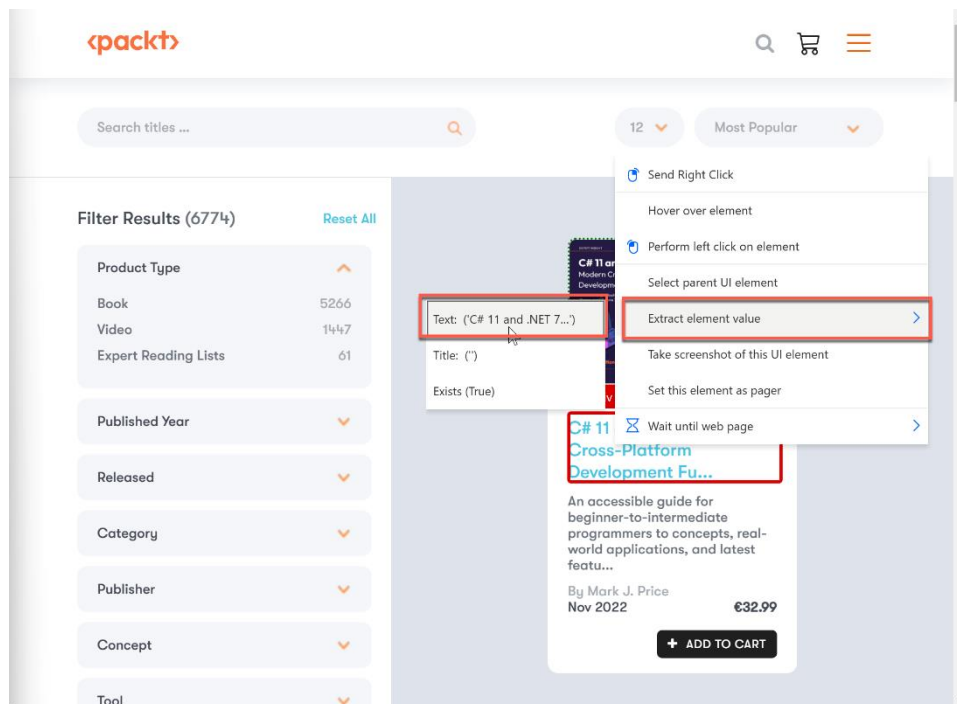
Cancel











## Recorder



Pause



Reset



### Recorded actions

#### Launch web browser



Attach Microsoft Edge with url:

<https://subscription.packtpub.com/search>

#### Extract data



Extract handpicked record(s) in the form of a 2-column data row and store value in `OutputData {x}`



Add a comment

Expert Reading Lists

61

Published Year



Released



Category



Publisher



Concept



Tool



Language



### C# 11 and .NET 7 Modern Cross-Platform Development Fu...

An accessible guide for beginner-to-intermediate programmers to concepts, real-world applications, and latest featu...

By Mark J. Price  
Nov 2022

€32.99

+ ADD TO CART



### Solutions Architect s Handbook



Early Access

### C# 11 and .NET 7 Modern Cross-Platform Development Fu...

An accessible guide for beginner-to-intermediate programmers to concepts, real-world applications, and latest featu...

By Mark J. Price  
Nov 2022

808 pages

<< < 1 2 3 4 5 6 7 > >>

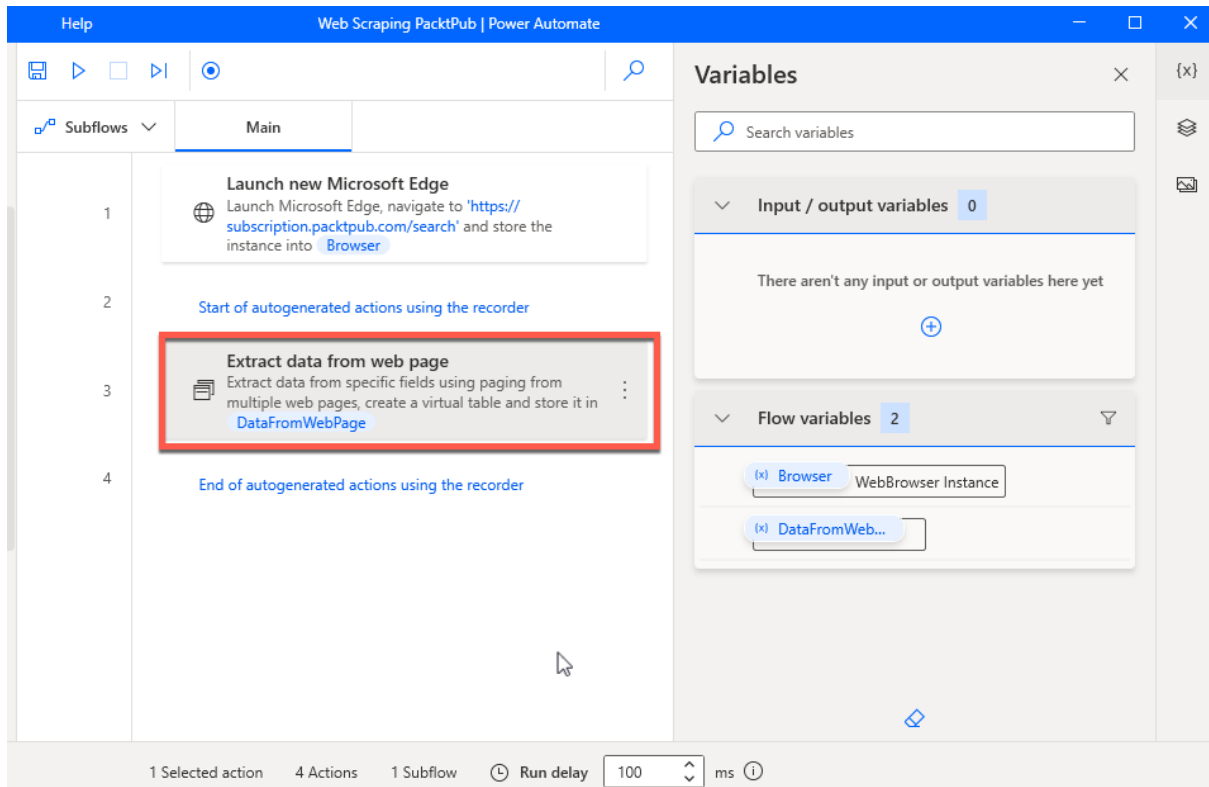
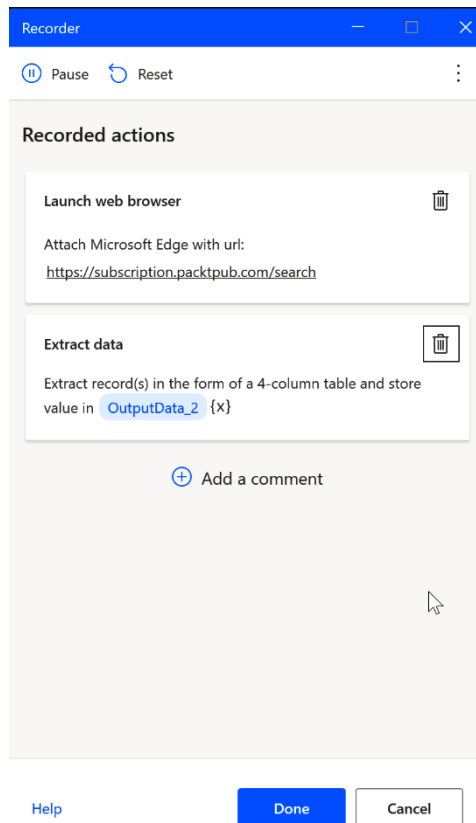
Mar 2023

Coming Soon



- 🖱 Send Right Click
- 🖱 Press button
- Hover over element
- 🖱 Perform left click on element
- Select parent UI element
- Extract element value >
- Take screenshot of this UI element
- Set this element as pager**
- ⌚ Wait until web page >

🌐 Button



## Extract data from web page



Extract data from specific parts of a web page in the form of single values, lists, rows or tables  
[More info](#)

### Select parameters

Web browser instance:

%Browser%



Bringing an actual web browser window to the foreground, while this dialog is open, will activate the live web helper.

Synopsis of data to be extracted: **Extract record(s) from multiple web-pages in the form of a 4-column table.**

Extract data from:

All available



Process data upon extraction:



Timeout:

60



Store data mode:

Variable



> Variables produced

OutputData\_2




On error

Save

Cancel

## Extract data from web page



 Extract data from specific parts of a web page in the form of single values, lists, rows or tables  
[More info](#)

Web browser instance:  

Bringing an actual web browser window to the foreground, while this dialog is open, will activate the live web helper.


Synopsis of data to be extracted: **Extract record(s) from multiple web-pages in the form of a 4-column table.**

Extract data from:  

Max web pages to process:  {x} 

Process data upon extraction: ☐ 

Timeout:  {x} 

Store data mode:  

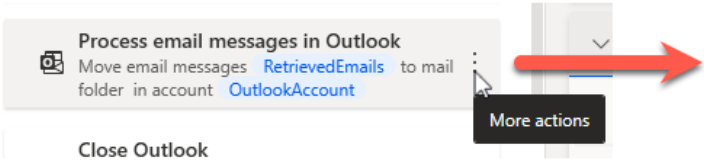
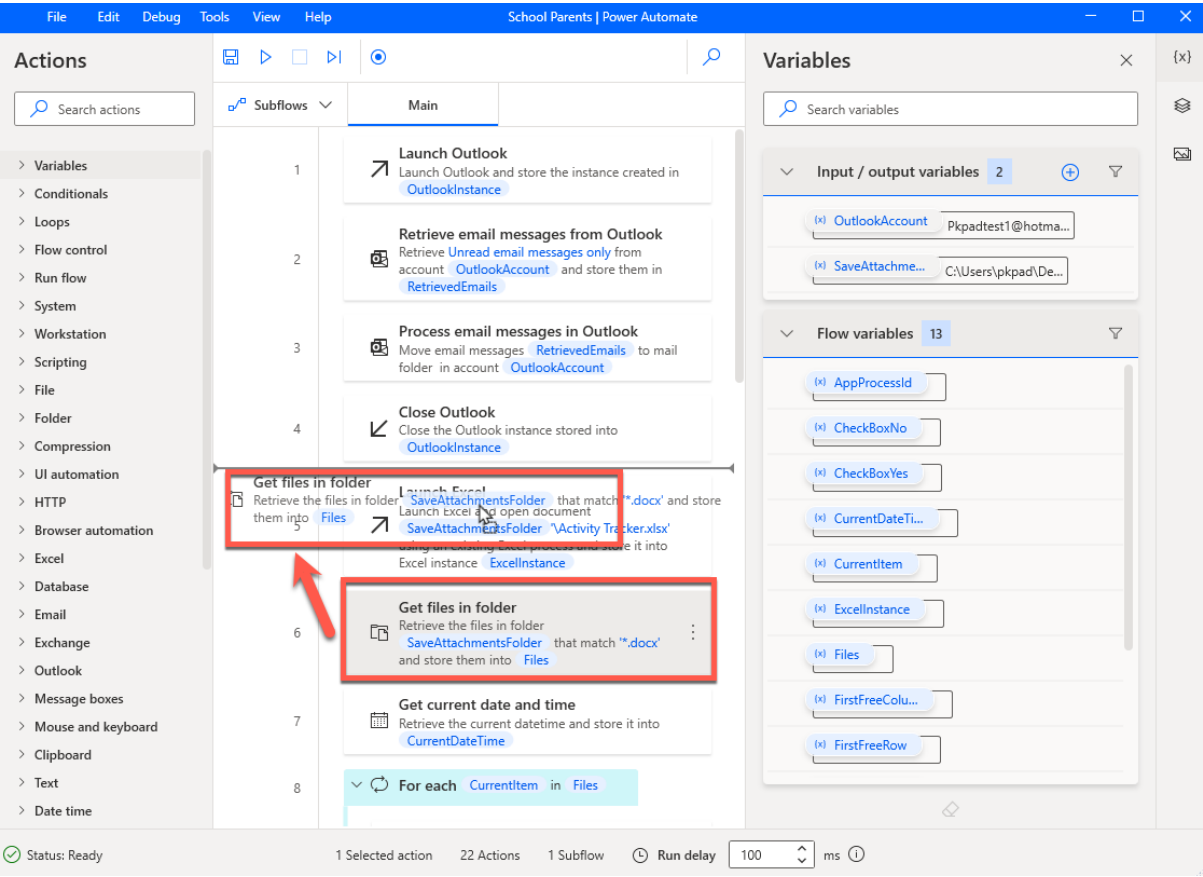
> Variables produced ExcelInstance

 On error

Save

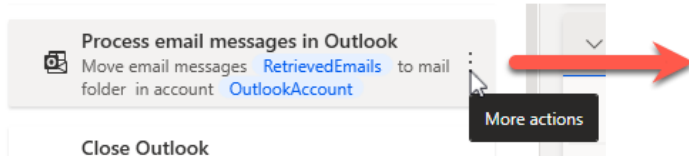
Cancel

# Chapter 3: Editing and Debugging UI Flows



	Edit	Enter
	Run from here	Alt+F5
	Undo	Ctrl+Z
	Redo	Ctrl+Y
	Cut	Ctrl+X
	Copy	Ctrl+C
	Paste	Ctrl+V
	Move up	Shift+Alt+Up
	Move down	Shift+Alt+Down
	Disable action	
	Delete	Del





	Edit	Enter
	Run from here	Alt+F5
	Undo	Ctrl+Z
	Redo	Ctrl+Y
	Cut	Ctrl+X
	Copy	Ctrl+C
	Paste	Ctrl+V
	Move up	Shift+Alt+Up
	Move down	Shift+Alt+Down
Disable action		
	Delete	Del

File Edit Debug Tools View Help Conditions | Power Automate

**Actions**

Search actions

- > Variables
- > Conditionals
  - Case
  - Default case
  - Else
  - Else if
  - If
  - Switch
- > Loops

Subflows Main

- 1 If "Test1" = "Peter" then
- 2 End
- 3 If process  
If process "msdtc" is running
- 4 End

**Variables**

Search variables

Input / output variables 0

I here aren't any input or output variables here yet

Flow variables 0

**Errors 1**

Subflow	Action	Error
Main	3	Parameter "": End statement is missing.

Status: Errors found 1 Selected action 4 Actions 1 Subflow Run delay 100 ms 1 Error

File Edit Debug Tools View Help Conditions | Power Automate

**Actions**

Search actions

- > Variables
- > Conditionals
  - Case
  - Default case
  - Else
  - Else if
  - If
  - Switch
- > Loops
- > Flow control

Subflows Main

1 If 'Test1' = 'Peter' then

2 End

3 If process  
If process 'msdtc' is running

4 End

**Variables**

Search variables

Input / output variables 0

There aren't any input or output variables here yet

Flow variables 0

**Errors 1**

Subflow	Action	Error
Main	3	Parameter '': End statement is missing.

Status: Errors found 1 Selected action 4 Actions 1 Subflow Run delay 100 ms 1 Error

## Get files in folder

Retrieve the list of files in a folder [More info](#)

### Select parameters

Folder: %SaveAttachmentsFolder% {x} ⓘ

File filter: \*.docx {x} ⓘ

Include subfolders: ⓘ

> Advanced

> Variables produced **Files**


On error

Save

Cancel

## Get files in folder



 Retrieve the list of files in a folder [More info](#)

### Select parameters

Folder:   {x} 

File filter:  {x} 

Include subfolders: ☐ 

> Advanced

> Variables produced **Files**

 On error

Save

Cancel

#### Variables produced



{x}

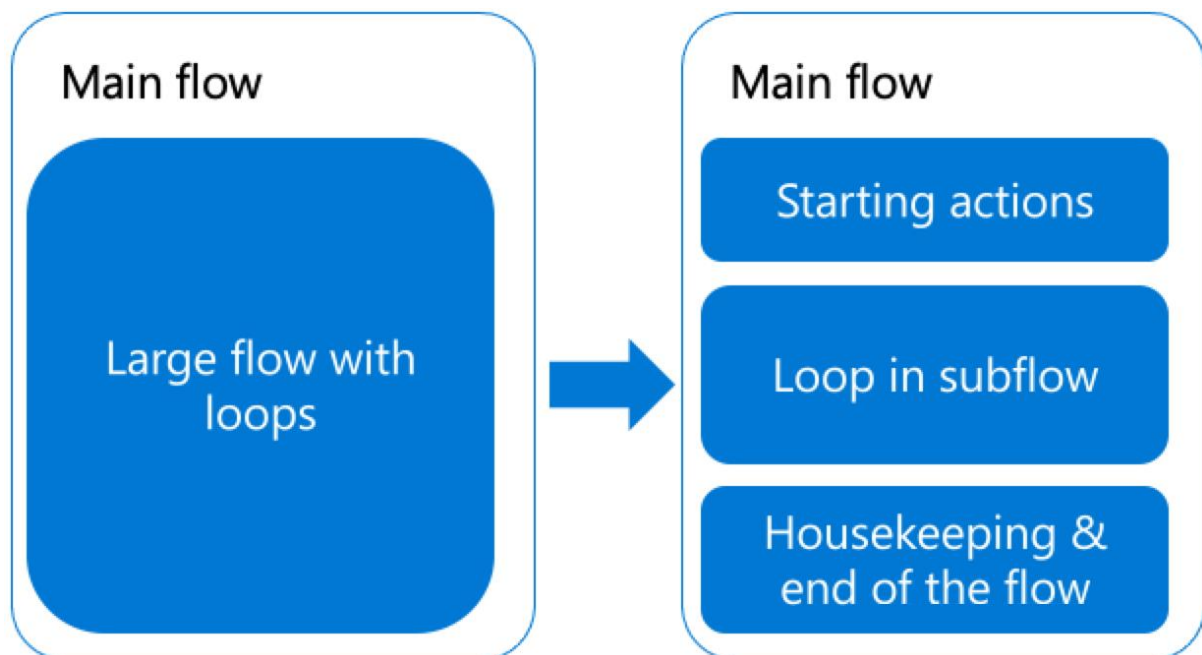
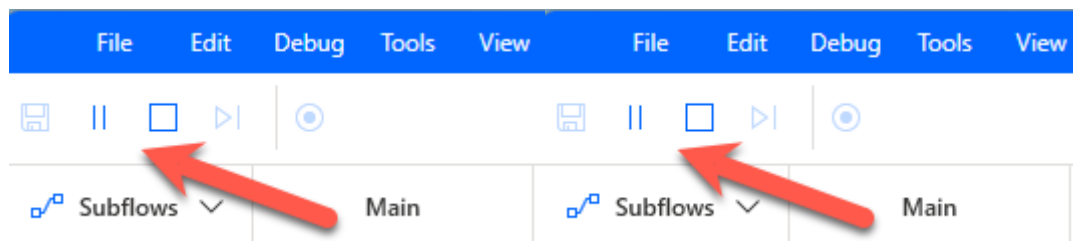
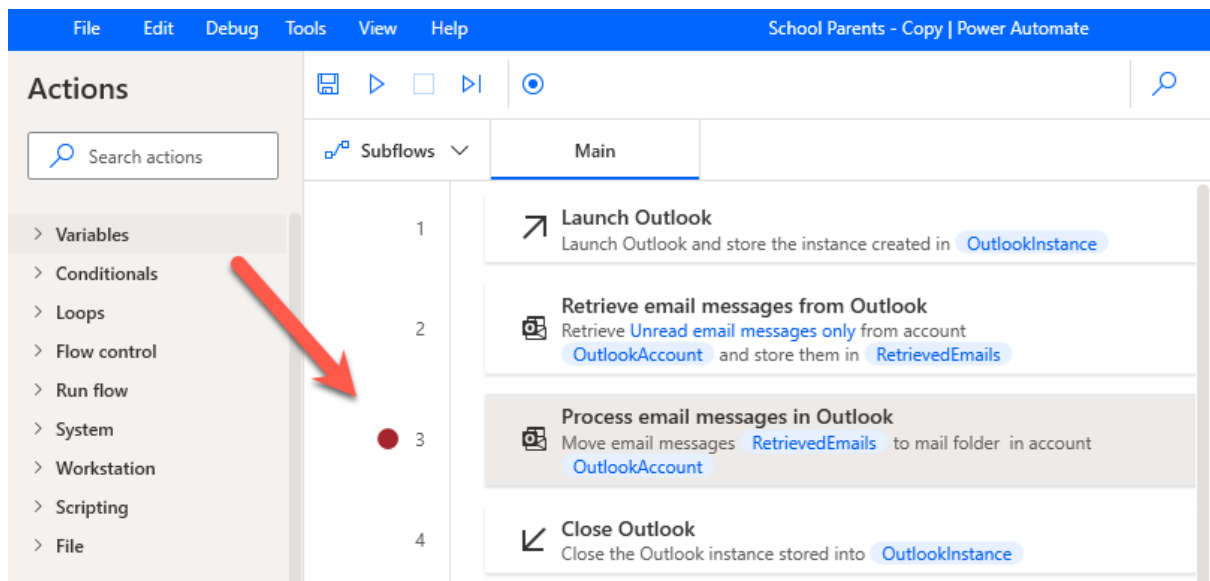
The retrieved files as a list of file objects

#### Variables produced

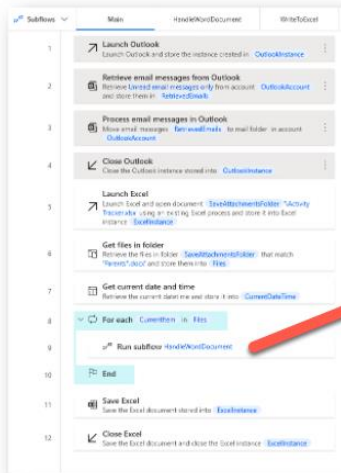


{x}

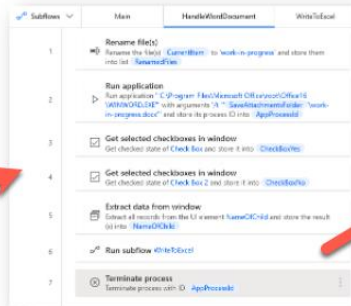
The retrieved files as a list of file objects



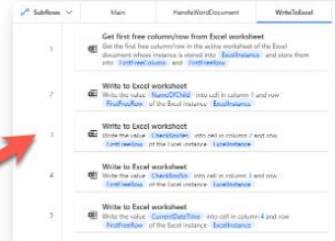
## Main Flow



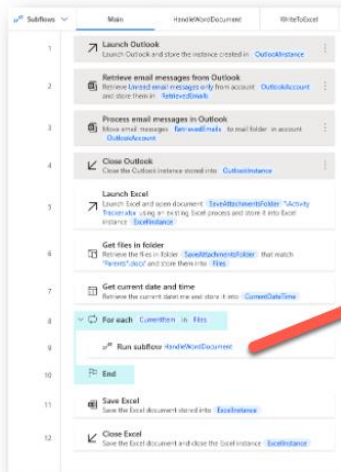
## HandleWordDocument Subflow



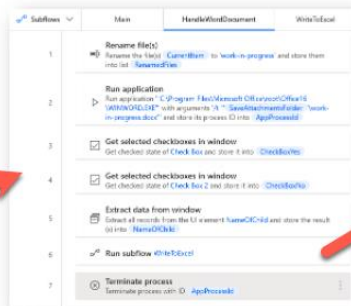
## WriteToExcel Subflow



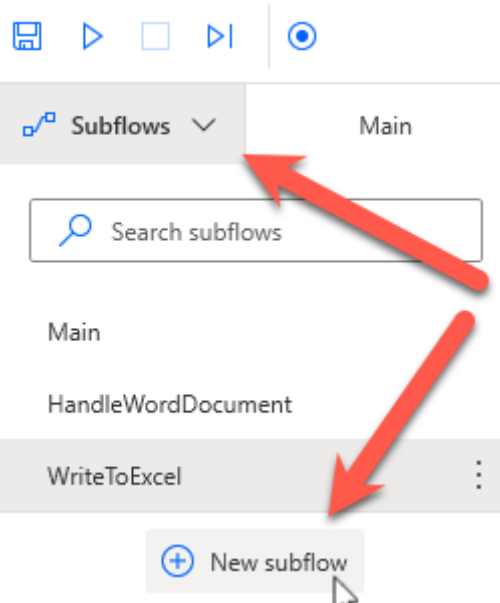
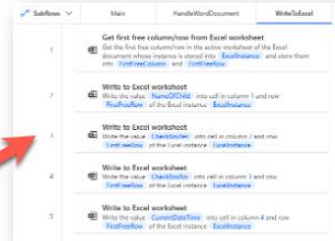
## Main Flow










## HandleWordDocument Subflow

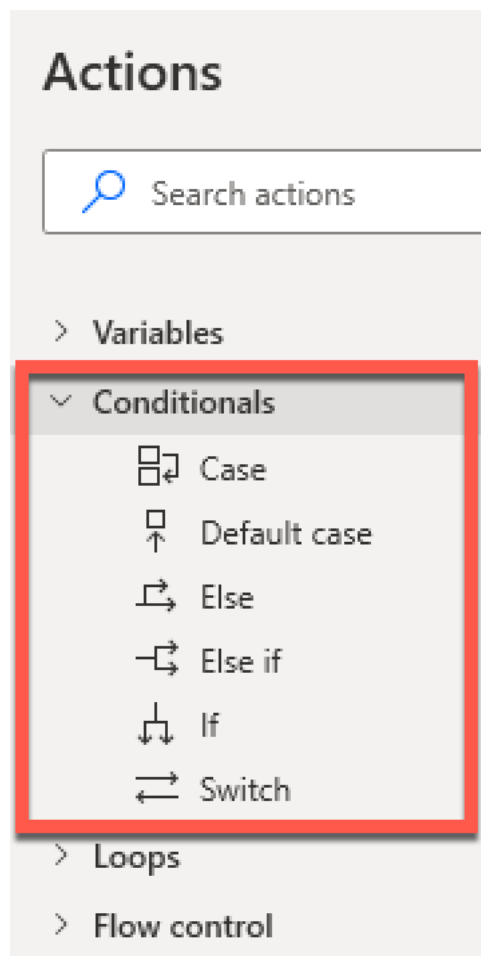


## WriteToExcel Subflow




Subflows ▾	Main	HandleWordDocument	WriteToExcel
1	<div>  <b>Launch Outlook</b>            Launch Outlook and store the instance created in <code>OutlookInstance</code> </div>		
2	<div>  <b>Retrieve email messages from Outlook</b>            Retrieve <code>Unread email messages only</code> from account <code>OutlookAccount</code> and store them in <code>RetrievedEmails</code> </div>		
3	<div>  <b>Process email messages in Outlook</b>            Move email messages <code>RetrievedEmails</code> to mail folder in account <code>OutlookAccount</code> </div>		
4	<div>  <b>Close Outlook</b>            Close the Outlook instance stored into <code>OutlookInstance</code> </div>		
5	<div>  <b>Launch Excel</b>            Launch Excel and open document <code>SaveAttachmentsFolder</code> <code>'Activity Tracker.xlsx'</code> using an existing Excel process and store it into Excel instance <code>ExcelInstance</code> </div>		
6	<div>  <b>Get files in folder</b>            Retrieve the files in folder <code>SaveAttachmentsFolder</code> that match <code>'Parents*.docx'</code> and store them into <code>Files</code> </div>		
7	<div>  <b>Get current date and time</b>            Retrieve the current datetime and store it into <code>CurrentDateTime</code> </div>		

## Chapter 4: Basic Structure Elements and Flow Control



If



 Marks the beginning of a block of actions that is run if the condition specified in this statement is met  
[More info](#)

Select parameters

First operand:

{x}



Operator:

Equal to (=)



Second operand:

{x}



Save

Cancel

4




Subflows



Main



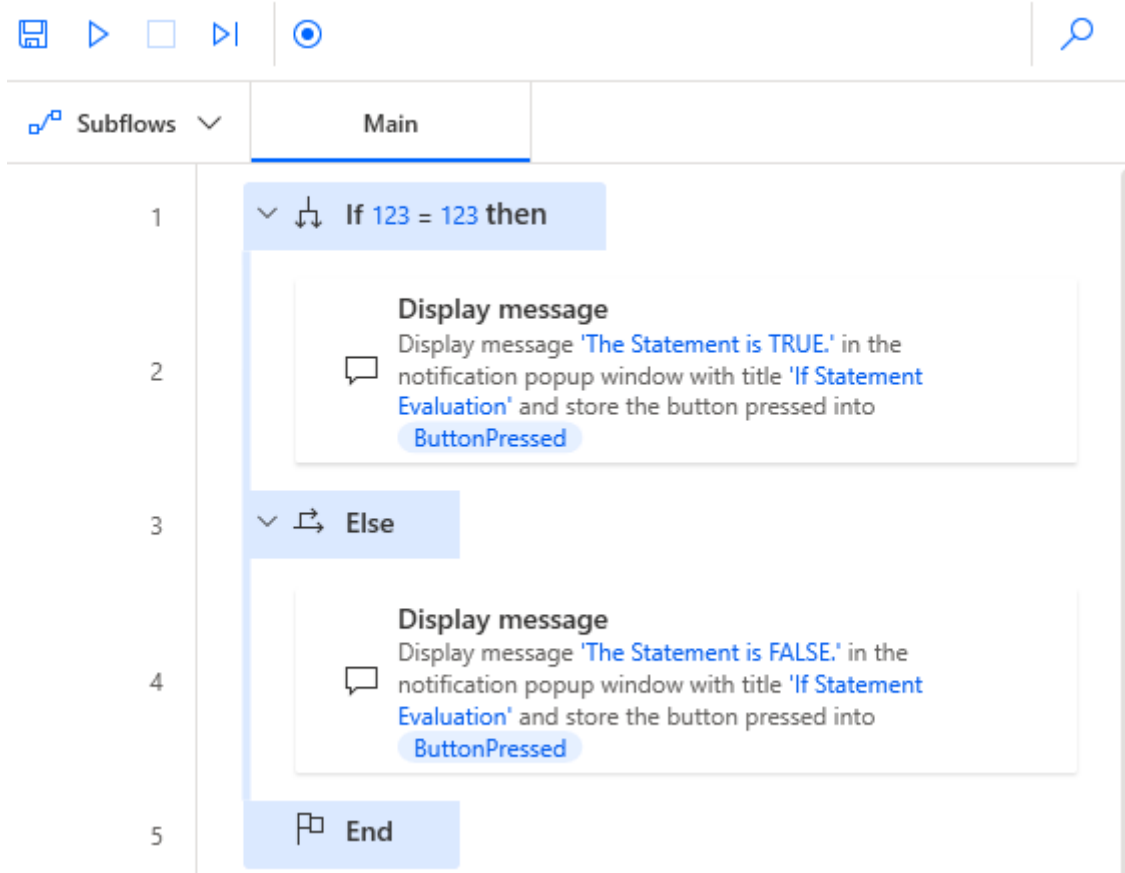
1

 If 123 = 123 then

2

 End





### Variables

Search variables

Input / output variables 5


- (x) NumberVar1 12345
- (x) NumberVar2 5
- (x) TextVar1 The quick brown fox
- (x) TextVar2 fox
- (x) TextVar3 12345


Input

Output

## Edit input variable




 Edit the properties of an existing input or output variable [More info](#)

Variable name:  

Data type:  

Default value:  

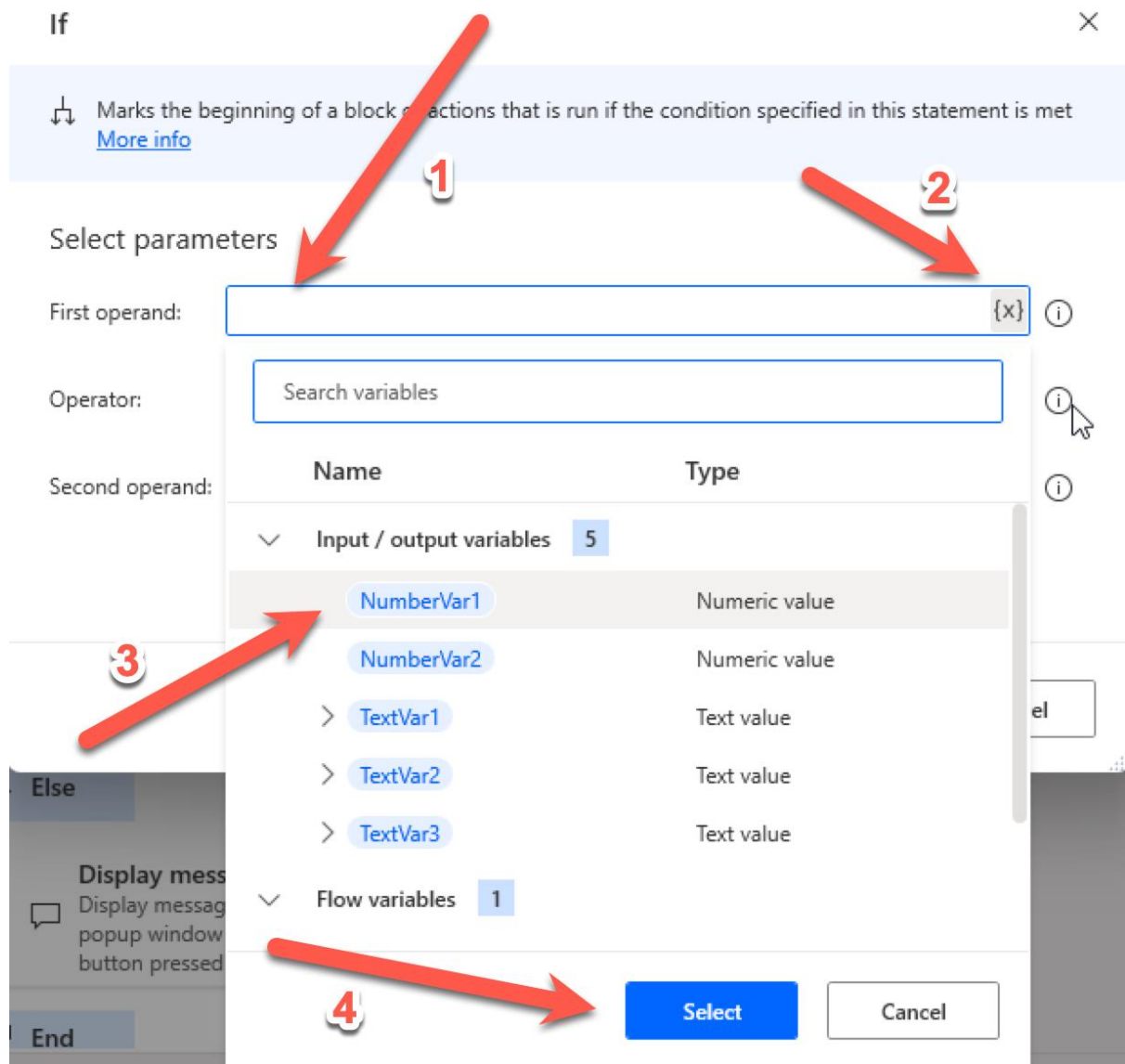
External name:  

Description:  

Mark as sensitive ☐ 

Save

Cancel



If



Marks the beginning of a block of actions that is run if the condition specified in this statement is met  
[More info](#)

Select parameters

First operand:  {x} ⓘ

Operator:  ⓘ

Second operand:  {x} ⓘ

Save

Cancel

▼ If TeamGameStatusWon = 'True' then

▼ If GoalDifference >= 5 then

Display message

Display message 'What a match !! Special thanks to ' ManOfTheMatch in the notification popup window with title 'Great match' and store the button pressed into ButtonPressed

▼ Else

Display message

Display message 'Very nice match !! Thanks to ' ManOfTheMatch in the notification popup window with title 'Nice match' and store the button pressed into ButtonPressed

End

▼ Else if NumberOfFouls <= 4 then

Display message

Display message 'Hard match !! Thanks to ' ManOfTheMatch in the notification popup window with title 'You lost, but very fair !!' and store the button pressed into ButtonPressed

▼ Else

Display message

Display message 'Try again next time...' in the notification popup window with title 'No worries...' and store the button pressed into ButtonPressed

End

▼ Input / output variables 4

(x) GoalDifference 5

(x) ManOfTheMatch Wayne Gretzky

(x) NumberOfFouls 4

(x) TeamGameStat... True

▼ Flow variables 1

(x) ButtonPressed OK

**Main**

Switch **StudentScorePercentage**

- Case  $\geq 90$ 
  - Display message  
Display message 'Well done!' in the notification popup window with title 'Grade A' and store the button pressed into **ButtonPressed**
- Case  $\geq 80$ 
  - Display message  
Display message 'Not bad!' in the notification popup window with title 'Grade B' and store the button pressed into **ButtonPressed**
- Case  $\geq 70$ 
  - Display message  
Display message 'Good !' in the notification popup window with title 'Grade C' and store the button pressed into **ButtonPressed**
- Default case
  - Display message  
Display message 'Try again next time !' in the notification popup window with title 'No Grade' and store the button pressed into **ButtonPressed**

End

Search variables

Input / output variables 1

- StudentScoreP...** 85

Flow variables 1

- ButtonPressed** OK

**Main**

Loop **LoopIndex** from 1 to 3 with step 1

- Generate random number  
Generate a random number between 1 and 6 and store it into **DiceResult**
- If **DiceResult** = 6 then
  - Display message  
Display message 'We have a 6 !!' in the notification popup window with title 'Yeah !!' and store the button pressed into **ButtonPressed**
  - Exit loop
- Else
  - Display message  
Display message 'It is a ' **DiceResult** ', We need to try again....' in the notification popup window with title 'Oh no!!' and store the button pressed into **ButtonPressed**

End

End

Search variables

Input / output variables 0

There aren't any input or output variables here yet

Flow variables 3

- ButtonPressed** OK
- DiceResult** 5
- LoopIndex** 4

**Generate random number**  
Generate a random number between 1 and 20 and store it into **RandomNumber** 1

**Loop condition** While ( **GuessRight** ) = ('False') 2

**Display input dialog**  
Display input dialog with message 'Enter your guess' in the notification popup window with title 'Enter your guess' and store the user's input into **UserInput** and button pressed into **ButtonPressed** 3

**If RandomNumber = UserInput then**

- Set variable**  
(x) Assign to variable **GuessRight** the value 'True' 4
- Display message**  
Display message 'You did it !!' in the notification popup window with title 'Congratulations' and store the button pressed into **ButtonPressed2**

**Else**

- Increase variable**  
Increase variable **Counter** by 1 5
- Display message**  
Display message 'You failed ' **Counter** ' times' in the notification popup window with title 'No luck' and store the button pressed into **ButtonPressed2**

**End**

**End**

**Input / output variables** 2

- (x) **Counter** 3
- (x) **GuessRight** True

**Flow variables** 4

- (x) **ButtonPressed** OK
- (x) **ButtonPressed2** OK
- (x) **RandomNumb...** 17
- (x) **UserInput** 17

## Set variable



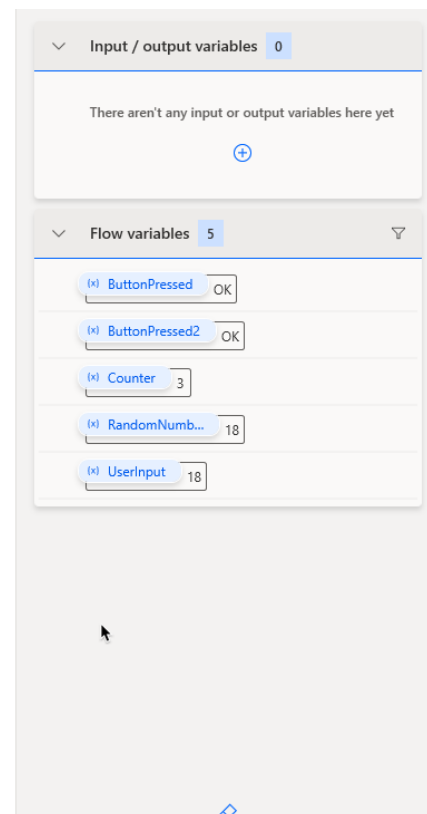
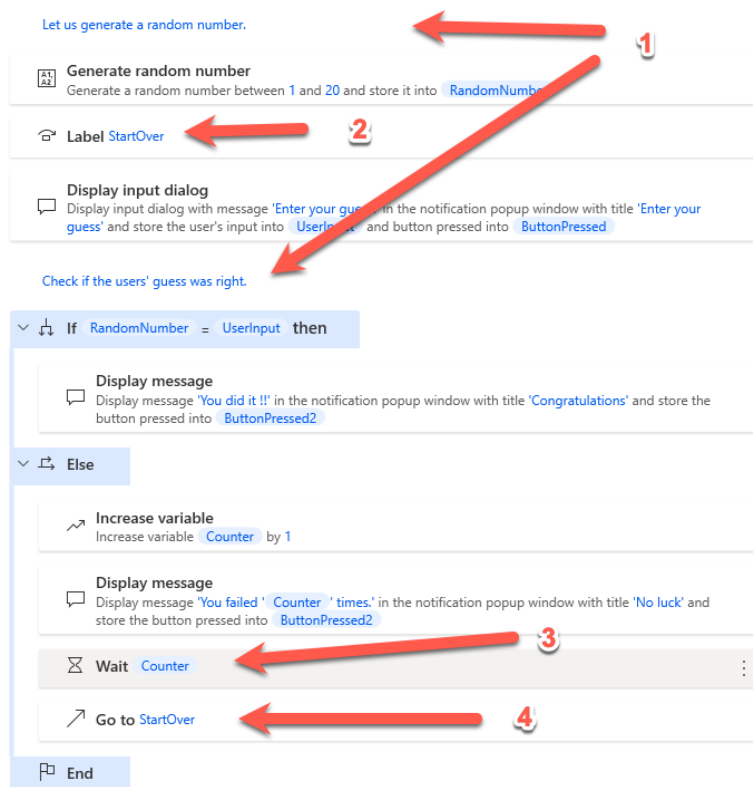
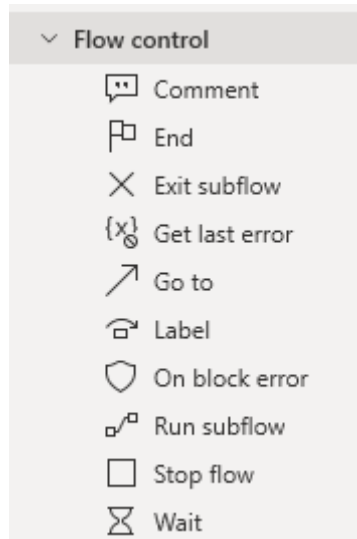
{x} Set the value of a new or existing variable, create a new variable or overwrite a previously created variable [More info](#)

Variable: **GuessRight** {x}

Value:  {x} ⓘ

Save

Cancel



Delete file(s)

The following rules will apply if the action fails [More info](#)

Retry action if an error occurs

Times

1

Interval

2

sec

✓ All errors

+

New rule

×

Clear all

No rules added

Continue flow run

Throw error

> All errors

← Return to parameters

Save

Cancel

Delete file(s)

The following rules will apply if the action fails [More info](#)

Retry action if an error occurs

Times

1

Interval

2

sec

✓ All errors

+

New rule

×

Clear all

Continue flow run

Throw error

Exception handling mode

Go to next action

> All errors

← Return to parameters

Save

Cancel



## Delete file(s)



The following rules will apply if the action fails [More info](#)

Retry action if an error occurs ☐

Times  Interval  sec

### ✓ All errors

New rule Clear all

Variable NewVar {x} to  {x}

Continue flow run

Throw error

Exception handling mode

Go to next action

### ✓ Advanced

File path contains invalid characters

New rule Clear all

File not found

New rule Clear all

Can't delete file

New rule Clear all

Return to parameters

Save

Cancel

Subflows

Main

1

Delete file(s)  
Delete file(s) `FileToDelete`

2

{x}

Get last error

Get the last error that occurred and store it into `LastError` without clearing the error value.

3

Display message

Display message 'This is the end of the flow. Last error was: ' `LastError` ' in the notification popup window with title 'End of flow' and store the button pressed into `ButtonPressed`

Subflows

Main

1

On block error `TryFileDeletion`

2

Delete file(s)

Delete file(s) `FileToDelete`

3

End

4

{x}

Get last error

Get the last error that occurred and store it into `LastError` without clearing the error value.

5

Display message

Display message 'This is the end of the flow. Last error was: ' `LastError` ' in the notification popup window with title 'End of flow' and store the button pressed into `ButtonPressed`

Search variables

Input / output variables 1

FileToDelete

c:\deletethis.txt

Flow variables 3

ButtonPressed

LastError

NewVar

Search variables

Input / output variables 1

FileToDelete

c:\deletethis.txt

Flow variables 2

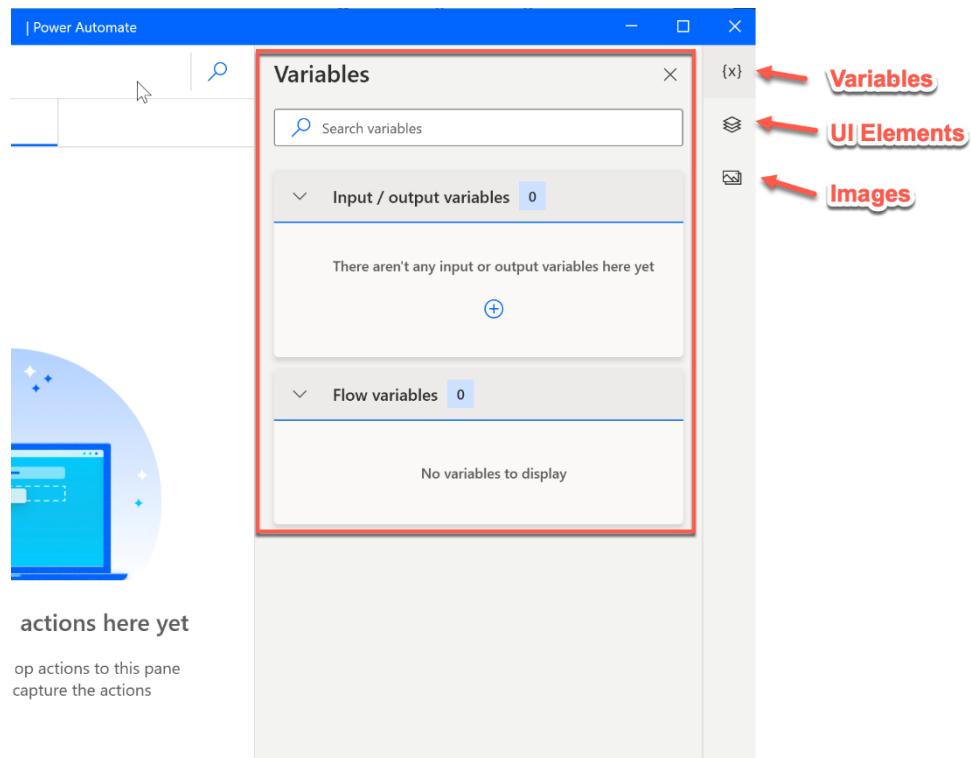
ButtonPressed

OK

LastError

File to delete not fo...

## Chapter 5: Variables, UI Elements, and Images



Variables

Search variables

Input / output variables0

There aren't any input or output variables here yet

Flow variables2

ButtonPressedOK

Cars [Rolls Royce, Renaul...]

Variable value

Cars (List of Text values)

#	Item
0	Rolls Royce
1	Renault
2	BMW
3	Toyota
4	Fiat
5	Mercedes
6	Ferrari
7	Citroen
8	Aston Martin
9	Chrysler

Close

Variables

Search variables

Input / output variables0

There aren't any input or output variables here yet

Flow variables2

ButtonPressed

Cars

Display message

×

Displays a message box

More info

Message box title:

Welcome %username%

{x}

i

Message to display:

The result is: %output1%

{x}

i

Message box icon:

None

▼

i

Message box buttons:

OK

▼

i

Default button:

First button

▼

i

Keep message box always on top:

☐

i

Close message box automatically:

☐

i

Variables produced

☒

ButtonPressed {x}

The text of the button pressed

On error

Save

Cancel

Power Automate

Peter Krause

×

+ New Flow

Settings ? Search Flows

Home

My flows

Examples

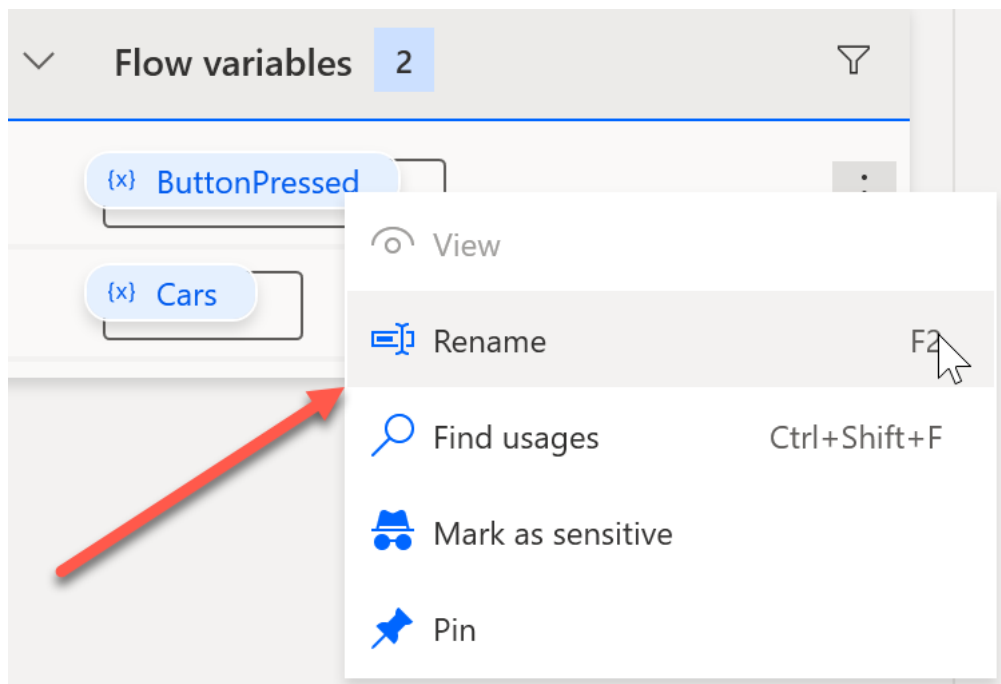
Example categories

**Excel Automation**  
Automate repetitive Excel tasks, extract and store data into Excel files, read and manipulate cells, and manage spreadsheets.  
5 examples

**Web Automation**  
Automate web applications, populate forms, extract structured data from web pages, and create reports using web data.  
3 examples

**Desktop Automation**  
Automate your daily desktop routine, launch applications, interact with Windows, and extract data for further processing in your flows.  
11 examples

**Datetime Handling**  
Handle date and time data, perform calculations using datetime values, and make conversions between datetime and text values.  
6 examples



## Variable name is already in use



Another variable with the same name already exists. Continuing will merge the two variables.

Continue

Cancel

## New input variable



⌵ Add a new variable to be used as input or output [More info](#)

Variable name:	<input type="text" value="NewInput"/>	
Data type:	<input type="text" value="Text"/>	
Default value:	<input type="text" value="Add a text value"/>	
External name:	<input type="text" value="NewInput"/>	
Description:	<input type="text" value="Add an input description"/>	
Mark as sensitive	<input type="checkbox"/>	

Save

Cancel

1

**Set variable**  
{x} Assign to variable **output1** the value **op1** + **op2**

2

**Display message**  
Display message 'The result is: ' **output1** ' in the notification popup window with title 'Welcome ' **username**

Input / output variables 4

**op1** 1

**op2** 1

**output1**

**username**

## Set variable



{x} Set the value of a new or existing variable, create a new variable or overwrite a previously created variable [More info](#)

Variable:	<input type="text" value="output1"/> {x}
Value:	<input type="text" value="%op1 + op2%"/> {x}

Save

Cancel

## Display message



Displays a message box [More info](#)

### Select parameters

Message box title:  {x} ⓘ

Message to display:  {x} ⓘ

## Flow inputs



With flow input variables you pass data to be used in the flow. Flow inputs are set and configured in the Power Automate designer. [More info](#)

Operand 1

123

1



Operand 2

123

1



Your Name

Abc

Add a text value

OK

Cancel

### Create new data table

Creates a new data table variable [More info](#)

#### Select parameters

New Table:  ⓘ

Variables produced: [DataTable](#)

Save

Cancel

### Edit datatable

	Column1	Column2
0		

Save

Cancel



	Manufacturer	Country	
0	BMW	Germany	
1	Aston Martin	United Kingdom	
2	Chrysler	USA	
3	Fiat	Italy	
4	Citroen	France	
5	Toyota	Japan	

Save

Cancel

The screenshot displays the Microsoft Power Automate interface for a flow named "Find and replace data in a table". The flow is configured with the following steps:

- Trigger:** Main (Manual trigger)
- Action 1:** Create new data table (Stores data into **CarsTable**)
- Action 2:** Insert row into data table (Adds the row **[Chevrolet]** at the end of **CarsTable**)
- Action 3:** Find or replace in data table (Finds all items in **CarsTable** containing the text **'USA'** and stores the matched cells into **DataTableMatches**)
- Action 4:** For each (Iterates over **CurrentItem** in **DataTableMatches**)
- Action 5:** Update data table item (Updates the data table **CarsTable** at row index **CurrentItem ['Row']** and column **CurrentItem ['Column']** with the value **'United States'**)

A red box highlights the "Find or replace in data table" and "For each" actions. A red arrow points from the "Find or replace in data table" action to the "Update data table item" action.

The "Variables" pane on the right shows the following variables:

- Input / output variables:** 0
- Flow variables:** 3
  - CarsTable** (Type: Table)
  - CurrentItem** (Type: Object)
  - DataTableMatc...** (Type: Table)

## Find or replace in data table



Finds and/or replaces data table values [More info](#)

### Select parameters

Data table:

Search mode:

All matches: ☒

Text to find:

Find using a regular expression: ☐

Match case: ☐

Match entire cell contents: ☐

Search by:

> Variables produced [DataTableMatches](#)

On error

Save

Cancel

## Update data table item



Update a data table row item on a defined column [More info](#)

### Select parameters

Data table:

Column:

Row:

New value:

### Variable value

[DataTableMatches](#) (Datatable)

#	Row	Column
0	2	1
1	6	1

Close

On error

Save

Cancel

### Variable value

CarsTable (Datatable)

#	Manufacturer	Country
0	BMW	Germany
1	Aston Martin	United Kingdom
2	Chrysler	USA
3	Fiat	Italy
4	Citroen	France
5	Toyota	Japan
6	Chevrolet	USA

### Variable value

CarsTable (Datatable)

#	Manufacturer	Country
0	BMW	Germany
1	Aston Martin	United Kingdom
2	Chrysler	United States
3	Fiat	Italy
4	Citroen	France
5	Toyota	Japan
6	Chevrolet	United States

#### New input variable

[-] Add a new variable to be used as input or output [More info](#)

Variable name:

Data type:

Default value:  [Edit](#)

External name:

Description:

Mark as sensitive ☐

[Save](#) [Cancel](#)

#### Edit custom object

JSON editor ☐

Name	Value
Brand	BMW
Color	blue
year of manufacture	2020
type of vehicle	SUV

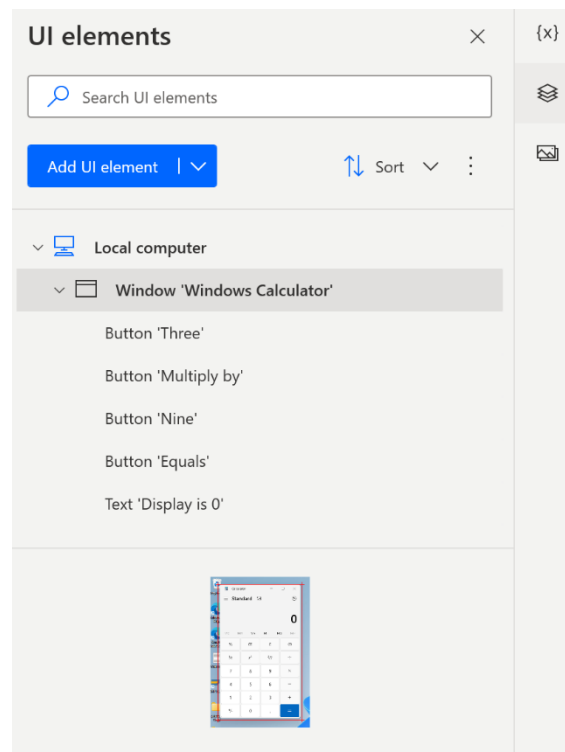
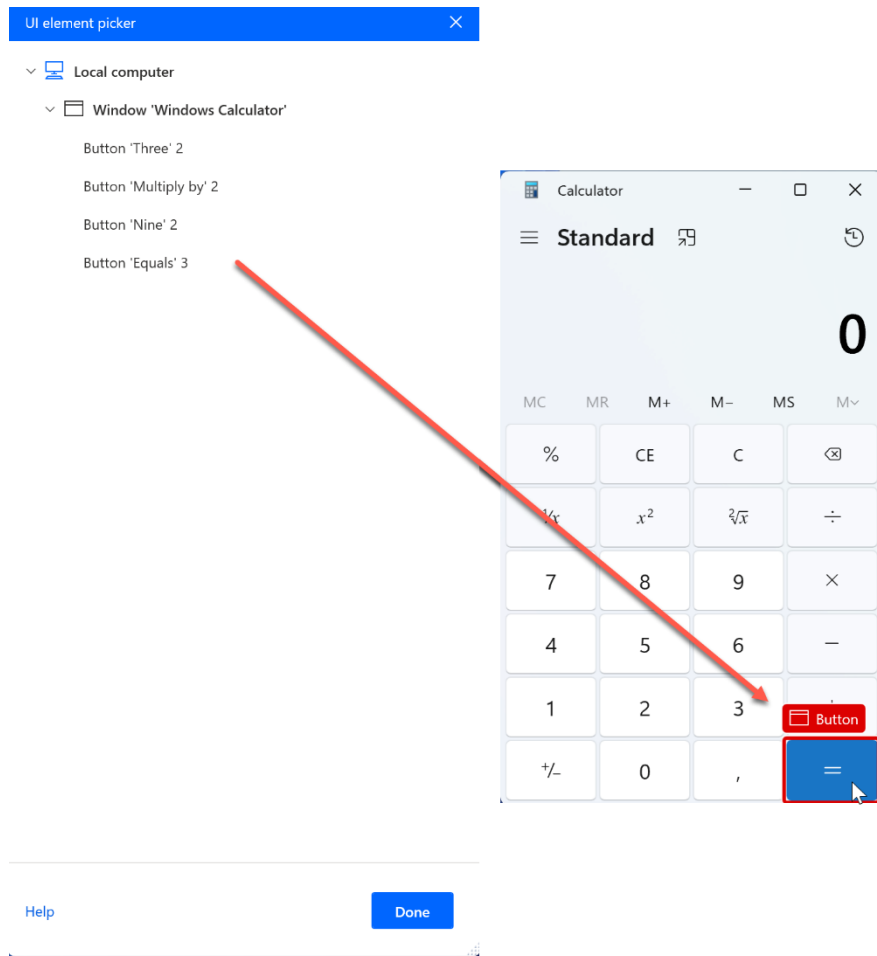
[Save](#) [Cancel](#)

#### Edit custom object

JSON editor ☒

```
1 { 'Brand': 'BMW', 'Color': 'blue', 'year of manufacture': 2020, 'type of vehicle': 'SUV' }
```

[Save](#) [Cancel](#)



UI elements

Search UI elements

Add UI element | Sort

Local computer

Window 'Windows Calculator'

- Button 'Three'
- Button 'Multiply by'
- Button 'Nine'
- Button 'Equals'
- Text 'Display is 0'

Subflows

Main

- Press button in window  
Press button Button 'Three'
- Press button in window  
Press button Button 'Multiply by'
- Press button in window  
Press button Button 'Nine'
- Press button in window  
Press button Button 'Equals'
- Get details of the UI element in window  
Get attribute 'Own Text' of UI element Text 'Display is 0' and store it into AttributeValue

Selectors of UI element "Button 'Three'"

Selector with recapture

Default Selector

Elements

- 1 Window 'Calculator'
- 2 UI Custom
- 3 Group 'LandmarkTarget'
- 4 Group 'Number pad'
- 5 Button 'Three'

Attribute	Operator	Value
<input checked="" type="checkbox"/> Class	Equal to	Windows.UI.Core.CoreWindow
<input type="checkbox"/> Enabled	Equal to	True
<input type="checkbox"/> Id	Equal to	
<input checked="" type="checkbox"/> Name	Equal to	Calculator
<input type="checkbox"/> Ordinal	Equal to	-1
<input type="checkbox"/> Process	Equal to	CalculatorApp
<input type="checkbox"/> Visible	Equal to	True

Text editor

Preview Selector

```
> window[Class="Windows.UI.Core.CoreWindow"][Name="Calculator"] > custom[Id="NavigationView"] > group[Class="LandmarkTarget"]  
> group[Class="NamedContainerAutomationPeer"][Id="NumberPad"] > button[Class="Button"][Id="num3Button"]
```

Save Cancel



## Move mouse to image



 Moves the mouse over an image found on screen or on the foreground window [More info](#)

### Select parameters

Image to move mouse to

 Activities



 Select image(s)

Mouse movement style: With animation (high speed)

Occurrence: 1

Send a click after moving mouse: ☒

Click type: Left click

#### Advanced

Wait for image to appear: ☒

Fail timeout: 5

Seconds before click: 0

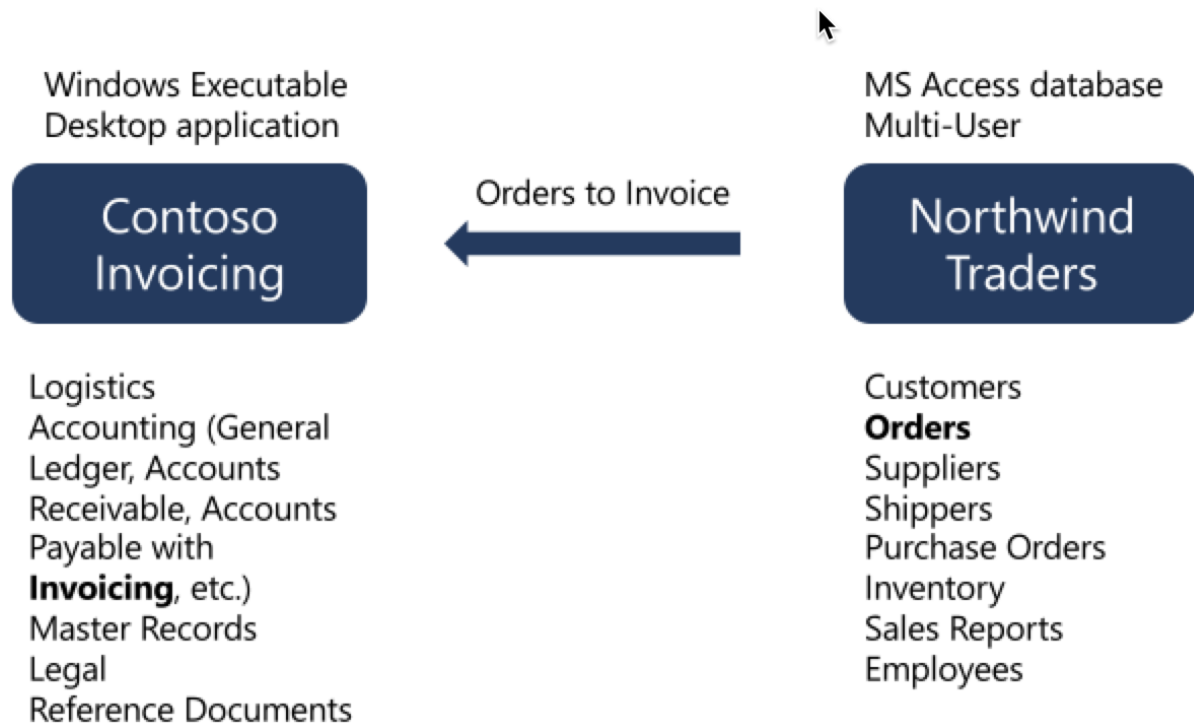
Image matching algorithm: Advanced

 On error

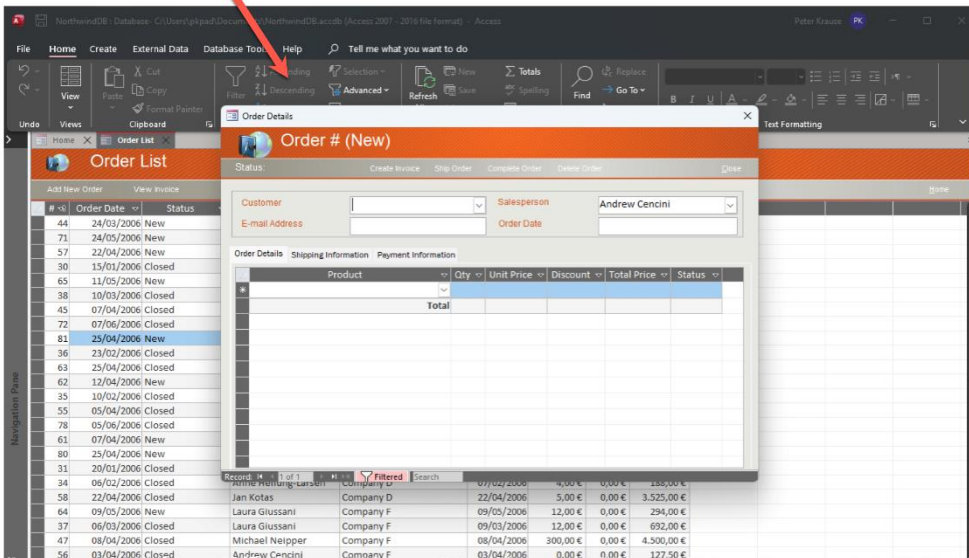
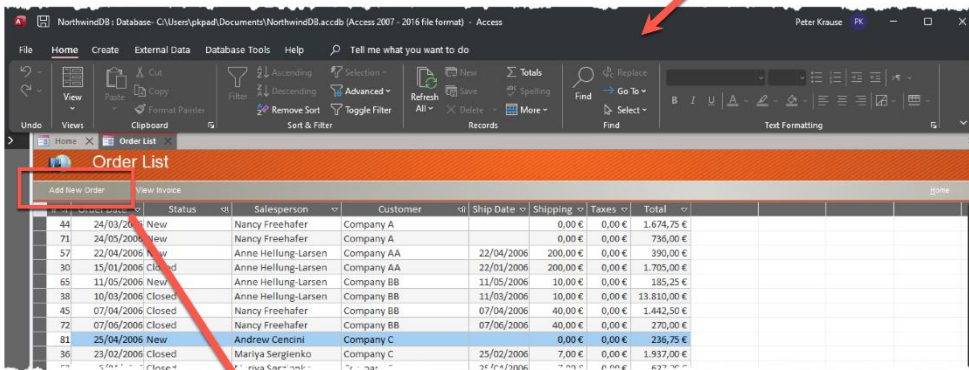
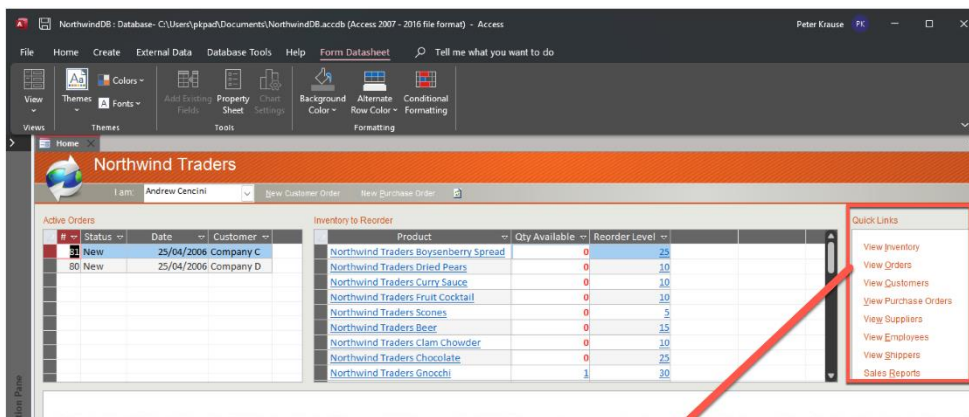
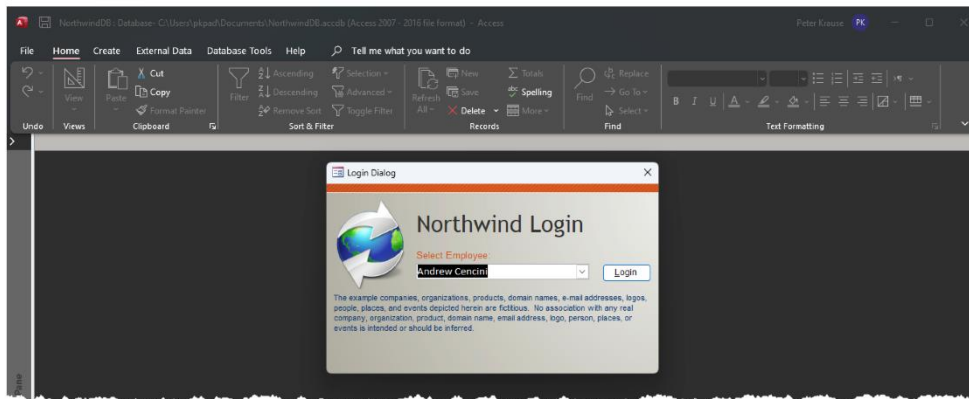
Save

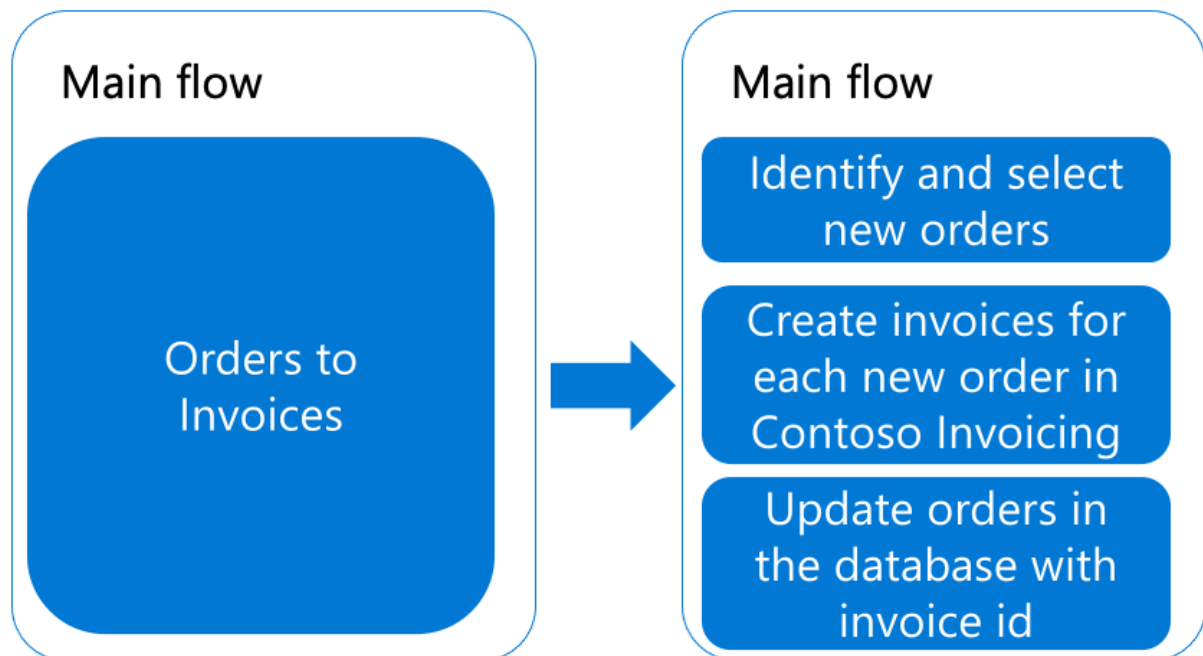
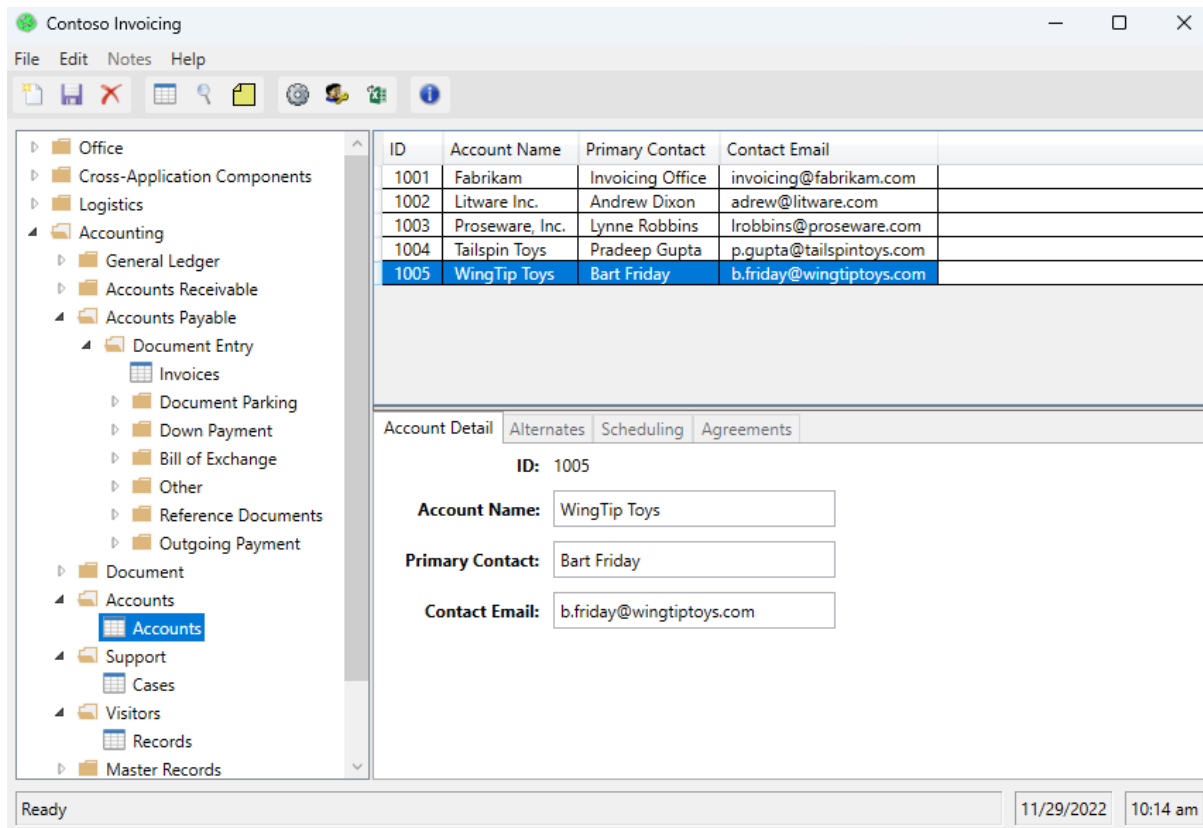
Cancel






## Chapter 6: Actions for UI Automation














Subflows ▾

Main

ExtractNewOrders

GetNorthW

← →

1

Run subflow [GetNorthWindNewOrders](#)

2






Run subflow [ExtractNewOrders](#)


3

Run subflow [InvoiceOrders](#)

4

Run subflow [WriteBackInvoiceIDs](#)





Subflows ▾

Main

ExtractNewOrders

GetNorthWindNewOr...


InvoiceOrders

WriteBackInvoiceIDs


1

Start MS Access with NorthWind DB


2

 Run application  
Run application [MSAccessApp](#) with arguments [NorthWindDB](#) and store its process ID into [MSAccessProcessID](#)


3

 Wait for window  
Wait for window [Window 'Login Dialog'](#) to open


4

 Click UI element in window  
Click on UI element [Button 'Login'](#)

5

 Wait for window  
Wait for window [Pane 'Home'](#) to open


6

 Move mouse to image  
Move the mouse pointer on the [center](#) of the image found on the [Entire screen](#) and send a [Left click](#) on it


7

This shows only new orders in the list


8

 Click UI element in window  
Click on UI element [Button 'Toggle Filter'](#)


9

 Click UI element in window  
Click on UI element [MenuItem: Select](#)

10

 Click UI element in window  
Click on UI element [MenuItem: Select → Select All](#)

11

 Click UI element in window  
Click on UI element [Button 'Copy'](#)

Save

Run

Stop

Run next action

Recorder

Subflows

Main

ExtractNewOrders

GetNorthWindNewOrd...

InvoiceOrders

WriteBackInvoiceIDs

1

Process new Orders from the database

2

Get clipboard text

Get the text from the clipboard and store it into ClipboardText

3

Split text

Split text ClipboardText by separating text elements with delimiter New line x 1 and store them into list NewOrdersTextList

4

Create new data table

Create a new data table and store it into NewOrders

5

For each

CurrentOrderText

in

NewOrdersTextList

6

Split text

Split text CurrentOrderText by separating text elements with delimiter Tab x 1 and store them into list CurrentOrderItemList

7

Set variable

{x} Assign to variable NewOrders the value NewOrders + [CurrentOrderItemList[0], CurrentOrderItemList[1], CurrentOrderItemList[2], CurrentOrderItemList[3], CurrentOrderItemList[4], CurrentOrderItemList[5], CurrentOrderItemList[6], CurrentOrderItemList[7], CurrentOrderItemList[8], ""]

8

End

9

Delete the first 2 rows from the table (empty first row and header)

10

Delete row from data table

Deletes the row with index 0 from data table NewOrders

11

Delete row from data table

Deletes the row with index 0 from data table NewOrders

Edit datatable

	OrderNumber	OrderDate	Status	SalesPerson	Customer	ShipDate	Shipping	Taxes	Total	InvoiceID	
0											



Save

Cancel

Contoso Invoicing

File Edit Notes Help

Icons: New, Open, Save, Print, Find, Recent, Help, etc.

Left Navigation Panel:

- Office
- Cross-Application Components
- Logistics
- Accounting
  - General Ledger
  - Accounts Receivable
  - Accounts Payable
    - Invoices**
    - Document Parking
    - Down Payment
    - Bill of Exchange
    - Other
    - Reference Documents
    - Outgoing Payment
  - Document
  - Accounts
    - Accounts
  - Support
    - Cases
  - Visitors
    - Records
  - Master Records

Main Table:

ID	Date	Account Name	Contact Email	Amount	Status
1043	12/4/2022	Company D	Andrew Cencini	\$ 380.00	Uninvoiced
1044	9/5/2006	Company F	Laura Giussani	\$ 294.00	Uninvoiced
1045	12/4/2022	Company G	Nancy Freehafer	\$ 13,800.00	Uninvoiced
1046	12/4/2022	Company G	Nancy Freehafer	\$ 2,949.00	Uninvoiced
1047	12/4/2022	Company H	Jan Kotas	\$ 5,555.00	Uninvoiced
1048	12/4/2022	Company J	Nancy Freehafer	\$ 525.00	Uninvoiced
1049	12/4/2022	Company K	Nancy Freehafer	\$ 2,195.00	Uninvoiced
1050	12/4/2022	Company K	Nancy Freehafer	\$ 800.00	Uninvoiced
1051	12/4/2022	Company L	Mariya Sergienko	\$ 2,753.00	Uninvoiced

Invoice Detail | Notes | Budgeting | Sales | Resources

**ID:** 1051

**Date:** 12/04/2022

**Account:** Company L

**Contact:** Mariya Sergienko

**Amount:** \$2,753.00

**Status:** Uninvoiced

Ready 12/4/2022 2:20 pm

Save

Run

Stop

Run next action

Recorder

Subflows

Main

ExtractNewOrders

GetNorthWindNewOrd...

InvoiceOrders

WriteBackInvoiceIDs

1

▶

Run application

Run application `ContosoInvoiceApp` with arguments and store its process ID into `ContosoInvoiceAppID`

2

⌵

Wait for window

Wait for window `Window 'Contoso Invoicing'` to open

3

🖱️

Click UI element in window

Click on UI element `Treeviewbutton 'Invoices'`

4

↺

Loop

`LoopIndex` from 0 to `NewOrders.RowCount - 1` with step 1

5

Click the 'New' Button

⋮

6

🖱️

Click UI element in window

Click on UI element `Button 'New'`

7

Fill in Date, Account, Contact and Amount

8

📄

Populate text field in window

Populate text box `Textbox 'Date'` with `NewOrders [ LoopIndex ][1]`

9

📄

Populate text field in window

Populate text box `Textbox 'Account'` with `NewOrders [ LoopIndex ][4]`

10

📄

Populate text field in window

Populate text box `Textbox 'Contact'` with `NewOrders [ LoopIndex ][3]`

11

📄

Recognize entities in text

Recognize all the `Currency` entities in text `NewOrders [ LoopIndex ][8]` and store the results into `RecognizedEntities`

12

{x}

Set variable

Assign to variable `AmountNumber` the value `RecognizedEntities [0][0]`

13

📄

Replace text

Replace text `'` with `'` in `AmountNumber` and store the result into `AmountNumber`

14

📄

Populate text field in window

Populate text box `Textbox 'Amount'` with `AmountNumber`

15

Extract InvoiceID

16

📄

Extract data from window

Extract all records from the UI element `Textbox 'InvoiceID'` and store the result(s) into `InvoiceID`

17

Fill in InvoiceID in Datatable

18

📄

Update data table item

Updates the data table `NewOrders` at row index `LoopIndex` and column 9 with the value `InvoiceID`

19

click the 'Save' button

20

🖱️

Click UI element in window

Click on UI element `Button 'Save'`

21

🏁

End

22

⊗

Terminate process

Terminate process with ID `ContosoInvoiceAppID`

## Recognize entities in text



Recognizes entities in text, such as numbers, units, date/time and others expressed in natural language across multiple languages [More info](#)

### Select parameters

Text to recognize from:

%NewOrders[LoopIndex][8]%

{x}



Entity type:

Currency



Language:

English

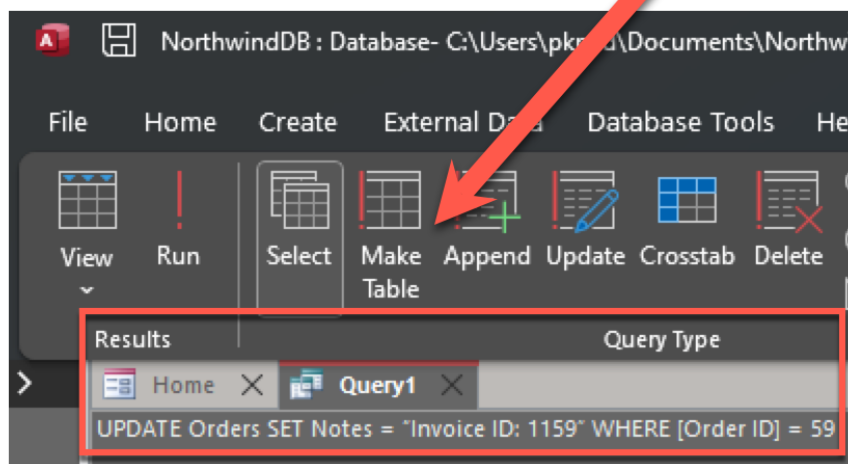
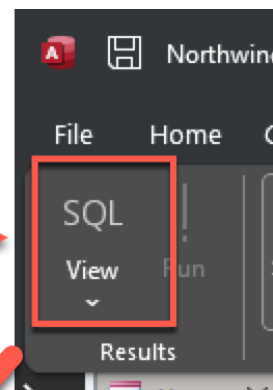
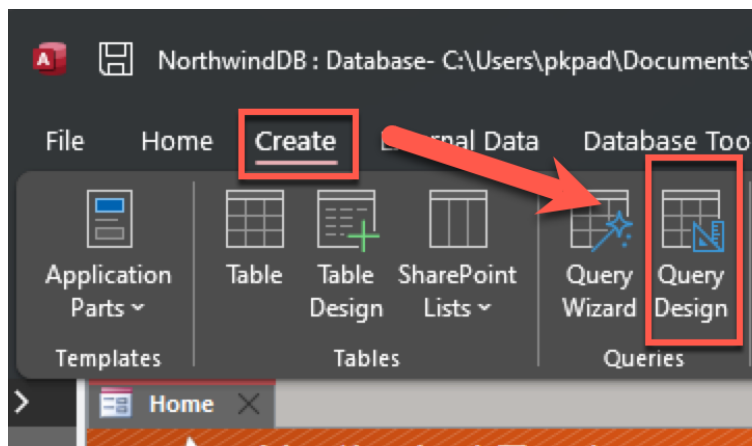


> Variables produced

RecognizedEntities

Save

Cancel





Subflows

ewOrd...

InvoiceOrders

WriteBackInvoiceIDs

1

For each CurrentItem in NewOrders

2

navigate to SQL input pane

3

Click UI element in window  
Click on UI element Tab Item 'Create'

4

Click UI element in window  
Click on UI element Button 'Query Design'

5

Click UI element in window  
Click on UI element MenuItem: More Options

6

Click UI element in window  
Click on UI element MenuItem: SQL View

7

create dynamic SQL statement

8

Populate text field in window  
Populate text box Pane 'Query1' with 'UPDATE Orders SET Notes = 'Invoice ID ' + CurrentItem [9] + ' WHERE [Order ID] = ' + CurrentItem [0]

9

Click UI element in window  
Click on UI element Button 'Run'

10

Wait for window  
Wait for window Window 'Microsoft Access' to open

11

close the query and start over

12

Click UI element in window  
Click on UI element Button 'Yes'

13

Click UI element in window  
Click on UI element Button 'NetUIDocumentTabCloseButton'

14

Wait for window  
Wait for window Window 'Microsoft Access' 2 to open

15

Click UI element in window  
Click on UI element Button 'No'

16

End

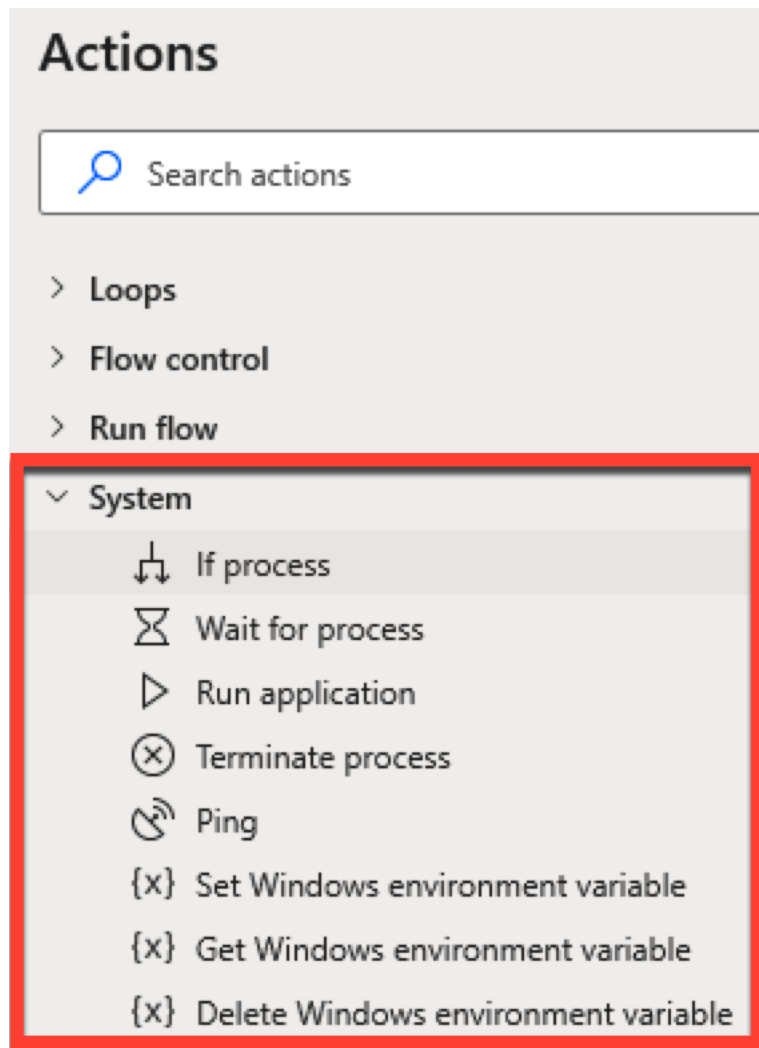
17

Close MS Access via UI action

18

Click UI element in window  
Click on UI element Button 'Close'

## Chapter 7: Automate Your Desktop and Workstation



## Run application



▶ Executes an application or opens a document by executing the associated application [More info](#)

### Select parameters

Application path:

"C:\Program Files\Microsoft Office\root\Office16  
WINWORD.EXE"



Command line arguments:

/t "C:\Users\pkpad\Documents\My New Document.docx"

{x}



Working folder:



Window style:

Normal



After application launch:

Continue immediately



> Variables produced **AppProcessId**

On error

Save

Cancel

## Actions



Windows



> System

> UI automation

▼ Windows services



If service



Wait for service



Start service



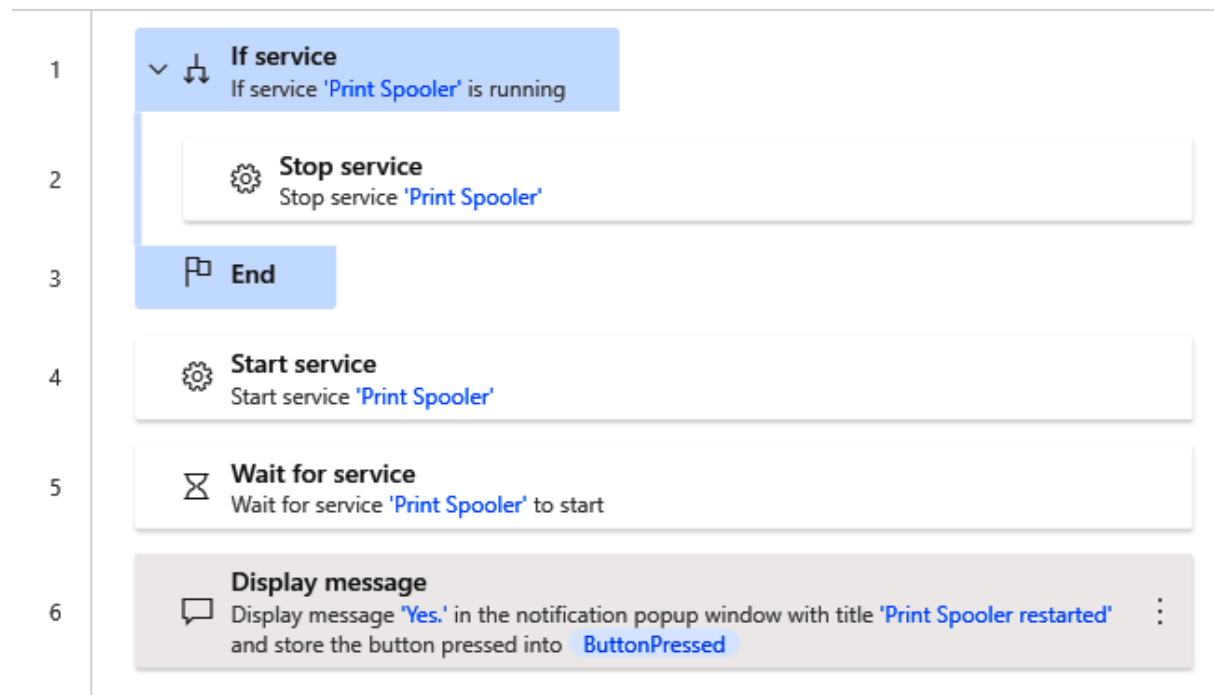
Stop service



Pause service



Resume service



## Stop service



⚙ Stop a running Windows service [More info](#)

### Select parameters

Service to stop:

Print Spooler

{x}



ActiveX Installer (AxInstSV)

Agent Activation Runtime\_3a371

AllJoyn Router Service

App Readiness

Application Identity

Application Information

Application Layer Gateway Service

AppX Deployment Service (AppXSVC)

Auto Time Zone Updater

AVCTP service

Background Intelligent Transfer Service

Background Tasks Infrastructure Service



On error

> Mouse and keyboard

> Clipboard

> Text

> Date time

> PDF

> CMD session

↗ Open CMD session

📄 Read from CMD session

📄 Write to CMD session

📄 Wait for text on CMD session

↙ Close CMD session

> Terminal emulation

> OCR

> Cryptography

> Windows services

1

#### Open CMD session

Start a new CMD session with working directory 'C:\Users\pkpad\Documents' and store it into

[CmdSession](#)

2

#### Write to CMD session

Execute the command 'net start' and then send Enter at CMD session [CmdSession](#)

3

#### Wait for text on CMD session

Wait for text 'The command completed successfully.' on CMD session [CmdSession](#)

4

#### Read from CMD session

Read output from CMD session [CmdSession](#) and store standard output to [CmdOutput](#) and store standard error to [CmdError](#)

5

#### Write text to file

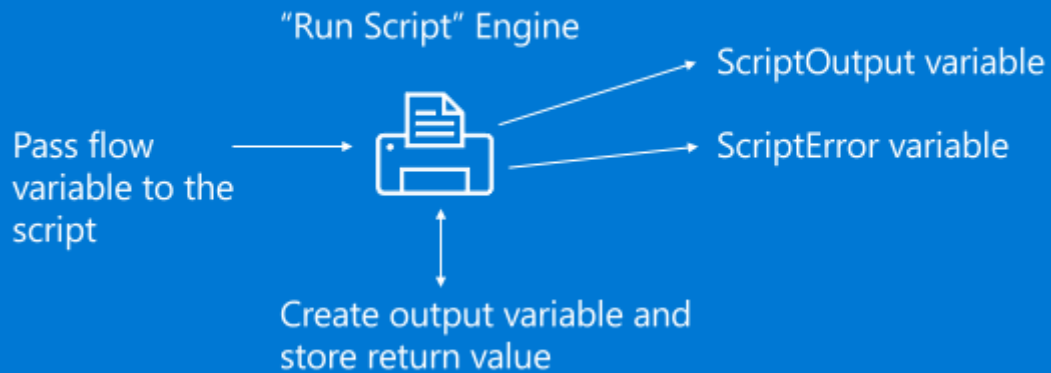
Write [CmdOutput](#) to 'C:\Users\pkpad\Desktop\directorycontent.txt'

6

#### Close CMD session

Close the CMD session [CmdSession](#)

## "Run Script" action

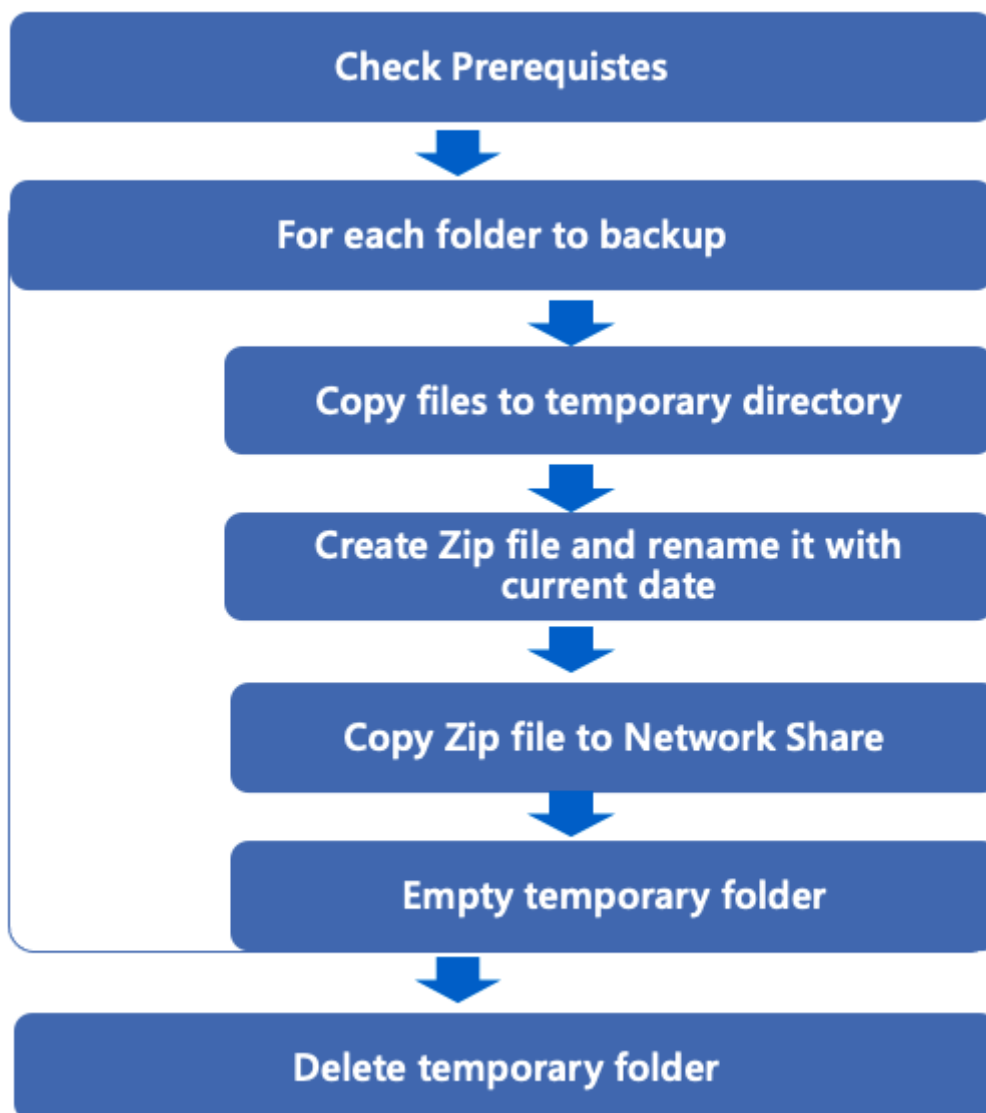


This screenshot shows a workflow editor with four steps: 1. Display input dialog, 2. Set variable (Assign to variable `InputValue` the value `UserInput`), 3. Run JavaScript (Run JavaScript and store its output into `JavascriptOutput` and its error into `ScriptError`), and 4. Display message. The "Run JavaScript" step is highlighted with a red box. A red arrow points from this step to a "Variables" panel on the right. The panel shows "Input / output variables" with `InputValue` (hello from pad) and "Flow variables" with `ButtonPressed` (OK), `JavascriptOutput...` (HELLO FROM PAD), `ScriptError`, and `UserInput` (hello from pad).

The "Run JavaScript" configuration dialog is shown. It includes a description: "Executes some custom JavaScript code and retrieves its output into a variable [More info](#)". Under "Select parameters", the "JavaScript to run:" field contains the following code: 

```
1 var inputValue1 = "%InputValue%";  
2 var outputVariable = inputValue1.toUpperCase();  
3 WScript.Echo(outputVariable);
```

 Below the code field, the "Variables produced" section lists `JavascriptOutput` and `ScriptError`. At the bottom, there are "Save" and "Cancel" buttons.

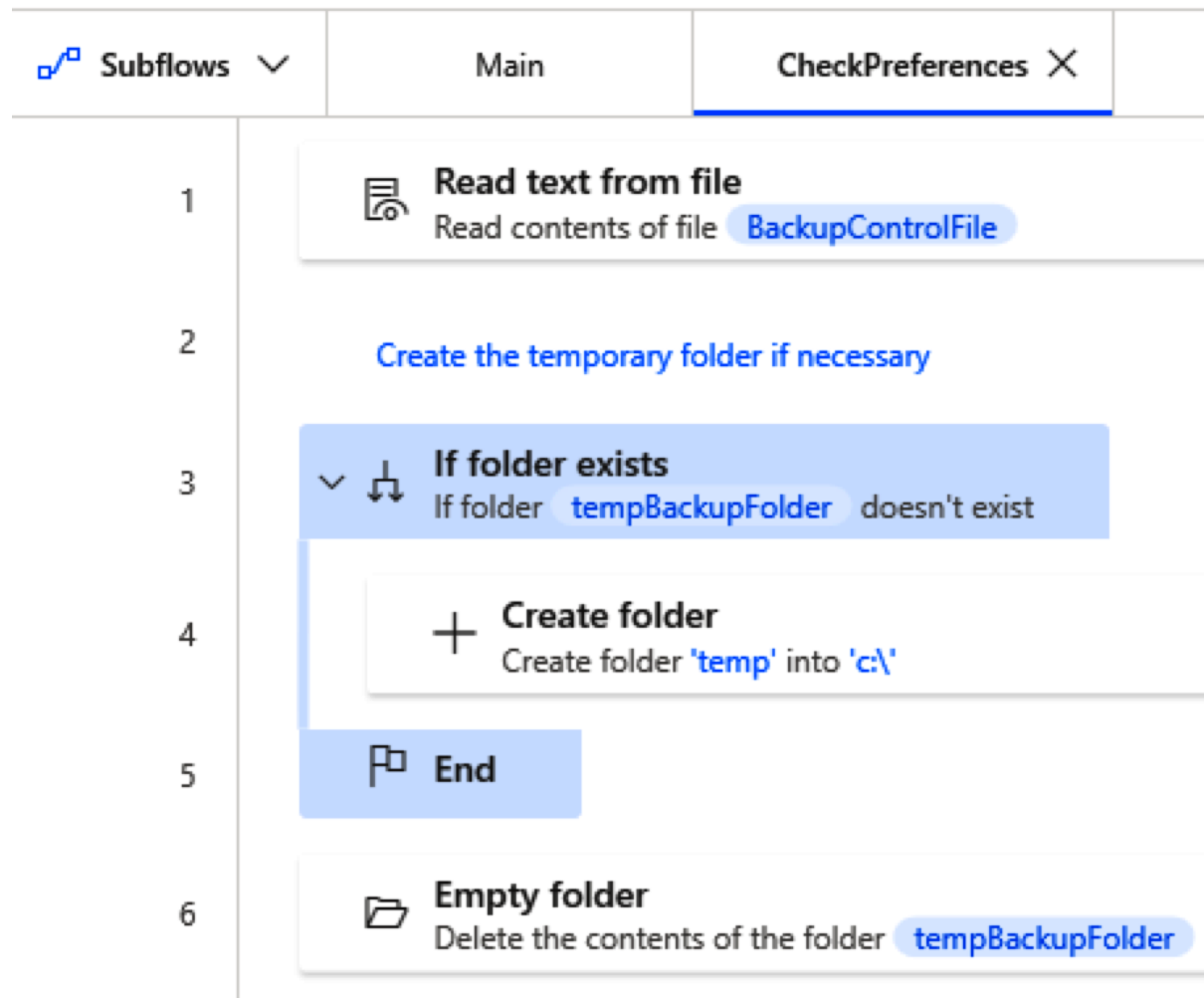


The screenshot shows a Notepad window with the title "DataToBackup.txt - Notepad". The menu bar includes "File", "Edit", and "View". The text content of the file is as follows:

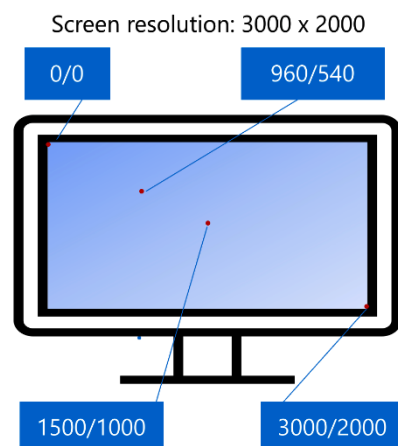
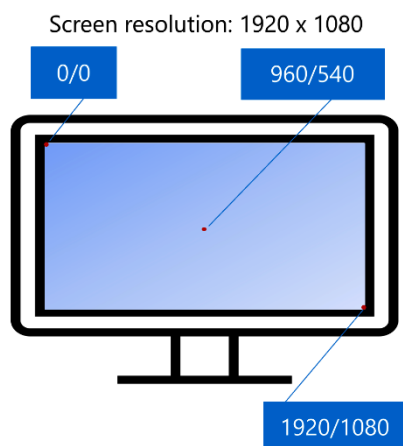
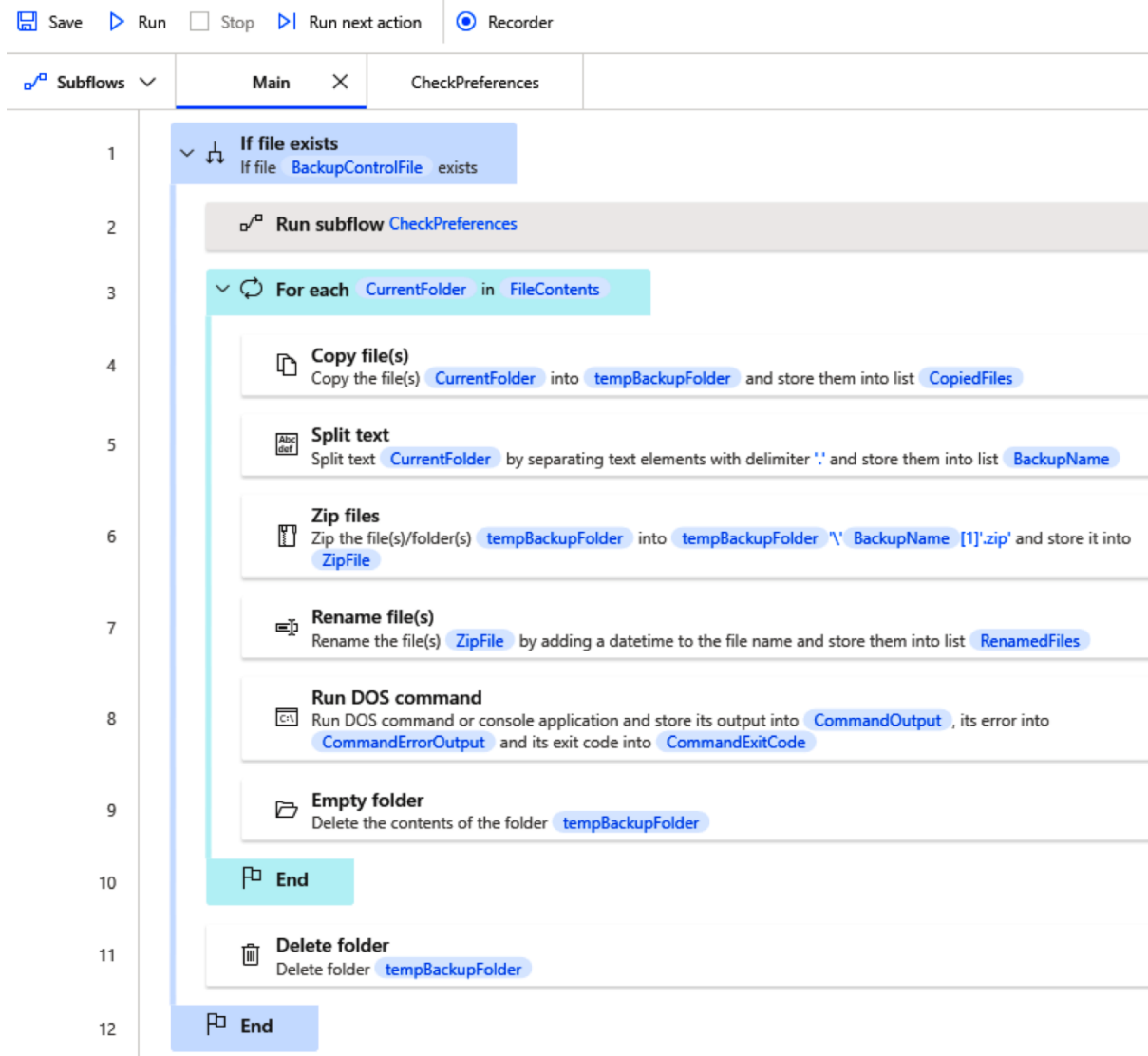
```
C:\Users\pkpad\Documents\*.accdb  
C:\Users\pkpad\Desktop\School Parents Association\*.docx
```

The status bar at the bottom indicates the cursor is at "Ln 1, Col 1", the zoom is "100%", the line endings are "Windows (CRLF)", and the encoding is "UTF-8".

 Save    Run    Stop    Run next action    Recorder








## Send keys



 Sends keys to the application that is currently active [More info](#)

### Select parameters

Send keys to:

Foreground window



Text to send:



Input as text, variable or expression

{x}



Insert special keys



Insert modifier




Delay between  
keystrokes:

10



Send Text as hardware keys:



 On error

Save

Cancel

Save Play Stop Step Through Search

Subflows Main LaunchPPT CreateSlide1

1	<b>Block Input</b> Block input from keyboard and mouse
2	<b>Run subflow</b> LaunchPPT
3	<b>Run subflow</b> CreateSlide1
4	<b>Run subflow</b> SystemInfoScreenshot
5	<b>Run subflow</b> CreateSlide2
6	<b>Run subflow</b> SaveAndClosePPT
7	<b>Block Input</b> Block input from keyboard and mouse

Save Play Stop Step Through Search

Subflows Main LaunchPPT X CreateSlide1

1	Hit the Windows button
2	<b>Send mouse click</b> Send a Left click event at 635, 951 (Screen)
3	Type "Powerpoint"
4	<b>Send keys</b> Send the following keystrokes: "Powerpoint" to the foreground window
5	Move the mouse and click the icon
6	<b>Send mouse click</b> Send a Left click event at 509, 402 (Screen)

Subflows Main LaunchPPT CreateSlide

- Click into the first slide and type some text
- Move mouse to image**  
Move the mouse pointer on the center of the image found on the Entire screen and send a Left click on it
- Get current date and time**  
Retrieve the current datetime and store it into CurrentDateTime
- Run PowerShell script**  
Run PowerShell script and store its output into PSHostname and its error into ScriptError
- Send keys**  
Send the following keystrokes: System Report CurrentDateTime to the foreground window

## Move mouse to image

Moves the mouse over an image found on screen or on the foreground window [More info](#)

### Select parameters

Image to move mouse to

PPTBlankPresentation

Blank Presentation

Select image(s)

Mouse movement style: Instant

Occurrence: 1

Send a click after moving mouse: ☒

Click type: Left click

> Advanced

> Variables produced

On error

Save

Cancel

## Run PowerShell script

Executes some custom PowerShell script and retrieves its output into a variable [More info](#)

### Select parameters

PowerShell code to run: 1 \$workstationname = hostname 2 Write-Output \$workstationname {x}

> Variables produced

PSHostname

ScriptError

On error

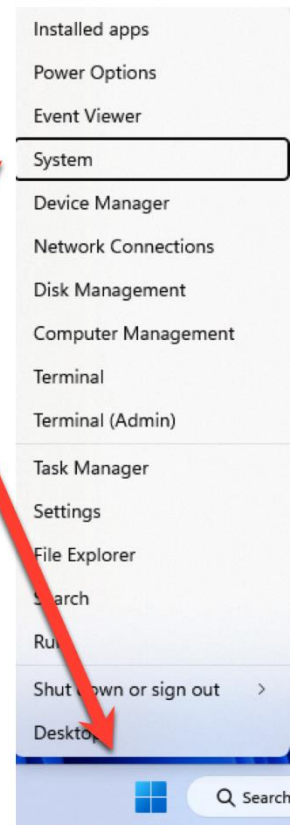
Save

Cancel

Subflows

SystemInfoScreenshot X CreateSlide2 SaveAndClosePPT

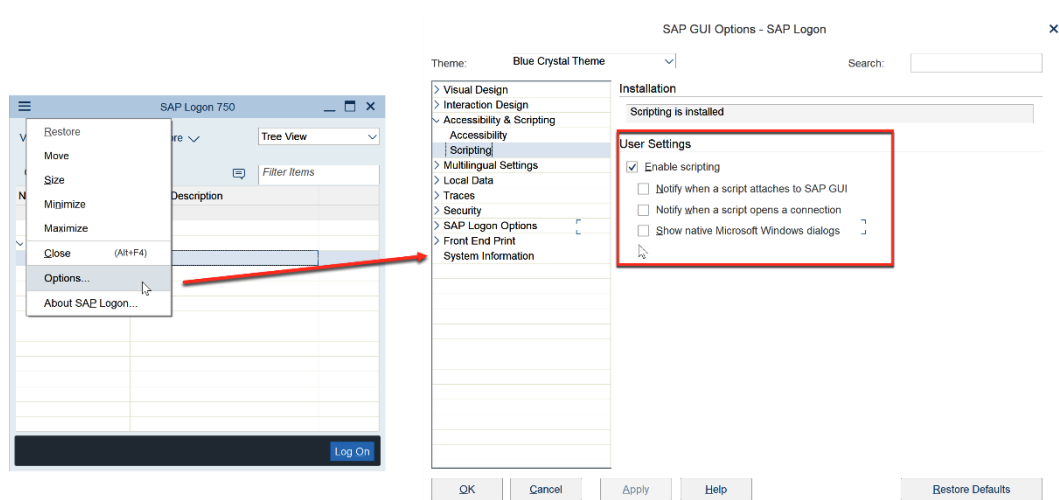
- 1 Launch "System" pane
- 2 Send mouse click  
Send a Right click event at 633, 954 (Screen)
- 3 Send keys  
Send the following keystrokes: '[Down:3]' to the foreground window
- 4 Send keys  
Send the following keystrokes: '[Return]' to the foreground window
- 5 Launch the System information window
- 6 Wait for image  
Wait for all images on the list to appear on the screen
- 7 Take a screenshot of the System information window
- 8 Take screenshot  
Take a screenshot of the foreground window and save it to the clipboard
- 9 Close the System information window
- 10 Send mouse click  
Send a Left click event at 1653, 76 (Screen)
- 11 Wait 2 seconds



Subflows		SystemInfoScreenshot	CreateSlide2	SaveAndClosePPT		
1	<b>Send keys</b> Send the following keystrokes: '{Control}{M}' to the foreground window					
2	<b>Send keys</b> Send the following keystrokes: '{Control}{V}' to the foreground window					
3	move image to the right					
4	<b>Send mouse click</b> Send a Left button down event at 711, 442 (Screen)					
5	<b>Send mouse click</b> Send a Left button up event at 880, 442 (Screen)					
6	Resize picture					
7	<b>Send mouse click</b> Send a Left button down event at 725, 210 (Screen)					
8	<b>Send mouse click</b> Send a Left button up event at 844, 334 (Screen)					
9	move image up					
10	<b>Send mouse click</b> Send a Left button down event at 1024, 493 (Screen)					
11	<b>Send mouse click</b> Send a Left button up event at 1024, 445 (Screen)					
12	click the title of the slide and save					
13	<b>Move mouse to image</b> Move the mouse pointer on the center of the image found on the Entire screen and send a Left click on it					
14	<b>Send keys</b> Send the following keystrokes: 'System Information' to the foreground window					



## Chapter 8: Automating Standard Business Applications





1

 **Run subflow** Launch\_SAPGUI


2

 **Run subflow** Launch\_Excel


3

 **Run subflow** Launch\_Outlook

4

 **Send keys**  
Send the following keystrokes: 'su01{Return}' to UI element Window 'SAP'

5

▼  **Loop** LoopIndex from 0 to ExcelData .RowCount - 1 with step 1

6

 **Run subflow** Read\_ExcelUserRow


7

 **Run subflow** Enter\_SAPUserData


8

 **Send email message through Outlook**  
Send email to 'peter.krause.x@outlook.com' with subject 'Your SAP credentials'


9

 **End**


10

 **Close Outlook**  
Close the Outlook instance stored into OutlookInstance


11








 **Close Excel**  
Save the Excel document and close the Excel instance ExcelInstance

12

 **Send keys**  
Send the following keystrokes: '{Alt}() {F4} {Tab} {Return}' to UI element Window 'SAP'

13

 **Terminate process**  
Terminate process with ID SAPGUIProcessId

1	 <b>Run application</b> Run application <code>"C:\Program Files (x86)\SAP\FrontEnd\SAPgui\saplogon.exe"</code> with arguments and store its process ID into <code>SAPGUIProcessId</code>
2	 <b>Wait for window</b> Wait for window <code>Window 'SAP Logon 750'</code> to open
3	 <b>Send keys</b> Send the following keystrokes: <code>{Return}</code> to UI element <code>Window 'SAP Logon 750'</code>
4	 <b>Wait for window</b> Wait for window <code>Window 'SAP'</code> to open
5	 <b>Send keys</b> Send the following keystrokes: <code>SAPUsername</code> <code>{Tab}</code> <code>SAPUserPassword</code> to UI element <code>Window 'SAP'</code>
6	 <b>Send keys</b> Send the following keystrokes: <code>{Return}</code> to UI element <code>Window 'SAP'</code>
7	 <b>Wait 2 seconds</b>



Monthly new Users.xlsx - Excel

Search

File Home Insert Page Layout Formulas Data Review View Help

Share

Undo Paste Clipboard Font Alignment Number Styles

Calibri 11

General Conditional Formatting Format as Table Cell Styles

Cells Editing

E8 bccoradini4\_alias

	A	B	C	D	E	F	G	H
1	User	First name	Last name	Email	Alias	initial Password		
2	sduer1	Sheridan	Duer	sduer1@house.gov	sduer1_alias			
3	sbrimmell2	Shoshanna	Brimmell	sbrimmell2@state.tx.us	sbrimmell2_alias			
4	cleonarde1	Candace	Leonarde	cleonarde0@answers.com	cleonarde1_alias			
5	sduer1	Sheridan	Duer	sduer1@house.gov	sduer1_alias			
6	sbrimmell2	Shoshanna	Brimmell	sbrimmell2@state.tx.us	sbrimmell2_alias			
7	kgiddings3	Karon	Giddings	kgiddings3@tinyurl.com	kgiddings3_alias			
8	bccoradini4	Brandice	Corradini	bccoradini4@ucla.edu	bccoradini4_alias			
9	cfann5	Cordi	Fann	cfann5@amazon.de	cfann5_alias			
10	wbusk6	Whit	Busk	wbusk6@purevolume.com	wbusk6_alias			
11	slighton7	Skip	Lighton	slighton7@list-manage.com	slighton7_alias			

Sheet1

Ready Accessibility: Good to go

- Launch Excel**

Launch Excel and open document 'C:\Users\pkpad\Documents\Monthly new Users.xlsx' using an existing Excel process and store it into Excel instance `ExcelInstance`
- Get first free row on column from Excel worksheet**

Get the first free row on column 4 in the active worksheet of the Excel document in instance `ExcelInstance`
- Read from Excel worksheet**

Read the values of the cells ranging from column 1 row 1 to column 5 row 1 and store it into `ExcelData`

## Send keys



 Sends keys to the application that is currently active [More info](#)

### Select parameters

Send keys to:

UI element



UI element:

appmask > Window 'SAP'



Text to send:

  Input as text, variable or expression {X}



su01{Return}

Insert special keys



Insert modifier




Delay between  
keystrokes:

10










Send Text as hardware keys:



 On error

Save

Cancel

Subflows ▾	Read_ExcelUserRow ✕	Enter_SAPUserData	Launch_Out	⏪ ⏩
1	<div>  <b>Create random text</b>            Create a random text of length between 8 and 10 characters using uppercase letters, lowercase letters, symbols and digits and store it in InitialPassword         </div>			
2	<div>  <b>Set variable</b>            Assign to variable Username the value ExcelData [ LoopIndex ][0]         </div>			
3	<div>  <b>Set variable</b>            Assign to variable UserFirstName the value ExcelData [ LoopIndex ][1]         </div>			
4	<div>  <b>Set variable</b>            Assign to variable UserLastName the value ExcelData [ LoopIndex ][2]         </div>			
5	<div>  <b>Set variable</b>            Assign to variable UserEmail the value ExcelData [ LoopIndex ][3]         </div>			
6	<div>  <b>Set variable</b>            Assign to variable UserAlias the value ExcelData [ LoopIndex ][4]         </div>			
7	<div>  <b>Write to Excel worksheet</b>            Write the value InitialPassword into cell in column 6 and row LoopIndex + 2 of the Excel instance ExcellInstance         </div>			

1



### Send keys

Send the following keystrokes: `UserName` `{Tab}` to UI element `Window 'SAP'`

2



### Send keys

Send the following keystrokes: `UserAlias` `{F8}` to UI element `Window 'SAP'`

7



### Send keys

Send the following keystrokes: `{Right}{Return}` to UI element `Window 'SAP'`

8



### Send keys

Send the following keystrokes: `UserAlias` `{Tab}{Tab}{Tab}{Tab}{Tab}{Tab}` to UI element `Window 'SAP'`

9



### Send keys

Send the following keystrokes: `InitialPassword` `{Tab}` `InitialPassword` `{Control}{S}` to UI element `Window 'SAP'`

## Send email message through Outlook



Create and send a new email message through Outlook [More info](#)

### Select parameters

Outlook instance:

%OutlookInstance%



Account:

pkpadtest1@hotmail.com



Send email message from:

Account



To:

%UserEmail%



CC:



BCC:



Subject:

Your SAP credentials



Body:

Dear %UserFirstName%,



we just created a SAP login for you:  
Username: %UserName%  
initial password: %InitialPassword%

Body is HTML:



Attachment(s):

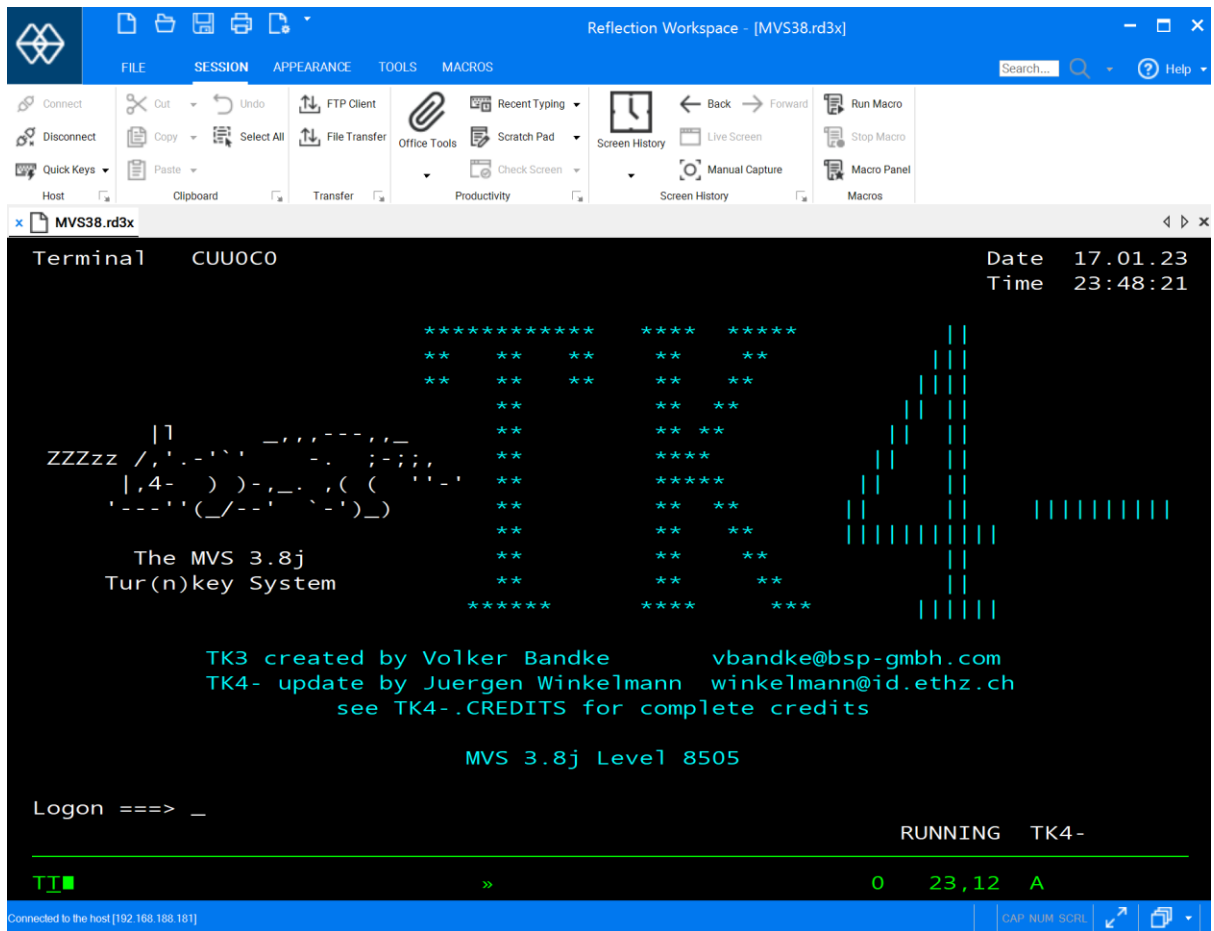


On error

Save

Cancel





## Open terminal session



➤ Open a new terminal session [More info](#)

### Select parameters

Provider:	<input type="text" value="Micro Focus Reflection"/>	
Installation path:	<input type="text" value="C:\Program Files (x86)\Micro Focus\Reflection"/>	{x}
Configuration:	<input type="text" value="Existing profile"/>	
Profile:	<input type="text" value="C:\Users\pkpad\Documents\Micro Focus\Reflection\MVS38.rd3x"/>	{x}
Attach to running session:	<input type="checkbox"/>	

> Variables produced [TerminalSession](#)

On error

Save

Cancel

1 ➤ **Open terminal session**  
Open a [Micro Focus Reflection](#) terminal session and store it into [TerminalSession](#)

2 **Set text on terminal session**  
Set text 'HERC03' on [Cursor position](#) on terminal session [TerminalSession](#)

3 **Send key to terminal session**  
Send key [Transmit](#) to terminal session [TerminalSession](#)

4 **Set text on terminal session**  
Set text 'pass4u' on [Cursor position](#) on terminal session [TerminalSession](#)

5 **Send key to terminal session**  
Send key [Transmit](#) to terminal session [TerminalSession](#)

6 **Wait 5 seconds**

7 [Press enter 2x to get to the main screen](#)

8 **Send key to terminal session**  
Send key [Transmit](#) to terminal session [TerminalSession](#)

9 **Send key to terminal session**  
Send key [Transmit](#) to terminal session [TerminalSession](#)

10 **Wait 2 seconds**

**Input / output variables** 0

There aren't any input or output variables here yet

**Flow variables** 2

```
Terminal CUU0C0                      Date 22.01.23
System TK4-                          Time 13:03:47
TSO User HERC03

Option ==> 1

      The MVS 3.8j Tur(n)key System
TK4- Version 1.00 Update 08 -- MVS PUT 8505

      TSO Applications

1  RFE      "SPF like" productivity tool
2  RPF      "SPF like" productivity tool
3  IM       IMON/370 system monitor
4  QUEUE    spool browser
5  HELP     general TSO help
6  UTILS    information on utilities and commands available
7  TERMTST  verify 3270 terminal capabilities

      Enter X to Terminate


PF3=Terminate

TT  »↑  0  5,14  A
```

11

Press '3' to start IMON / 370 system monitor


12



Set text on terminal session

Set text 3 on Cursor position on terminal session TerminalSession

13




Send key to terminal session

Send key Transmit to terminal session TerminalSession

14

extract field 'CPU serial '


15



Get text from terminal session

Get text from Field with label 'CPU-SERIAL-' of terminal session TerminalSession and store it into TerminalText


16



Set text on terminal session






Set text 'x' on Cursor position on terminal session TerminalSession

















17



Send key to terminal session

Send key Transmit to terminal session TerminalSession

18	Press 'x' to exit TSO	
19	 <b>Set text on terminal session</b> Set text 'x' on <a href="#">Cursor position</a> on terminal session <a href="#">TerminalSession</a>	
20	 <b>Send key to terminal session</b> Send key <a href="#">Transmit</a> to terminal session <a href="#">TerminalSession</a>	
21	 <b>Set text on terminal session</b> Set text 'logoff' on <a href="#">Cursor position</a> on terminal session <a href="#">TerminalSession</a>	
22	logoff	
23	 <b>Send key to terminal session</b> Send key <a href="#">Transmit</a> to terminal session <a href="#">TerminalSession</a>	
24	 <b>Close terminal session</b> Close terminal session <a href="#">TerminalSession</a>	

1	Start of autogenerated actions using the recorder	1	Start of autogenerated actions using the recorder
2	 <b>Launch new Microsoft Edge</b> Launch Microsoft Edge, navigate to 'https://	2	 <b>Launch new Microsoft Edge</b> Launch Microsoft Edge, navigate to <a href="#">NewQuoteURL</a> and store the instance into <a href="#">Browser</a>
3	 <b>Populate text field on web page</b> Populate text field <a href="#">Input text 'Name'</a> with 'new quote from a customer' using emulated typing	3	 <b>Populate text field on web page</b> Populate text field <a href="#">Quote Name</a> with 'new quote from a customer' using emulated typing
4	 <b>Press button on web page</b> Press web page button <a href="#">Button 'id- ... d_2_search'</a>	4	 <b>Press button on web page</b> Press web page button <a href="#">Lookup Price List</a>
5	 <b>Click link on web page</b> Click on <a href="#">List item ' ... tainer_0_0'</a> of web page	5	 <b>Click link on web page</b> Click on <a href="#">Default Price List Item</a> of web page
6	 <b>Press button on web page</b> Press web page button <a href="#">Button 'id- ... d_4_search'</a>	6	 <b>Press button on web page</b> Press web page button <a href="#">Lookup Customer</a>
7	 <b>Click link on web page</b> Click on <a href="#">List item ' ... tainer_0_0' 2</a> of web page	7	 <b>Click link on web page</b> Click on <a href="#">Fabrikam Account Item</a> of web page
8	 <b>Press button on web page</b> Press web page button <a href="#">Button 'quo ... y00-button'</a>	8	 <b>Press button on web page</b> Press web page button <a href="#">Button Save</a>
9	 <b>Get details of element on web page</b> Get attribute 'innertext' of element <a href="#">Input text 'Quote ID'</a> on web page and store it into <a href="#">innertext</a>	9	 <b>Get details of element on web page</b> Get attribute 'innertext' of element <a href="#">Quote ID</a> on web page and store it into <a href="#">QuoteID</a>
10	End of autogenerated actions using the recorder	10	End of autogenerated actions using the recorder

## Populate text field on web page



Fill a text field in a web page with the specified text [More info](#)

### Advanced

If field isn't empty:

Replace text



Populate text using physical  
keystrokes:



Emulate typing:



Unfocus text box after filling it:



Wait for page to load:



Timeout on webpage load:

60

{x}



On error

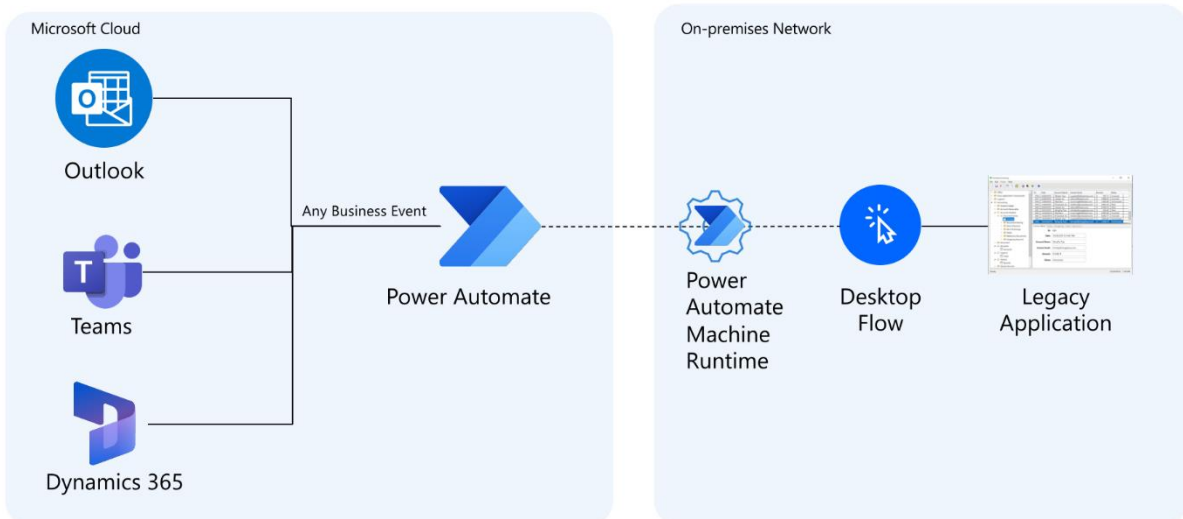
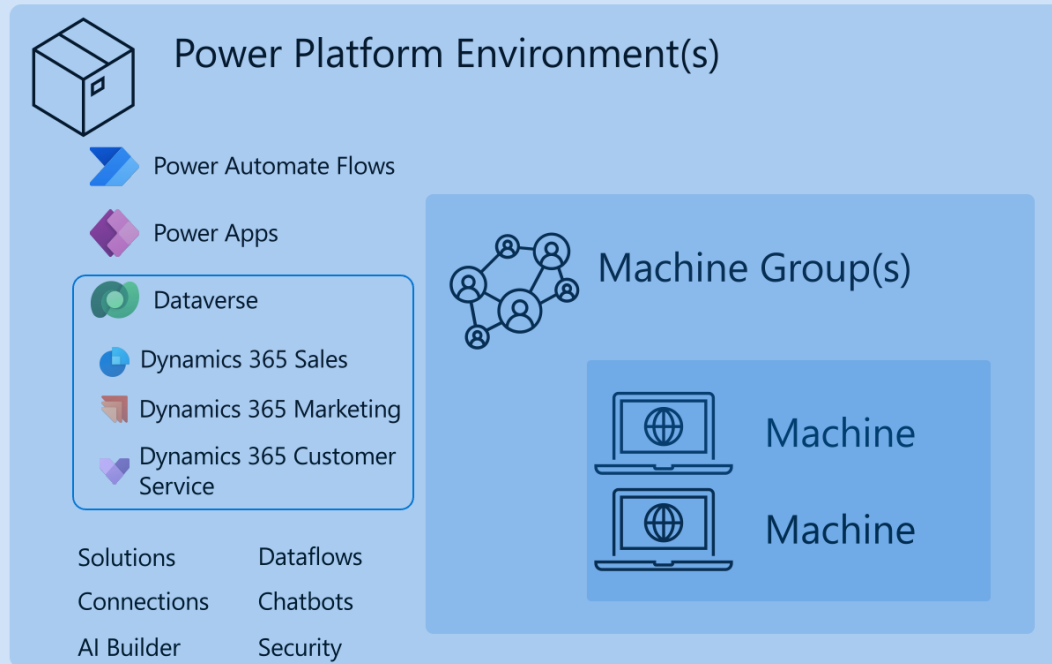
Save

Cancel

The screenshot displays the Dynamics 365 Sales Hub interface for creating a new quote. The top navigation bar includes the Dynamics 365 logo, Sales Hub, a search bar, and various utility icons. The left sidebar shows the navigation menu with options like Home, Recent, Pinned, My Work, Sales accelerator, Activities, Dashboards, Customers, Accounts, Contacts, Sales, Leads, Opportunities, Competitors, Collateral, Quotes, Orders, and Sales. The main content area is titled "new quote from a customer" and shows a "Quote" record with a total amount of €0.00 and a status of "In Progress". The form is divided into several sections: "Summary" (containing fields for Quote ID, Revision ID, Owner, Name, Currency, and Price List), "PRODUCTS" (with a table for product details), "SALES INFORMATION" (including Opportunity and Potential Customer), and "SHIPPING INFORMATION" (with a field for Shipping Method). The "Save" button in the top navigation bar is highlighted with a red box.

## Chapter 9: Leveraging Cloud Services and Power Platform

### Customer Tenant (Azure)



Power Automate machine runtime

←

☰

Machine settings

Machine group

Troubleshoot

About

Machine settings

Machine details

Machine environment

Refresh

Connected with Power Automate as PADWIN11-A in Environment D365 Contoso Demo

Name

PADWIN11-A

Description

Edit details

View machine in portal

Machine environment

Your machine will run flows in this environment.

D365 Contoso Demo

Power Automate machine runtime

←

☰

Machine settings

Machine group

Troubleshoot

About

Machine groups

Available machine groups

New machine group

Refresh

Joined machine group

Machines

2

PAD Demo Group

local installations

Owner:

Available machine groups

D365 Contoso Demo machine groups

Added machines will adopt group permissions and connections, losing their individual permissions and connections [Learn more](#)

Search

**Power Automate** Search

Environments: D365 Contoso Demo

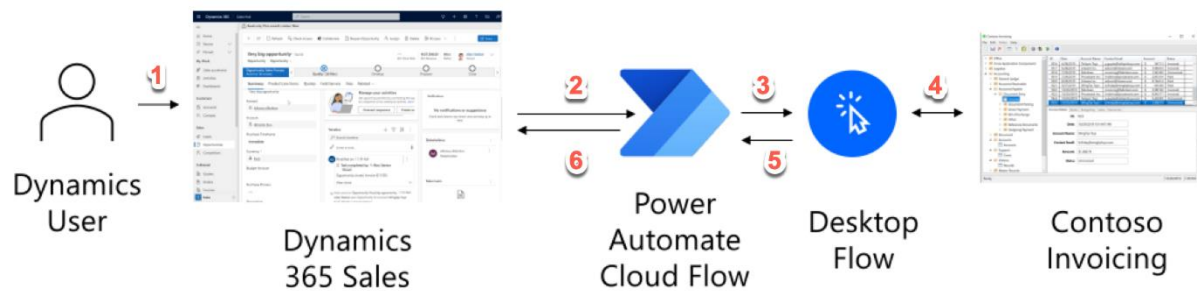
+ New Scale with Starter Kit Search

**Machines**  
Check the real-time health and status of your machines and the desktop flows running on them. [Learn more](#)

Machines Machine groups VM images Network connections (preview) Gateways

Machine name ↑ ↓	Descrip... ↓	Version	Group ↓	Status	Flows...	Flows...	Ac... ↓	Owner
PADWIN11-A	—	2.28.135.23016	PAD Demo G...	✓ Connect...	0	—	Owner	
PADWIN11-B	—	2.28.135.23016	PAD Demo G...	✗ Disconn...	0	—	Owner	

Home Approvals My flows Create Templates Connectors Data Monitor Cloud flow activity Desktop flow activity (preview) Desktop flow runs Machines AI Builder Process advisor Solutions Ask a chatbot learn





# Variables



{x}



Search variables



Input / output variables

4



{x} AccountName

Test Account

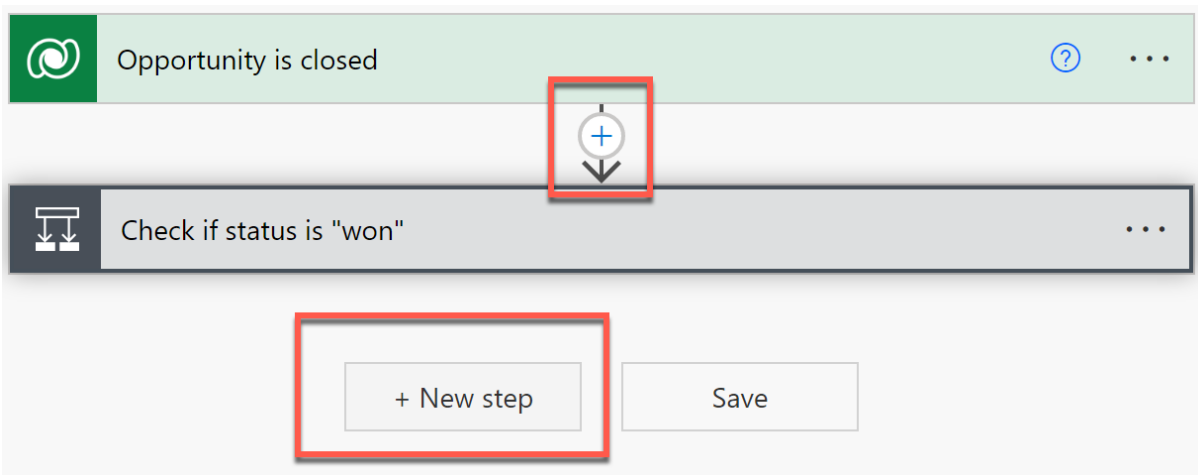
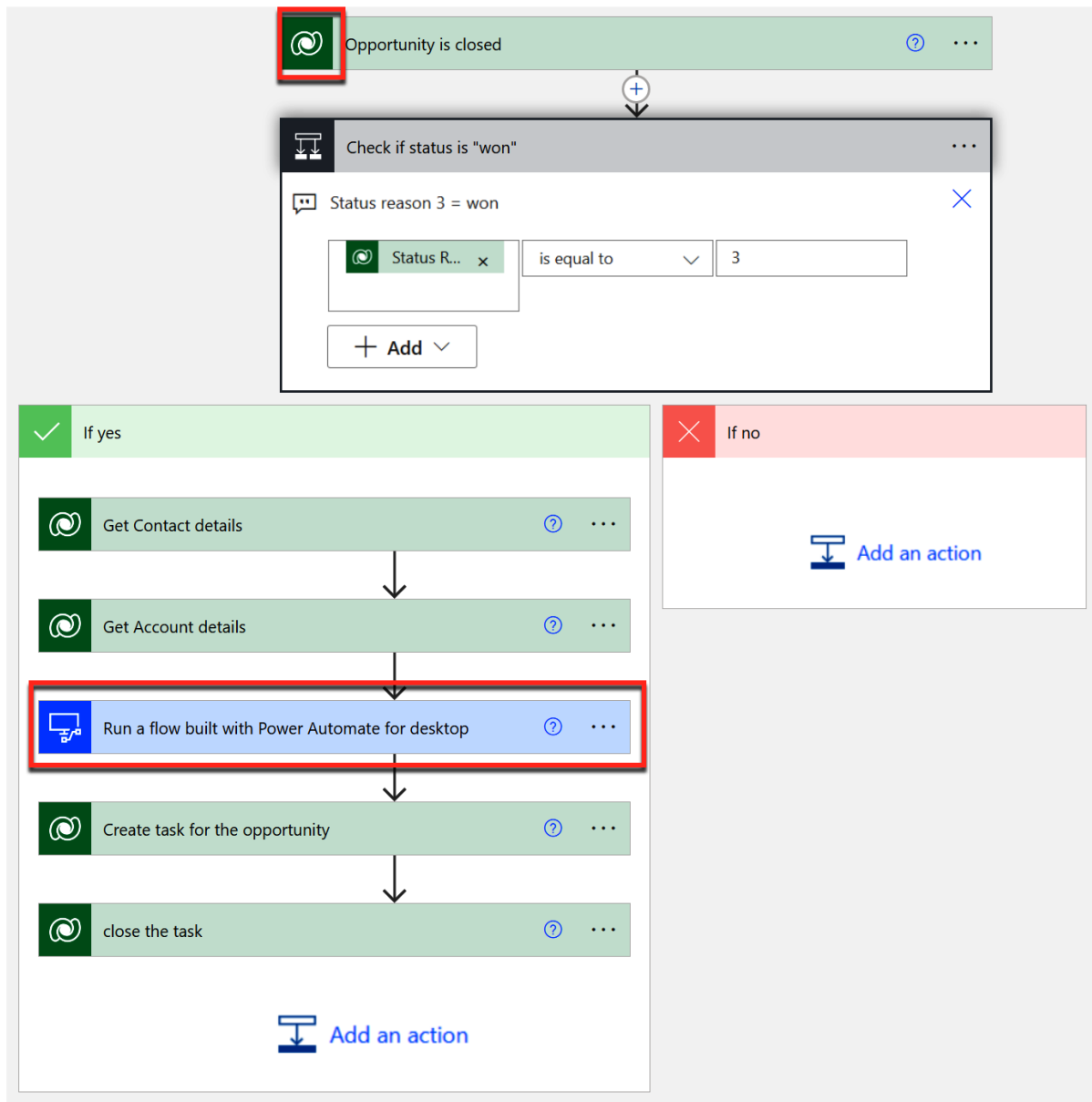
{x} Amount

123

{x} ContactName

Test Contact

{x} InvoiceID





## Choose an operation



 Search connectors and actions

All

Built-in

Standard

Premium

Custom

My clipboard



Control



AI Builder



Desktop flows



Excel Online (Business)



MSN Weather



Mail



Microsoft Dataverse

Triggers

Actions

See more



Condition  
Control



Analyze positive or negative sentiment in text  
AI Builder




Classify text into categories with one of your custom models  
AI Builder



Classify text into categories with the standard model (preview)  
AI Builder





Desktop flows

...

\* Connect

①

Directly to machine

▼

\*

\* Machine or machine group

①

Select an item

▼

\* Domain and username

①

Format as domain\username or username@domain.com


\* Password

①

Password credential

Create

← Opportunity-2-Contoso Invoicing

 Run a flow built with Power Automate for desktop ? ...


\* Desktop flow

09 - Sales Opportunity to Contoso ▼ Edit


\* Run Mode

Attended (runs when you're signed in) ▼


AccountName

 Account Name ×


ContactName

 Full Name ×

Amount

 Actual Revenue ×


[Show advanced options](#) ▼

 Create task for the opportunity ? ...

\* Table name

Tasks ▼


\* Subject

Opportunity closed. Invoice ID  InvoiceID ×

Description

Type additional information to describe the task.

Due Date

 Modified On ×

Duration

Type the number of minutes spent on the task. The duration is used in reportin

Owner (Owners)

Enter the user or team who is assigned to manage the record. This field is upda

Priority

High ▼

Regarding (Accounts)

Choose the record that the task relates to.

SharePoint

Search this site

PA

Power Automate Desktop Community

Public group 2 members

Home

Conversations

Documents

Notebook

Pages

Site contents

Recycle bin

Edit


+ New

Page details

Analytics

News

+ Add



Keep your team updated with news on your team site

From the site home page you'll be able to quickly author a news post - a status...


Add News

Activity

PA

Power Automate Desktop Community

Power Automate Desktop Com...



View and share files

Collaborate on content with your team.

Quick links

Learn about a team site

Learn how to add a page

Documents

See all

All Documents

Name


Business Leadeers Guide RPA.pdf

Release Infos

Whitepaper

## Copy file (Preview)

There are no valid connection references available. Fix invalid connections references, use an existing or create a new one to utilize the connector. [Connection references](#)

 Copies a file. Works in a similar way to the "Copy to" command in SharePoint libraries. Returns information about the new file after copy. [More info](#)

### Select parameters

 Connection references

Current Site Address:


{x} v ⓘ

File to Copy:

 {x} ⓘ

Destination Site Address:

{x} v ⓘ

 On error

Save

Cancel

### Connection references ⓘ

 Add new connection reference v



### No Connection references available

Please try one of the following:

1. Create a new connection reference
2. Make sure desktop flows service [endpoints](#) required for runtime are not blocked
3. Check your internet connection or run the [Windows internet connections troubleshooter](#)

Retry

Ok

Cancel

Subflows

Main

1

Convert file to binary data

Get the binary data of file 'C:\Users\ ■■■ \Documents\Inventory list with highlighting.xlsx' and store it into **BinaryData**

2

Create file (Preview)

Uploads a file to a SharePoint site. Make sure to pick an existing library. Stores output into **CreateFileResponse**

### Create file (Preview)

Uploads a file to a SharePoint site. Make sure to pick an existing library. [More info](#)

#### Select parameters

Connection references

Site Address: **Power Automate Desktop Community - https://** {x} ⌵ ⓘ  
/sites/PowerAutomateDesktopCommunity

Folder Path: **/Shared Documents** ⓘ

File Name: **inventory.xlsx** {x} ⓘ

File Content: **%BinaryData%** {x} ⓘ

> Advanced

> Variables produced **CreateFileResponse**

On error

Save

Cancel



## Create EC2 session



Create an EC2 client to automate EC2 web services [More info](#)

### Select parameters

Access keys:



Access key ID:



Secret:



Region endpoint:

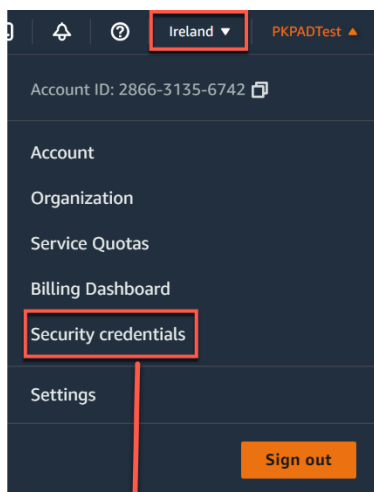


> Variables produced **Ec2Client**

On error

Save

Cancel



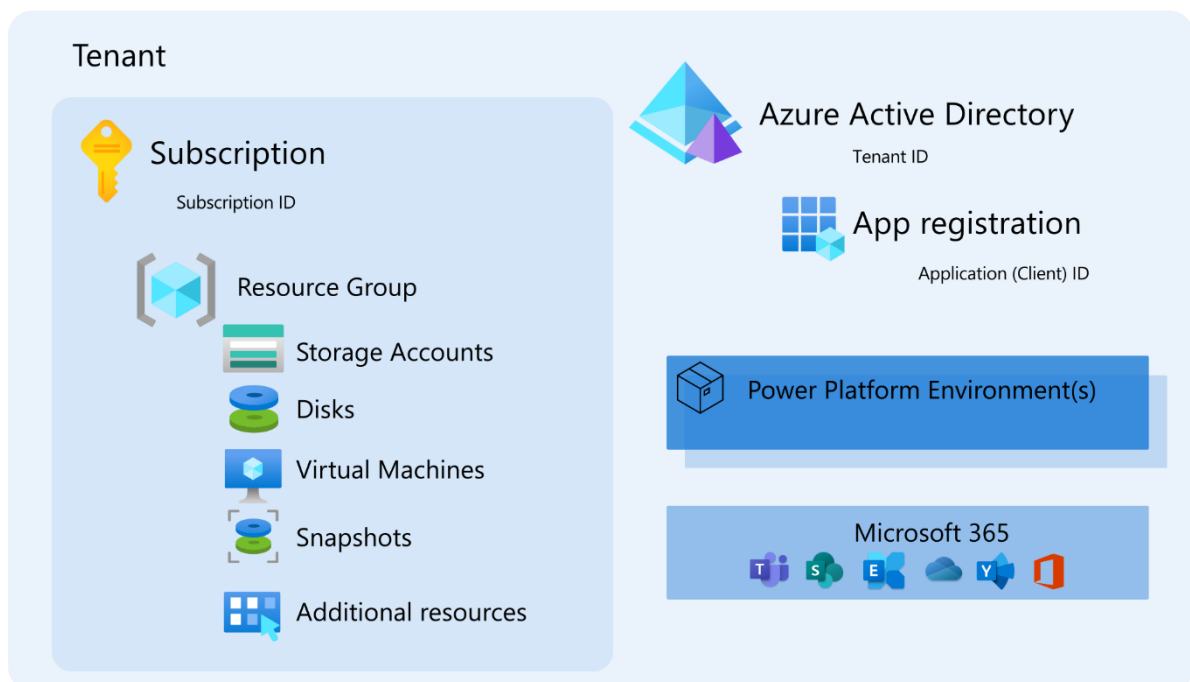
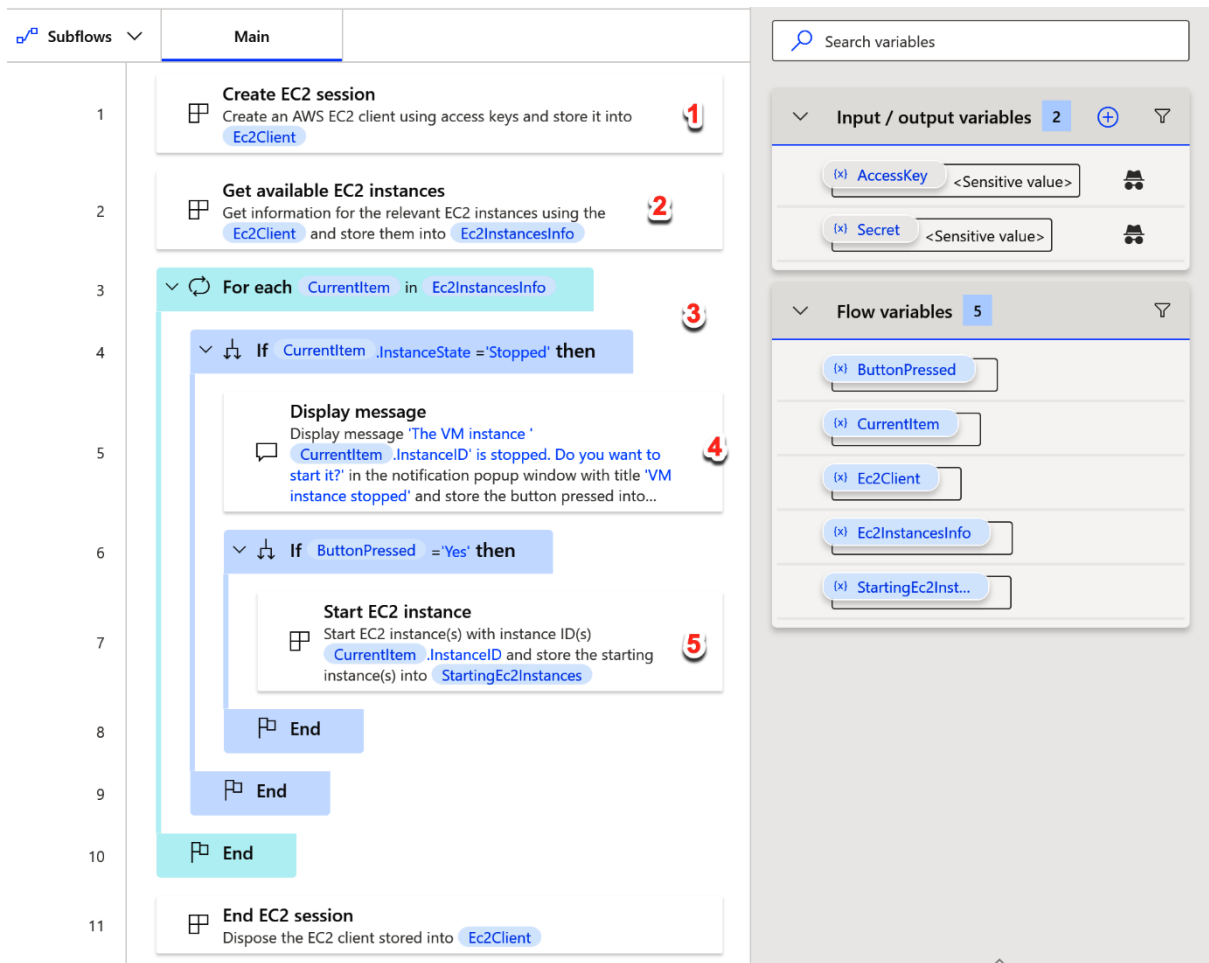
### Access keys (1)

Use access keys to send programmatic calls to AWS from the AWS CLI, AWS Tools for PowerShell, AWS SDKs, or direct AWS API calls. You can have a maximum of two access keys (active or inactive) at a time. [Learn more](#)

Actions ▼

Create access key

	Access key ID	Created on	Access key last used	Region last used	Service last used	Status
	AKIAUFPEPNFDJICURVP	13 minutes ago	10 minutes ago	eu-west-1	ec2	Active



Search

<

Delete

Endpoints

Preview features

Overview

Quickstart

Integration assistant

Manage

Branding & properties

Authentication

Certificates & secrets

Token configuration

API permissions

Expose an API

App roles

Owners

Roles and administrators

Manifest

Essentials

Display name : [Power Automate Desktop](#)

Application (client) ID : [REDACTED]

Object ID : [REDACTED]

Directory (tenant) ID : [REDACTED]

Supported account types : [My organization only](#)

Client credentials : [0 certificate, 1 secret](#)

Redirect URIs : [1 web, 0 spa, 4 public client](#)


Application ID URI : [Add an Application ID URI](#)

Managed application in I... : [Power Automate Desktop](#)

Get Started Documentation


Build your application with the Microsoft identity platform

The Microsoft identity platform is an authentication service, open-source libraries, and application management tools. You can create modern, standards-based authentication solutions, access and protect APIs, and add sign-in for your users and customers. [Learn more](#)



## Create session



 Creates an Azure session using service principal [More info](#)

### Select parameters

Client ID :  {x} ⓘ

Client secret: ⓘ  ⓘ

Tenant ID:  {x} ⓘ

Subscription ID:  {x} ⓘ

> Variables produced [AzureClient](#)

 On error

Save

Cancel

Subflows

Main

1

Create session

Create an Azure session with service principal and store the result in `AzureClient`

2

Get virtual machines

Get basic information for the virtual machines and store it into `VirtualMachinesInfo`

3

Set variable

{x} Assign to variable `VMName` the value `VirtualMachinesInfo[0].Name`

4

If `VirtualMachinesInfo[0].Status <> 'running'` then

5

Start virtual machine

Start virtual machine `VMName` of resource group `ResourceGroup`

6

End

7

Create managed disk

Create managed disk `'paddisk2'`

8

Wait 10 seconds

9

Attach disk

Attach managed disk `'paddisk2'` to VM `VMName` of resource group `ResourceGroup`

10

End session

End the Azure session stored in `AzureClient`

Variables

Search variables

Input / output variables 3

`ClientSecret` <Sensitive value>

`Location` northeurope

`ResourceGroup` PAD

Flow variables 4

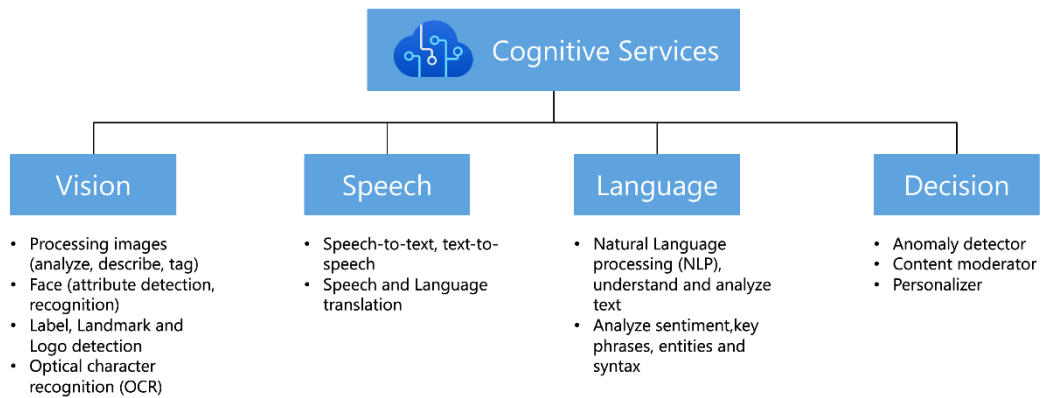
`AzureClient` Azure client

`ManagedDisk`

`VirtualMachin...` [...]

`VMName` PADVMTest1

## Chapter 10: Leveraging Artificial Intelligence



**MSCognitiveServices** Cognitive services multi-service account

Search Bing Statistics Add-in Delete

**Overview**

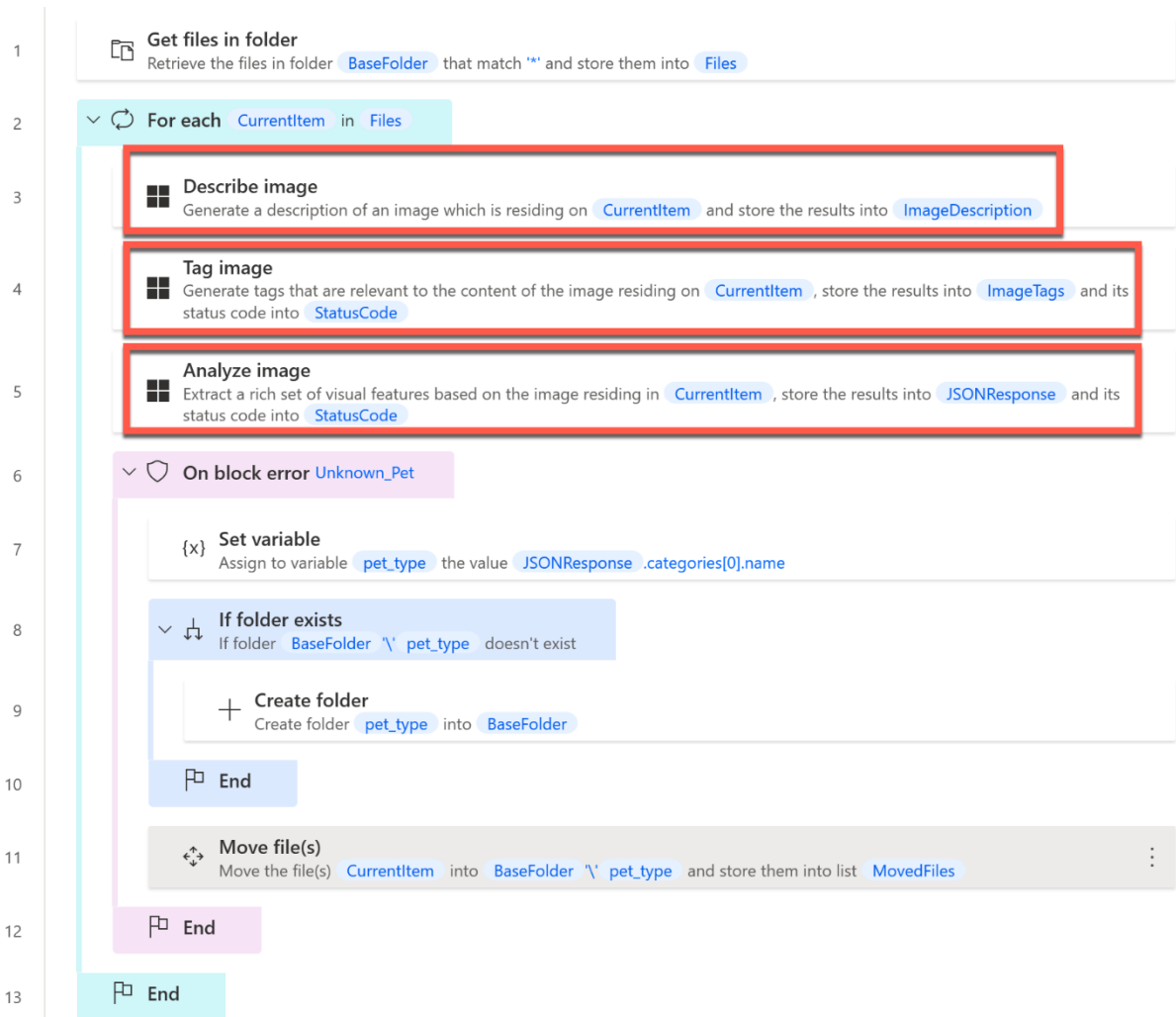
- Activity log
- Access control (IAM)
- Tags
- Diagnose and solve problems

**Resource Management**

- Keys and Endpoint**
- Pricing tier
- Networking
- Identity
- Cost analysis

**Essentials** JSON View

Resource group ( <a href="#">move</a> )	: <a href="#">PAD</a>
Status	: Active
Location	: North Europe
Subscription ( <a href="#">move</a> )	:
Subscription ID	:
API type	: All Cognitive Services
Pricing tier	: Standard
Endpoint	: https://mscognitiveservices.cogni...
Manage keys	: <a href="#">Click here to manage keys</a>
Autoscale	: <a href="#">Disabled</a>
Tags ( <a href="#">edit</a> )	: <a href="#">Click here to add tags</a>



← Variable value

**ImageTags** ['tags'] (List of Custom objects)

#	Item
0	{ 'name': cat, 'confidence': 0.999975919723511 } <a href="#">More</a>
1	{ 'name': animal, 'confidence': 0.936986923217774 } <a href="#">More</a>
2	{ 'name': carnivore, 'confidence': 0.929778277873993 } <a href="#">More</a>
3	{ 'name': mammal, 'confidence': 0.857378482818604, 'hint':... } <a href="#">More</a>
4	{ 'name': kitten, 'confidence': 0.716156482696533 } <a href="#">More</a>
5	{ 'name': grey, 'confidence': 0.705203890800476 } <a href="#">More</a>
6	{ 'name': feline, 'confidence': 0.682758450508118 } <a href="#">More</a>
7	{ 'name': felidae, 'confidence': 0.643301963806152 } <a href="#">More</a>
8	{ 'name': kitty, 'confidence': 0.638205409049988 } <a href="#">More</a>
9	{ 'name': domestic cat, 'confidence': 0.630339682102203 } <a href="#">More</a>
10	{ 'name': gray, 'confidence': 0.582340598106384 } <a href="#">More</a>

[Close](#)

← Variable value

**ImageDescription** ['description'] (Custom object)

Name	Value
tags	[cat, animal, sitting, mammal, grey, gray, standing, looking, house, fr... <a href="#">More</a>
captions	[[ 'text': a cat that is looking at the camera, 'confidence': 0.92504803... <a href="#">More</a>

[Close](#)

← Variable value

**JSONResponse** ['categories'][0] (Custom object)

Name	Value
name	animal_cat
score	0.99609375

[Close](#)



Google Cloud



My First Project ▼



Cloud overview



View all products

PINNED

**API**

**APIs & Services**



Billing



IAM & Admin



Marketplace

Enabled APIs & services

Library

Credentials

OAuth consent screen

Page usage agreements

Welcom

Google Cloud My First Project sentiment

APIs & Services

Enabled APIs & services

Library

Credentials

OAuth consent screen

Page usage agreements

API/Service Details

Cloud Natural Language API

Provides natural language understanding technologies, such as sentiment analysis, entity recognition, entity sentiment analysis, and other text annotations, to developers.

By Google Enterprise API

Service name: language.googleapis.com

Type: Public API

Status: Enabled

CREATE CREDENTIALS

CREATE CREDENTIALS

DELETE

RESTORE DELETED CREDENTIALS

API key

Identifies your project using a simple API key to check quota and access

OAuth client ID

Requests user consent so your app can access the user's data

Service account

Enables server-to-server, app-level authentication using robot accounts

Help me choose

Asks a few questions to help you decide which type of credential to use

API Keys

No API keys to display

Bud Spencer - Notepad

File Edit View

Carlo Pedersoli, known professionally as Bud Spencer, was an Italian actor, professional swimmer and water polo player. He is one of the greatest actors in history.

Ln 1, Col 1

100%

Windows (CRLF)

UTF-8



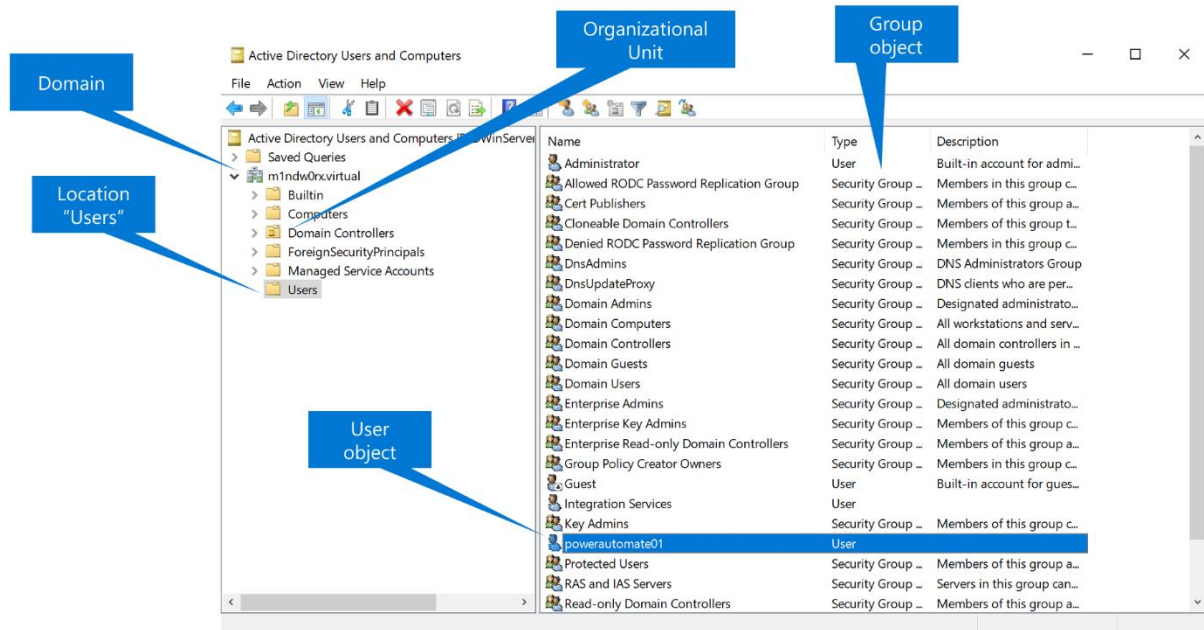


JSONResponse2 ['entities'] (List of Custom objects)

#	Item
0	{ 'name': Carlo Pedersoli, 'type': PERSON, 'metadata': { 'mid': /m/018gm, 'wikipedia_url': https://en.wikipedia.org/w... <a href="#">More</a>
1	{ 'name': swimmer, 'type': PERSON, 'metadata': { }, 'salience': 0,067237504, 'mentions': [ ...] } <a href="#">More</a>
2	{ 'name': water polo player, 'type': OTHER, 'metadata': { }, 'salience': 0,061917726, 'mentions': [{ ... }] } <a href="#">More</a>
3	{ 'name': Italian, 'type': LOCATION, 'metadata': { }, 'salience': 0,039996617, 'mentions': [ ...] } <a href="#">More</a>
4	{ 'name': actors, 'type': PERSON, 'metadata': { }, 'salience': 0,026108624, 'mentions': [ ...] } <a href="#">More</a>
5	{ 'name': history, 'type': OTHER, 'metadata': { }, 'salience': 0,01615264, 'mentions': [ ...] } <a href="#">More</a>
6	{ 'name': one, 'type': OTHER, 'metadata': { }, 'salience': 0,012968908, 'mentions': [ ...] } <a href="#">More</a>
7	{ 'name': one, 'type': NUMBER, 'metadata': { 'value': 1 }, 'salience': 0, 'mentions': [ ...] } <a href="#">More</a>

Close

## Chapter 11: Working with APIs and Services



powerautomate01 Properties



Published Certificates	Member Of	Password Replication	Dial-in	Object
Security	Environment	Sessions	Remote control	
General	Address	Account	Profile	Telephones
Remote Desktop Services Profile	COM+	Attribute Editor		

Attributes:

Attribute	Value
desktopProfile	<not set>
destinationIndicator	<not set>

String Attribute Editor



Attribute: distinguishedName

Value:

CN=powerautomate01,CN=Users,DC=m1ndw0rx,DC=virtual

Clear

OK

Cancel

employeeType	<not set>
extensionName	<not set>
facsimileTelephoneNumber	<not set>

View

Filter

OK


Cancel

Apply

Help



## Open SQL connection




 Open a new connection to a database [More info](#)

### Select parameters

Connection string:

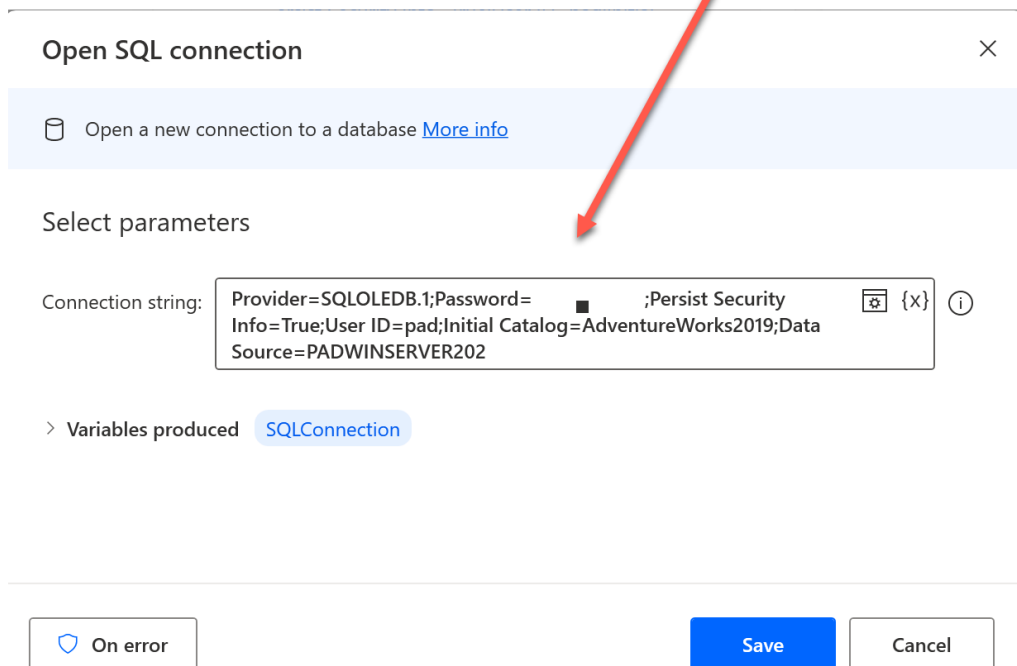
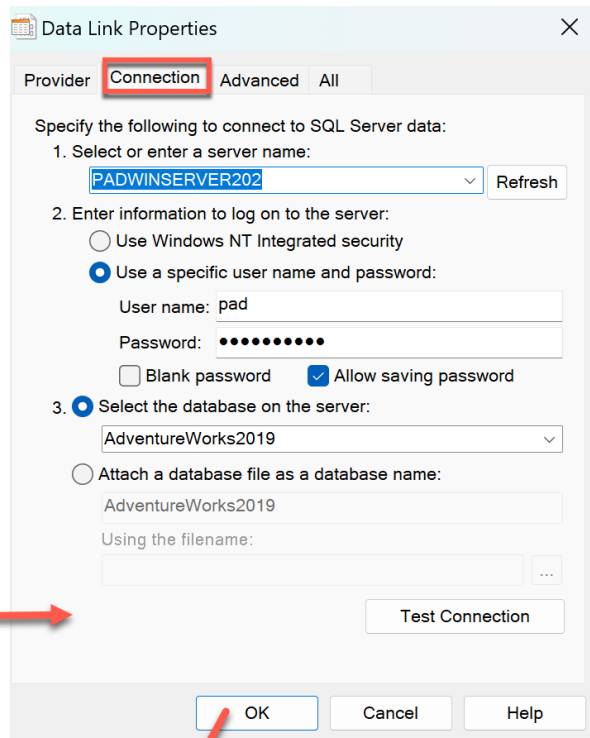
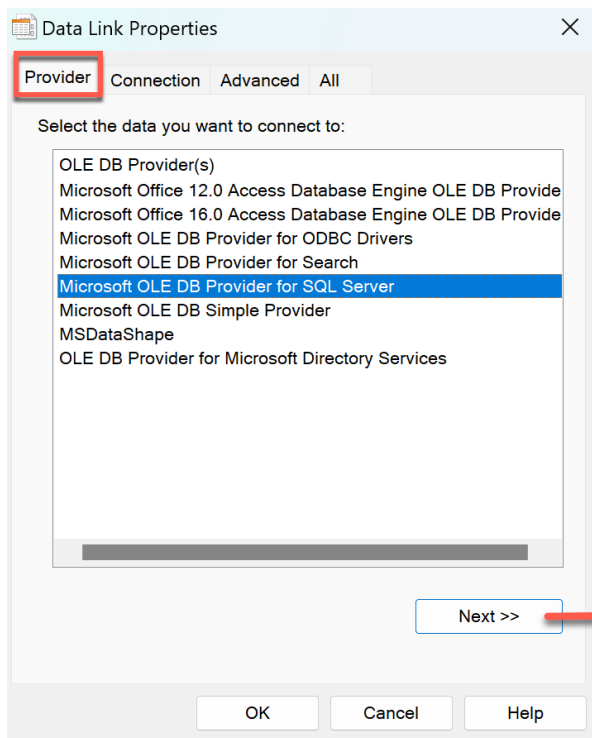
  

> Variables produced **SQLConnection2**

 On error

Save

Cancel



## Execute SQL statement



 Connect to a database and execute an SQL statement [More info](#)

### Select parameters

Get connection by:   

SQL connection:   

SQL statement: 



1 **SELECT TOP** (1000) [CustomerID]

2     ,[TerritoryID]

3     ,[AccountNumber]


4     ,[ModifiedDate]

5 **FROM** [AdventureWorks2019].[Sales].[Customer]

Timeout:   

> Variables produced QueryResult

 On error

Save












Cancel

## Synchronize directories




 Synchronize the files and subdirectories of a given folder with a given remote FTP directory [More info](#)

### Select parameters

FTP connection:	<input type="text" value="%FTPConnection%"/>	
Synchronization direction:	<input type="text" value="Remote -&gt; Local (Download)"/>	
Files to sync:	<input type="text" value="All files"/>	
Local folder:	<input type="text" value=""/>  {x}	
FTP directory:	<input type="text" value="/"/>	
Delete if source is absent:	<input type="checkbox"/>	
Include subdirectories:	<input checked="" type="checkbox"/>	
Time difference in hours:	<input type="text" value="0"/> {x}	
Time difference in minutes:	<input type="text" value="0"/> {x}	
Time difference ahead:	<input checked="" type="checkbox"/>	

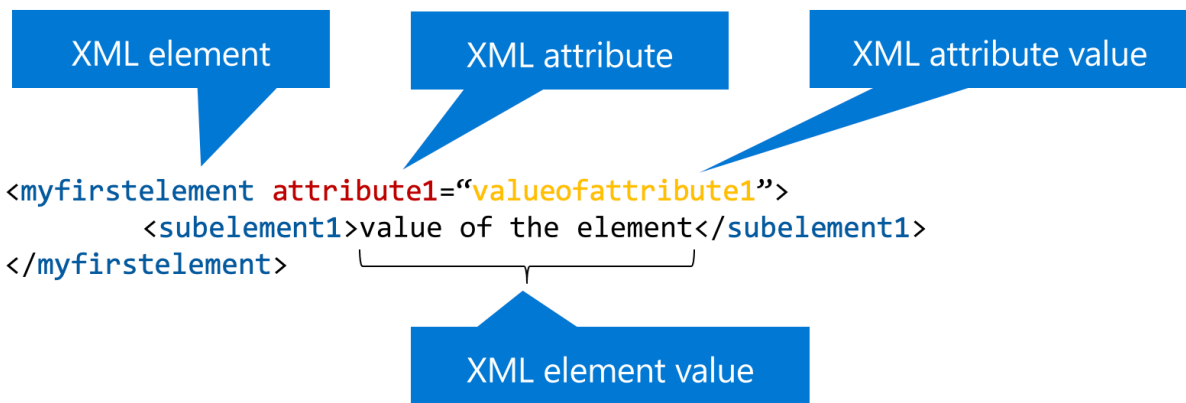
> Variables produced FilesAdded FilesModified FilesDeleted

 On error

Save

Cancel





## Zip files

Compress one or more files or folders into a zip archive [More info](#)

### Select parameters

Archive path:

{x} ⓘ

File(s) to zip:

{x} ⓘ

Compression level:

Best balance of speed and compression

⌵ ⓘ

Password:

ⓘ ⌵

ⓘ

Archive comment:

{x} ⓘ

> Variables produced

ZipFile

On error

Save

Cancel

## Create random text



Generates a text of specified length consisting of random characters. This can be useful for generating passwords [More info](#)

### Select parameters

Use uppercase letters (A-Z):



Use lowercase letters (a-z):



Use digits (0-9):



Use symbols ( , . < > ? ! + - \_ # \$ ^ ):



Minimum length:

{x}



Maximum length:

{x}



> Variables produced

RandomText

Save

Cancel

Custom form designer ⓘ

Form elements

Search

Inputs

Text input

Number input

Date input

Time input

Toggle input

Choice set input

File input

Elements

Text block

Image

Actions

Submit

Open URL

Form structure

Add an item from the form elements pane

Double click or drag an item from the sidebar to create or add it to your form. [Learn more](#)

Preview form

Custom form ⓘ

Specifications

Title ⓘ

{x}

Layout

Minimum height in pixels ⓘ

0

Vertical content alignment ⓘ

Top

Background image

URL ⓘ

{x}

Fill mode ⓘ

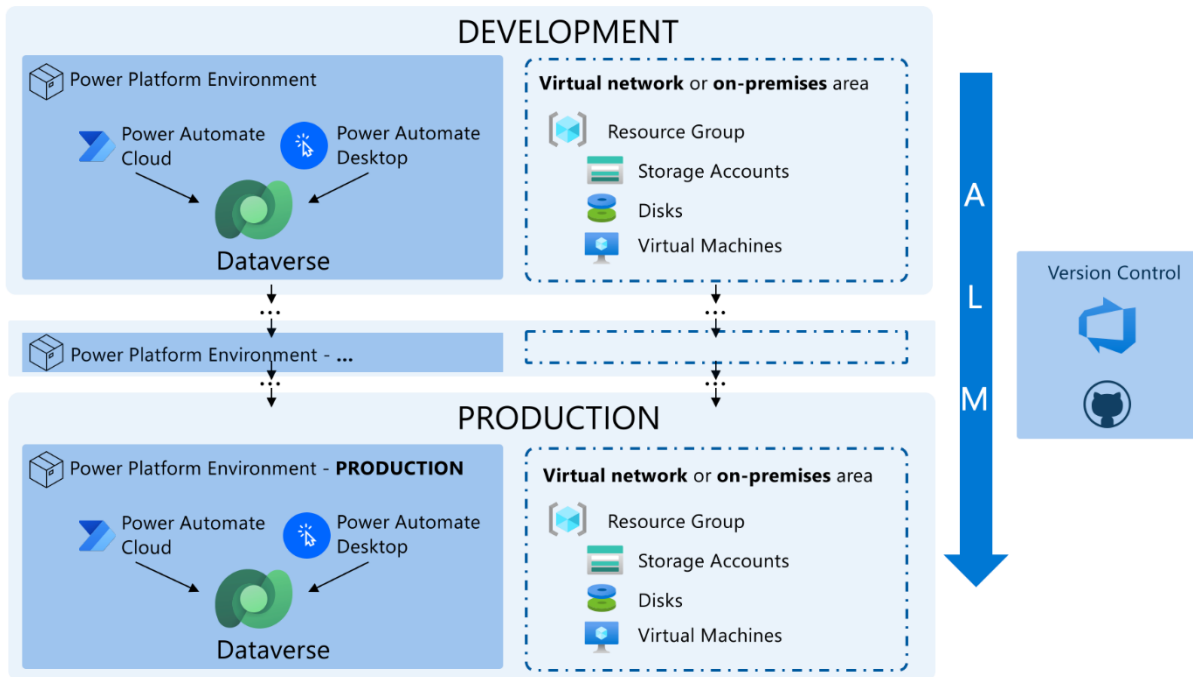
Horizontal alignment ⓘ

Vertical alignment ⓘ

Done

Cancel

## Chapter 12: PAD Enterprise Best Practices



### Azure Virtual Desktop Integration Configuration App

Home > Create Connections

- Create connection**
- Select machine group
- Configure virtual machines
- Scaling strategy
- Next Steps

#### Tenant Uri

#### Tenant Id

#### Environment Id

#### Azure Virtual Desktop Service Principal

Learn more about how to create a service principal and communicate with Azure services.

#### Client Id Name

#### Secret Name

#### Service Principal Power Automate

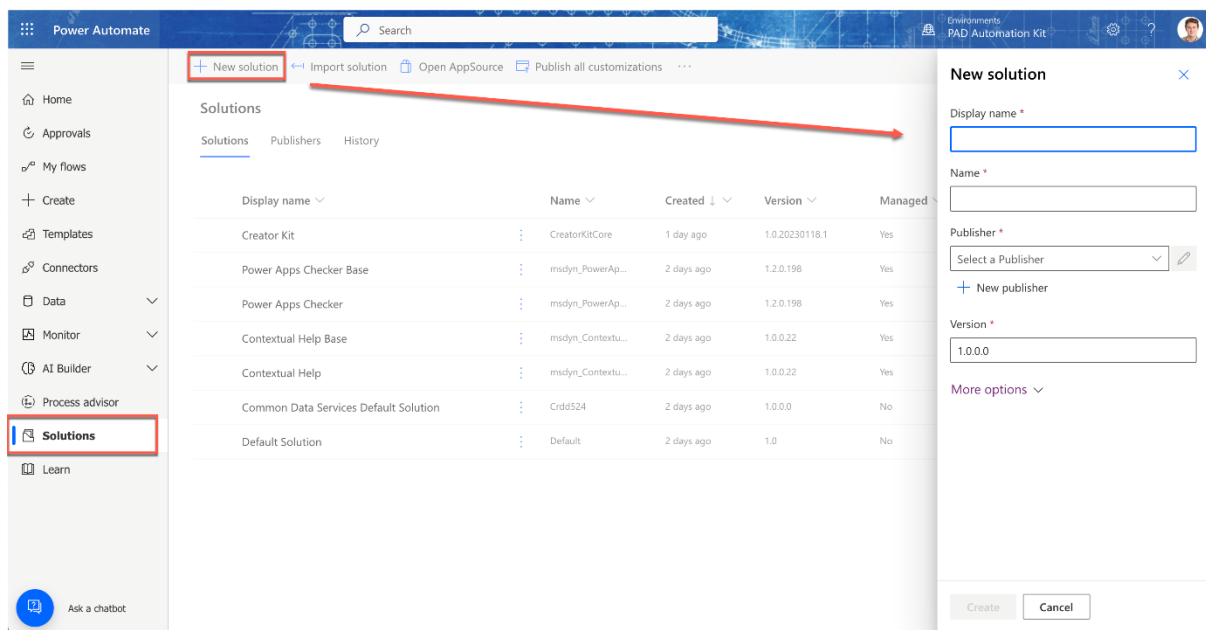
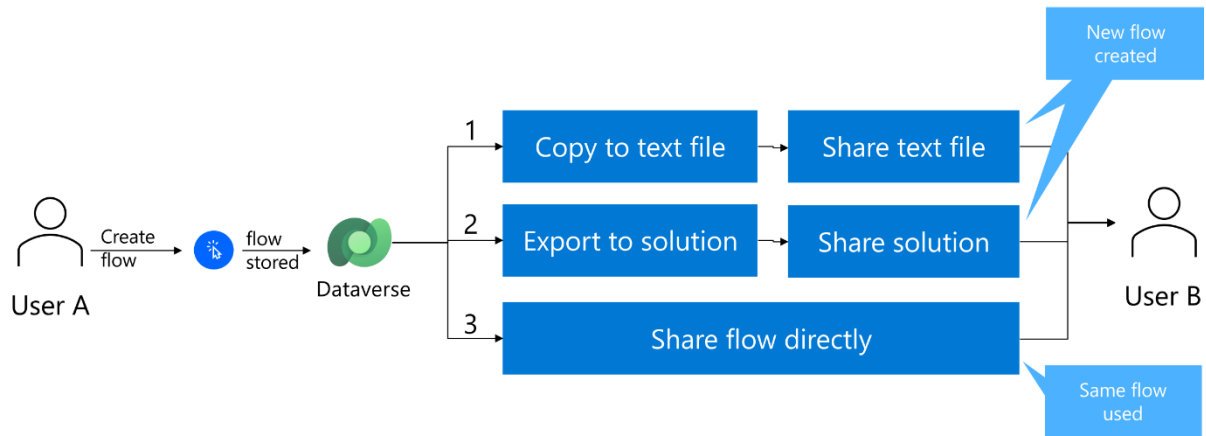
Learn how to create a service principal to call Power Platform APIs

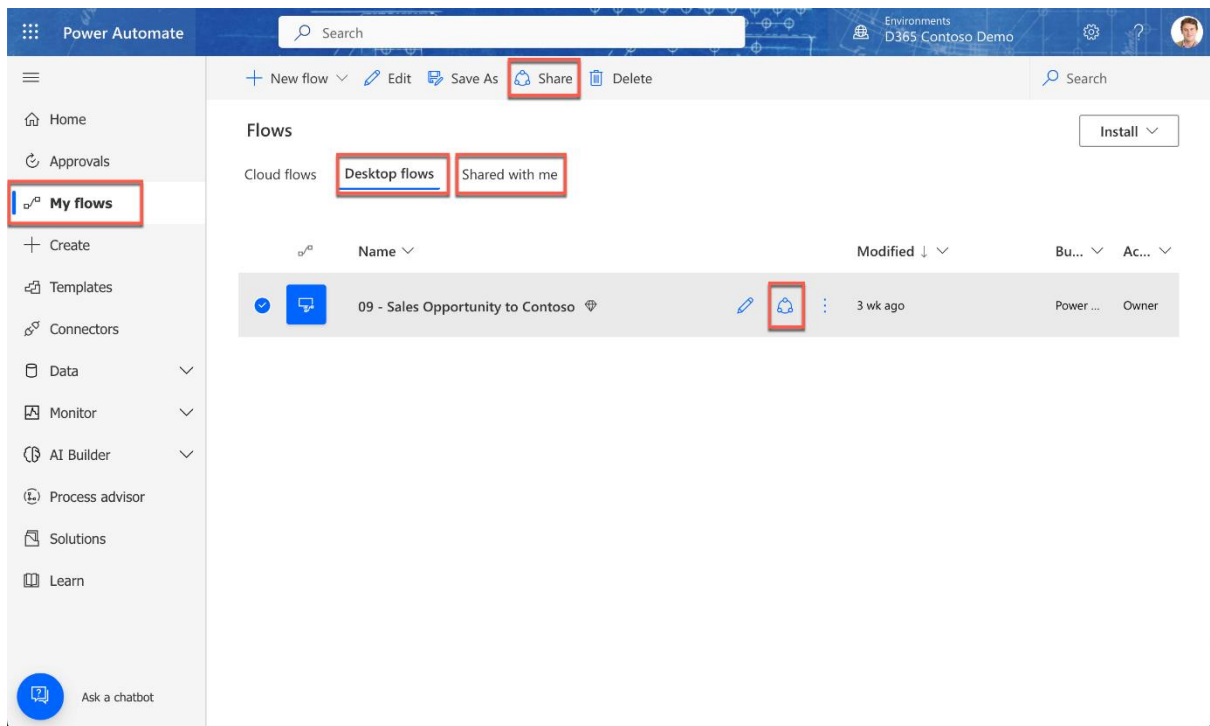
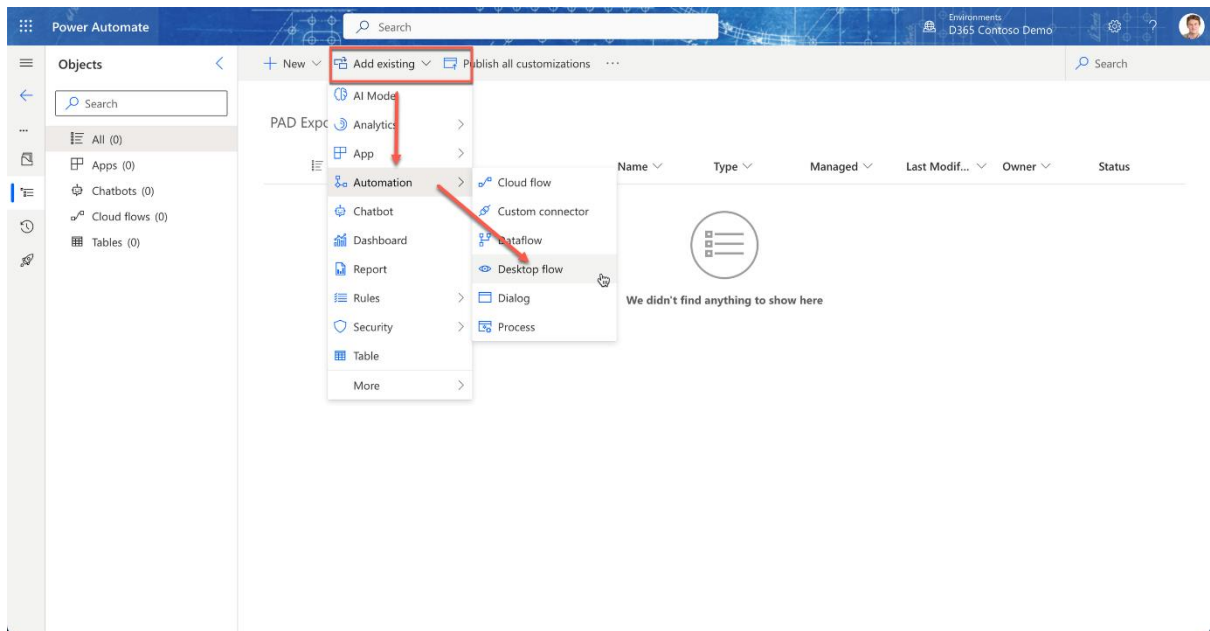
#### Client Id Name

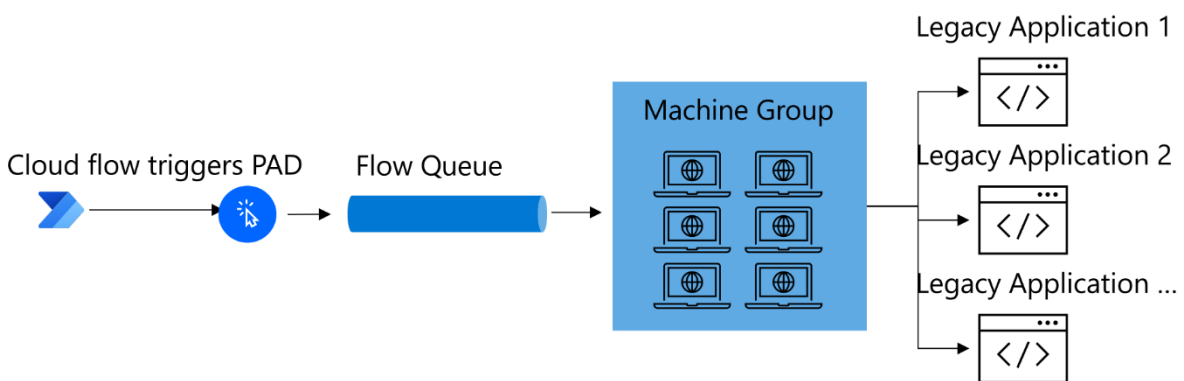
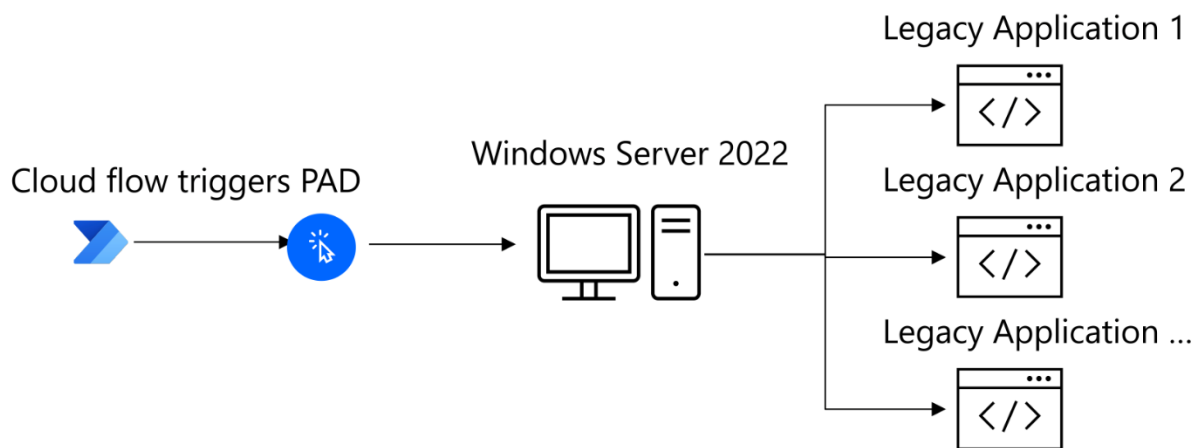
#### Secret Name

Next

Cancel







Power Platform admin center

DLP Policies > New Policy

Policy name

Prebuilt connectors

Custom connectors

Scope

Review

Assign connectors

Business (0) Non-business (974) | Default Blocked (0)

Search connectors

Connectors for non-sensitive data. Connectors in this group can't share data with connectors in other groups. Unassigned connectors will show up here by default.

Name	Blockable	Endpoint configu...	Class	Publisher	Ab...
Azure Resource Manager	Yes	No	Premium	Microsoft	<a href="#">Learn</a>
Power Automate Management	Yes	No	Standard	Microsoft	<a href="#">Learn</a> <a href="#">Man</a>
Azure Key Vault	Yes	No	Premium	Microsoft	<a href="#">Learn</a>
Power Apps for Makers	Yes	No	Standard	Microsoft	<a href="#">Learn</a>
Microsoft Dataverse	No	No	Premium	Microsoft	<a href="#">Learn</a>
Excel Online (Business)	No	No	Standard	Microsoft	<a href="#">Learn</a>
Microsoft Dataverse (legacy)	No	No	Premium	Microsoft	<a href="#">Learn</a> <a href="#">(leg...</a>

Back Next Cancel

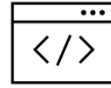
PAD



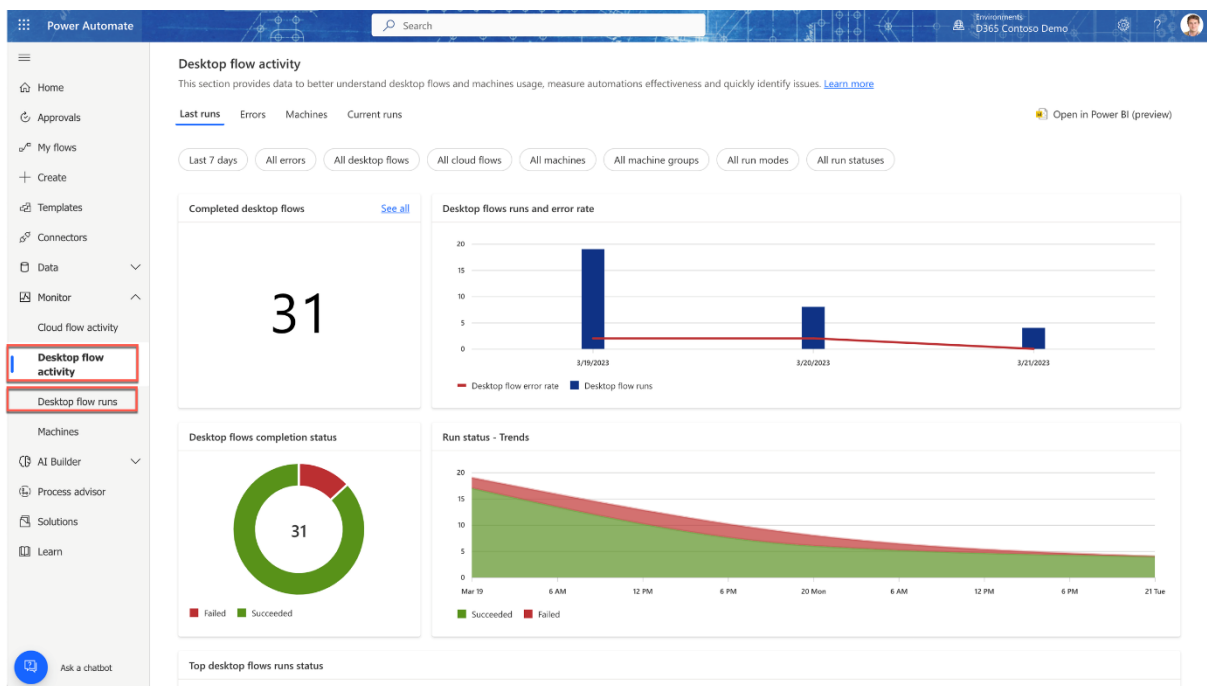
1. Generate access token
2. Access vault and secret
3. Return secret
4. Use credentials for legacy application



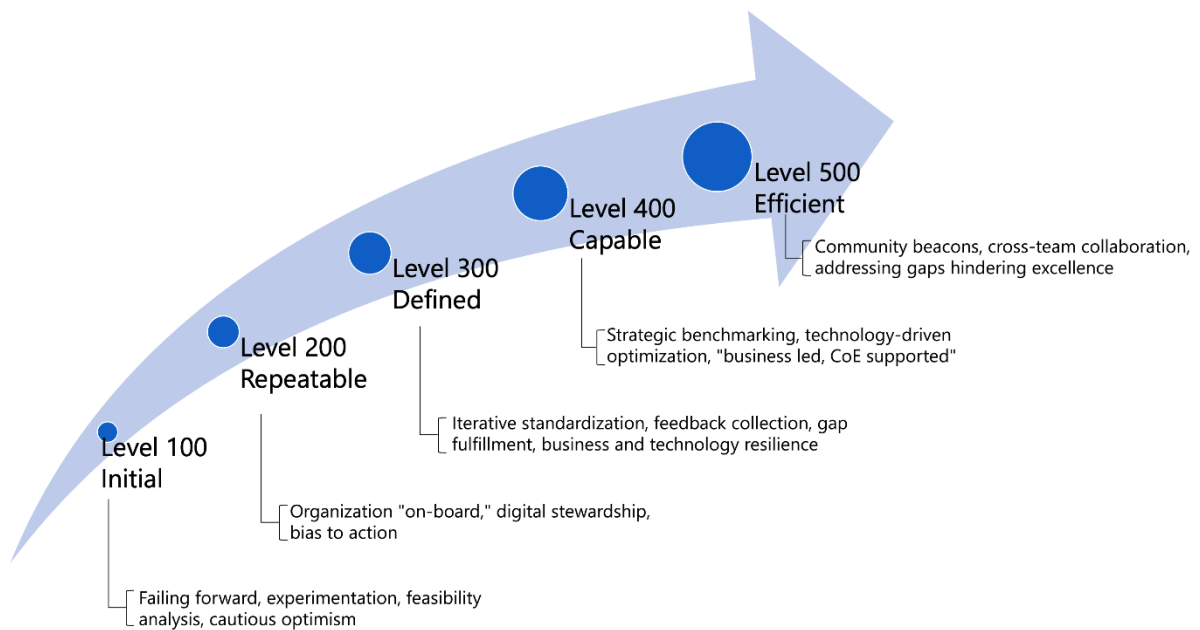
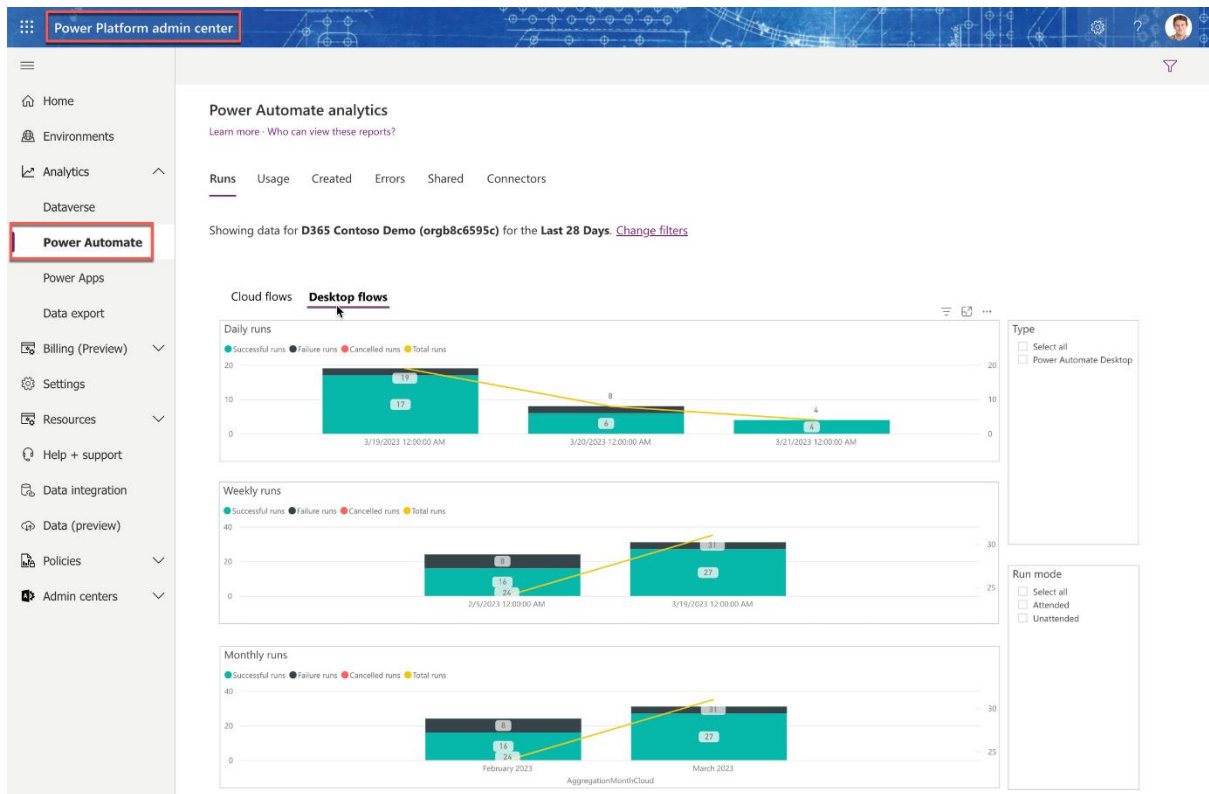
Azure Key Vault

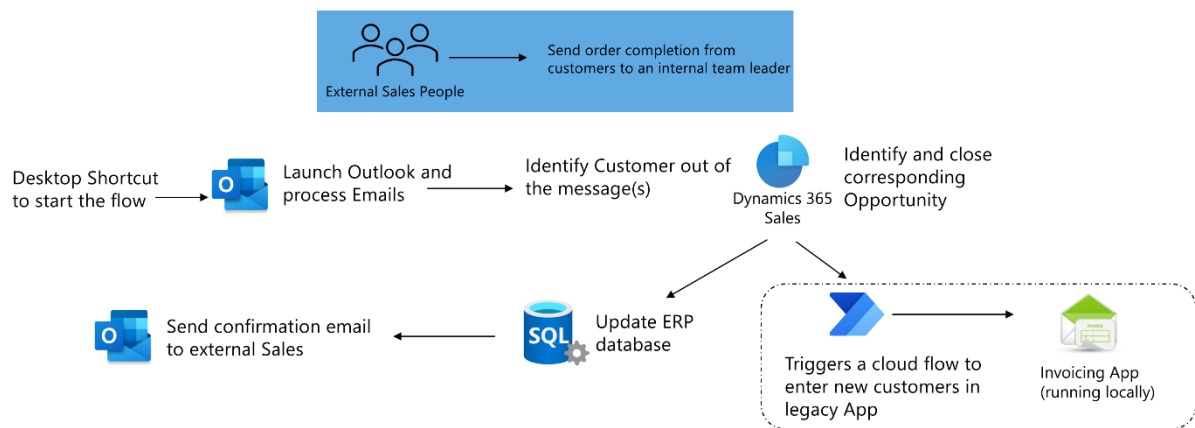
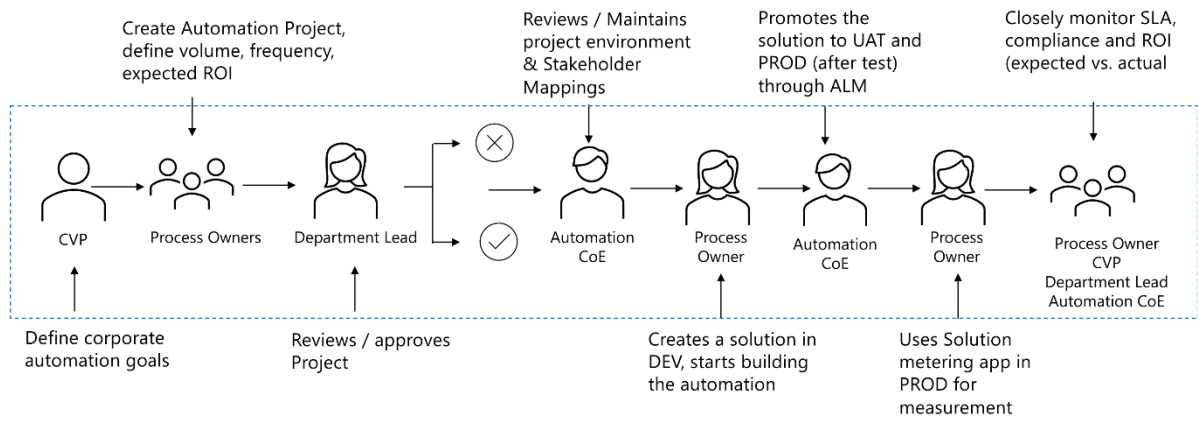


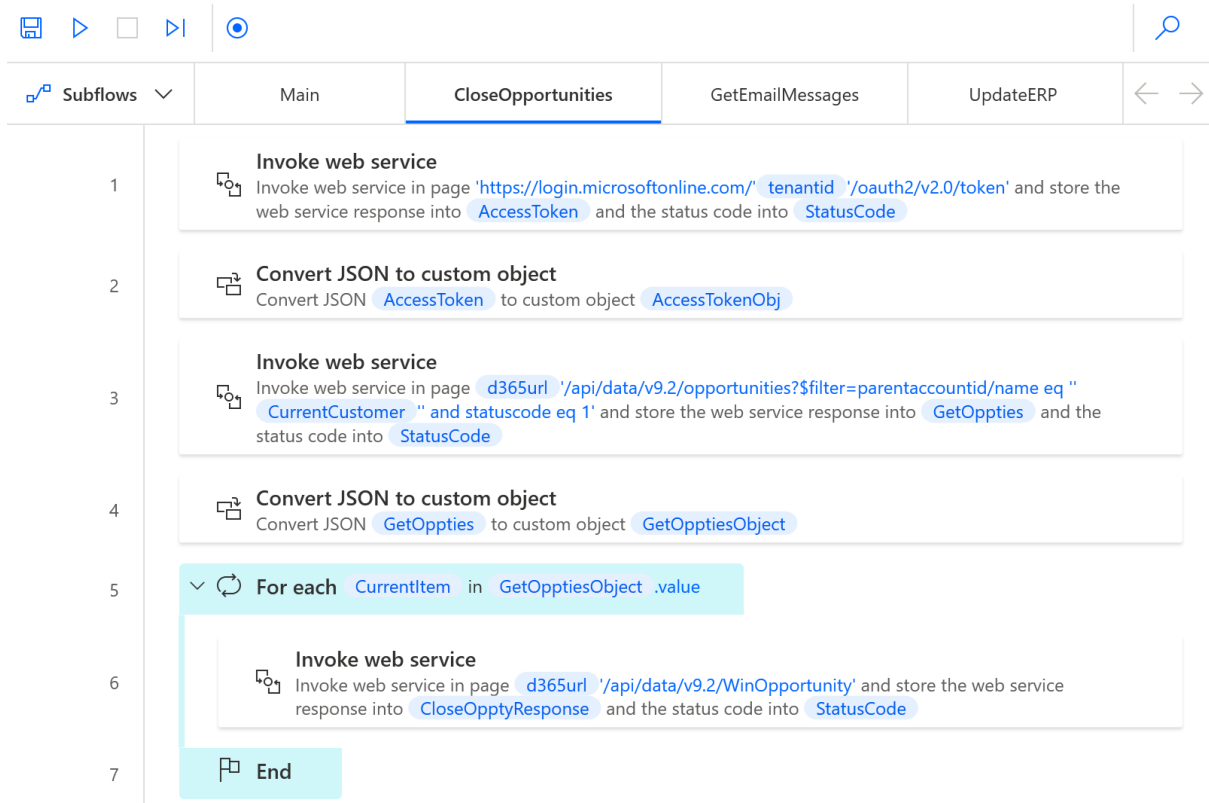
Legacy Application











Machine groups > PAD Demo Group

local installations on VMware Fusion

Overview Machines **Run queue**

Queue	Desktop flow	Requested	Priority	Status	Machine	Run start	Parent flow
	09 - Sales Opportunity to Contoso	25 sec ago	Normal	Running	PADWIN11-B	20 sec ago	Opportunity...
	09 - Sales Opportunity to Contoso	23 sec ago	Normal	Running	PADWIN11-A	17 sec ago	Opportunity...
1	09 - Sales Opportunity to Contoso	20 sec ago	Normal	Next to run	—	—	Opportunity...
2	09 - Sales Opportunity to Contoso	20 sec ago	Normal	Queued	—	—	Opportunity...
3	09 - Sales Opportunity to Contoso	20 sec ago	Normal	Queued	—	—	Opportunity...
4	09 - Sales Opportunity to Contoso	20 sec ago	Normal	Queued	—	—	Opportunity...

**Machines**

make.powerautomate.com/environments/.../machines