

# PowerApps Bootcamp: Expert

Hands-On Lab Guide

Published (Jan 2026)

# Disclaimer

The content, code samples, and guidance provided in this lab document (the “Hands-On Lab Guide”) are for informational and educational purposes only. The Materials are provided “as is” and reflect general practices and platform capabilities at the time of publication. While reasonable efforts have been made to ensure accuracy, no representation or warranty, express or implied, is made as to the completeness, accuracy, reliability, or suitability of the Materials for any purpose.

The Materials are not intended to be used as-is in production environments. It is the sole responsibility of the user to evaluate and test all solutions, configurations, and code examples in a controlled, non-production environment prior to any implementation in a live setting. Use of these Materials in a production environment is done entirely at the user’s own risk.

No warranties or guarantees of any kind are provided, including but not limited to warranties of merchantability, fitness for a particular purpose, non-infringement, or the absence of errors or defects. Under no circumstances shall the authors, contributors, or any affiliated parties be held liable for any damages, including but not limited to direct, indirect, incidental, consequential, or special damages, arising out of or in connection with the use of the Materials.

By using or adapting any portion of the lab document in any capacity, including in production environments, you expressly acknowledge and agree that you are assuming full responsibility and liability for such use, and that you release the authors and affiliated parties from any and all claims or liabilities that may arise as a result.

## Contents

Disclaimer .....	1
Exercise 1: Preparing data and creating a new solution .....	1
Objectives .....	1
Requires .....	1
Estimated time to complete this lab .....	1
Task 1: Importing Invoice Table Schema and Data .....	2
Task 2: Creating a Solution .....	9
Exercise 2: Creating the Canvas App and Designing the Home Screen .....	12
Objectives .....	12
Requires .....	12
Estimated time to complete this lab .....	12
Task 1: Creating the responsive app .....	13
Task 2: Editing Home Screen .....	15
Exercise 3: Connecting to Data and Displaying it .....	20
Objectives .....	20
Estimated time to complete this lab .....	20
Task 1: Connecting to the Dataverse Table .....	20
Task 2: Displaying the Data in a Table Control .....	22
Exercise 4: Filtering the Table .....	27
Objectives .....	27
Estimated time to complete this lab .....	27
Task 1: Creating Filter Controls .....	28
Task 2: Applying Filter .....	31
Exercise 5: Allow Users to Submit New Invoices .....	32
Objectives .....	32
Estimated time to complete this lab .....	32
Task 1: Duplicate the existing Screen .....	33
Task 2: Adding Controls to the Screen .....	35
Task 3: Extracting Values from Receipt Processor .....	40
Task 4: Submitting Extracted Values .....	44
Exercise 6: Creating Message and Loading Popups using Components .....	46
Objectives .....	46
Estimated time to complete this lab .....	46

Task 1: Adding a New Message Popup Component with Custom Properties.....	47
Task 2: Adding Controls into the Component.....	52
Task 3: Adding Message Popup Component to Submission Screen.....	57
Task 4: Adding a New Loading Component.....	60
Task 5: Adding Loading Component to Submission Screen.....	63

# Exercise 1: Preparing data and creating a new solution

In this exercise, we will create a new solution to store all the flows and apps related to our app.

## Objectives

After completing this lab, you will be able to:

- Create a new solution for your Power Platform developments

## Requires

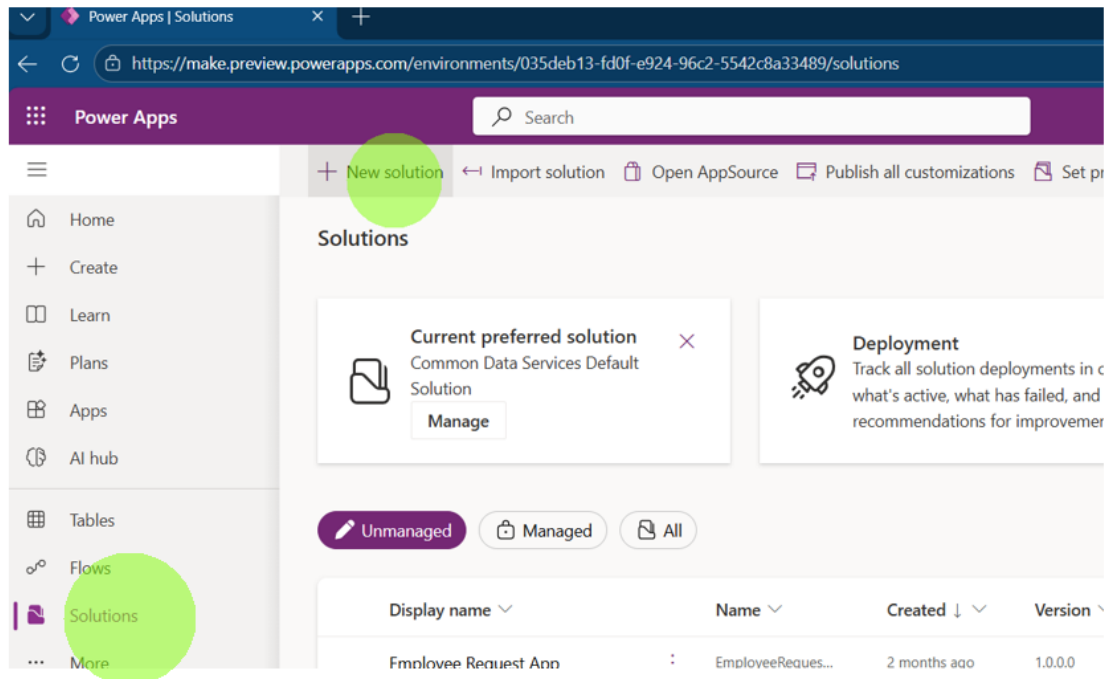
- InvoiceProcessingApplImportData\_1\_0\_0\_1.zip
- invoices\_last3months\_simple\_approval.xlsx

## Estimated time to complete this lab

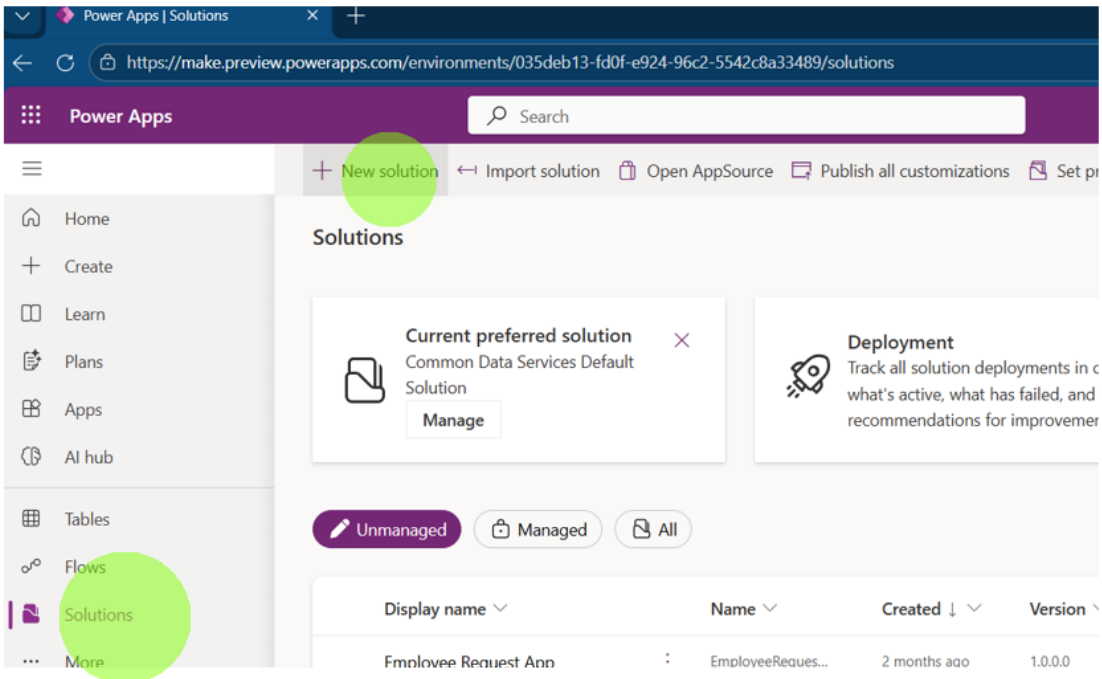
15 mins

## Task 1: Importing Invoice Table Schema and Data

1. Go to [make.powerapps.com](https://make.powerapps.com).



2. Click Solutions.



Power Apps | Solutions

https://make.preview.powerapps.com/environments/035deb13-fd0f-e924-96c2-5542c8a33489/solutions

Power Apps

Search

+ New solution ← Import solution Open AppSource Publish all customizations Set pr

Solutions

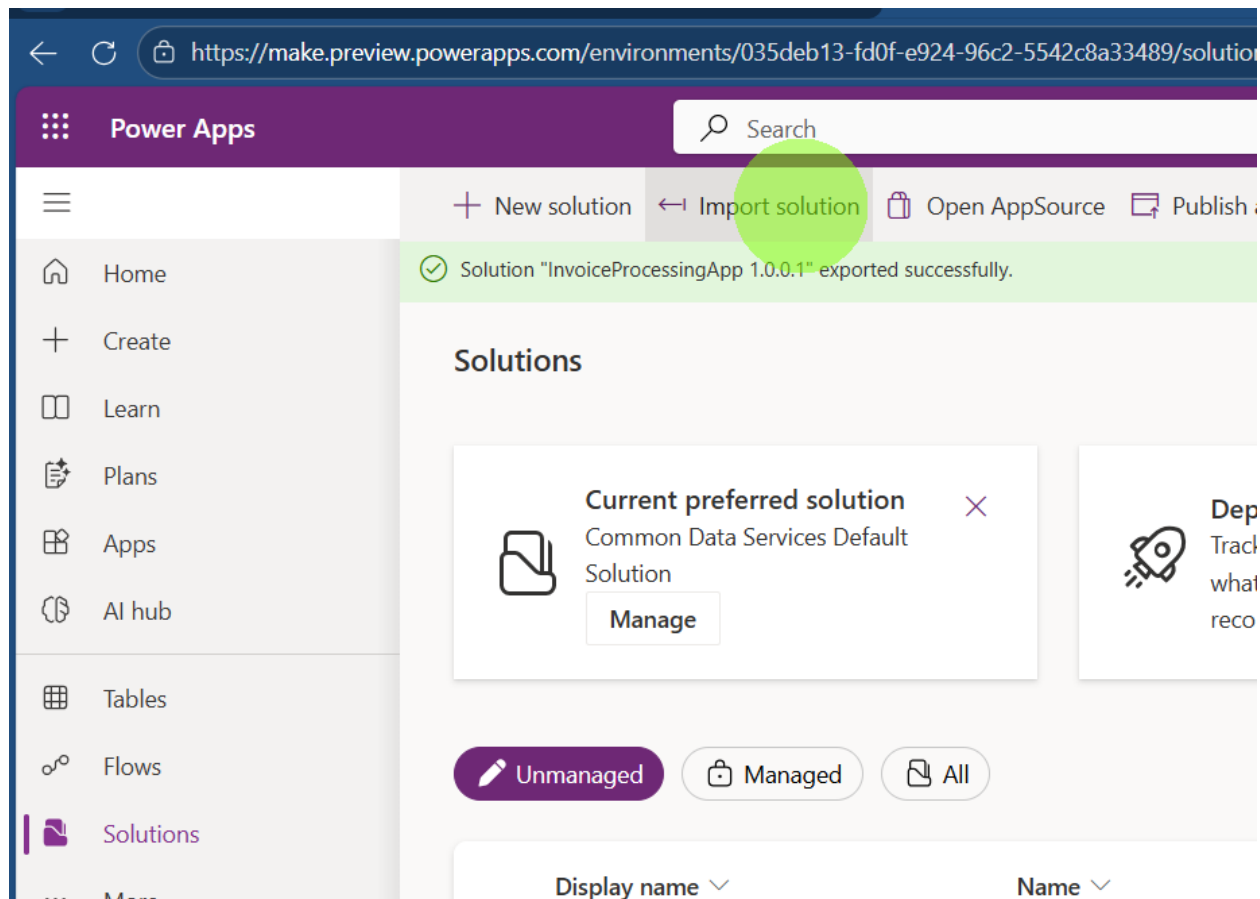
Current preferred solution ×  
Common Data Services Default  
Solution  
[Manage](#)

Deployment  
Track all solution deployments in c  
what's active, what has failed, and  
recommendations for improveme

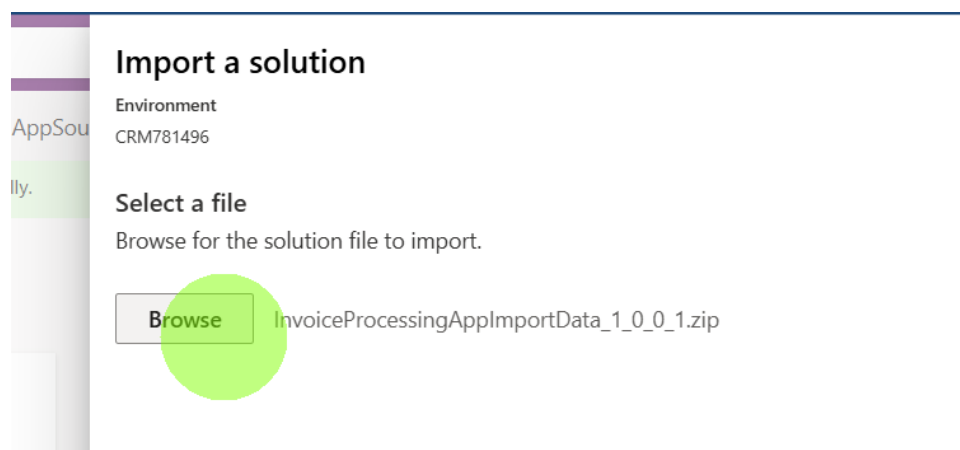
Unmanaged Managed All

Display name ▾	Name ▾	Created ↓ ▾	Version ▾
Employee Request App	EmployeeReques...	2 months ago	1.0.0.0

3. Click Import Solution.



4. Click Browse and select InvoiceProcessingAppImportData\_1\_0\_0\_1.zip from lab files.



5. Click Next and Import. The solution will take awhile to import.



### Import a solution

Environment  
CRM781496

**Select a file**  
Browse for the solution file to import.

**Browse** InvoiceProcessingApplImportData\_1\_0\_0\_1.zip

**Next** Cancel

### Import a solution

Environment  
CRM781496

**Connections**  
Re-establish connections to activate your solution.

☒ Review and adjust all connections

0 updates needed. 2 updates optional.

**Import** Cancel

6. Meanwhile upload invoices\_last3months\_simple\_approval.xlsx to your OneDrive folder.

#### invoices\_last3months\_simple\_approval

You frequently open  
11m ago

**Open**


#### Request-Sample-Data-6050f71d-23fb-440e-83ef

You recently opened  
Jan 19

**Open**

Recent

**All** Word Excel PowerPoint PDF More

Name	Opened	Owner
 invoices_last3months_simple_approval My Files	8m ago	System Administrator

- Once the solution is imported, you will receive a message as below. Click into the imported solution.

Search CRM78149

+ New solution ← Import solution Open AppSource Publish all customizations Set preferred solution See history Connect

① Solution "InvoiceProcessingAppImportData" imported successfully with warnings: The original workflow definition has been deactivated and replaced.

### Solutions

**Current preferred solution** ×

Common Data Services Default Solution

Manage

**Deployment**

Track all solution deployments in one place. See what's active, what has failed, and get recommendations for improvement.

**Create plan**

Select an environment to generate a support plan.

Unmanaged Managed All

Display name ▾	Name ▾	Created ▾	Version ▾	Publisher ▾	Solution checked
InvoiceProcessingAppImportData	InvoiceProcessing...	14 minutes a...	1.0.0.1	CDS Default Publ...	Hasn't been run

- Once inside the solution, open Import Invoice Data Flow to edit in a new tab.

< + New ▾ Add existing ▾ Edit ▾ Details ▾ See analytics Turn off Adv...

✓ Solution "InvoiceProcessingAppImportData" imported successfully.

InvoiceProcessingAppImportData > All

Display name ↑ ▾	Name ▾	Type ▾	Managed ▾
Import Invoice Data	Import Invoice D...	Cloud Flow	No
Invoice			

Edit Details See analytics

Edit Edit in new tab

9. Expand List rows present in a table step and select "My Drive" for Document Library.

**List rows present in a table**

Parameters Settings Code view Testing About

Location \*  
OneDrive for Business

Document Library \*  
OneDrive

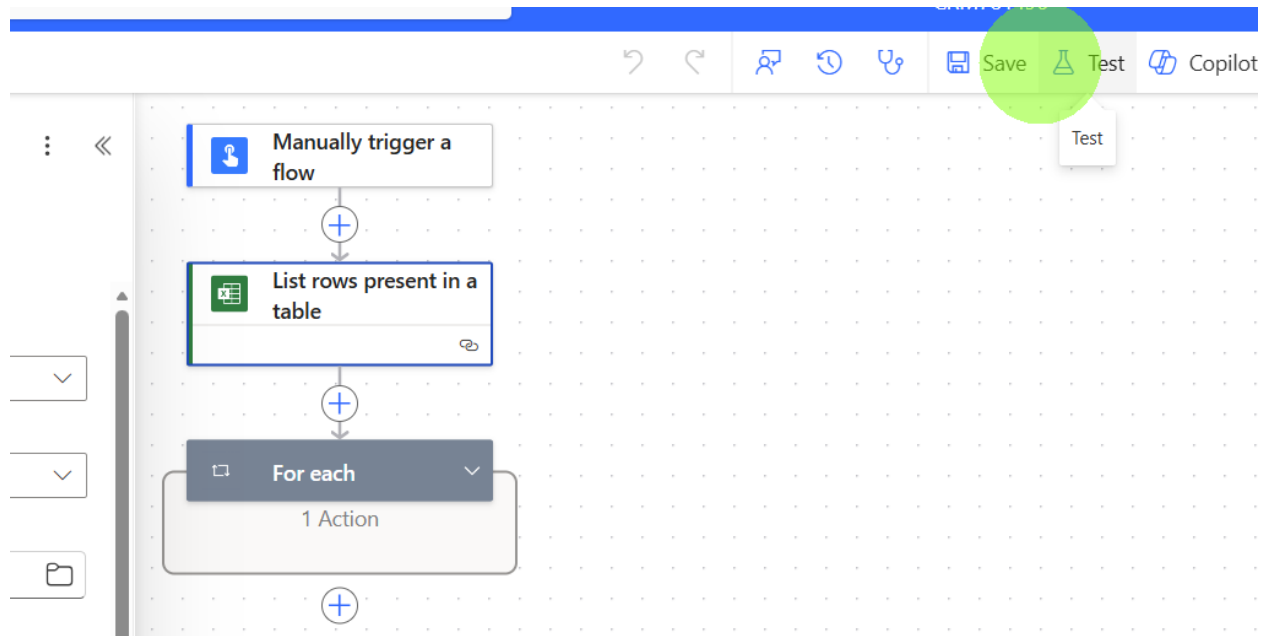
File \*  
/invoices\_last3months\_simple\_approval.xlsx

Table \*  
Table1

Extract Sensitivity Label  
A boolean whether to extract sensitivity label Id for associated artefact.

Sensitivity Label Metadata  
A boolean whether to fetch sensitivity label Metadata for associated LabelId.

10. Save and Test run the flow to import data.



11. Once run, Invoice table should have data inside.

< + New ▾ Add existing ▾ Open ▾ Edit ▾ Import ▾ Export ▾

✓ Solution "InvoiceProcessingAppImportData" imported successfully.

InvoiceProcessingAppImportData > All

	Display name ↑ ▾	Name ▾	Type ▾	N
🔗	Excel Online (Business)	new_sharedexcel...	Connection Refe...	N
🔄	Import Invoice Data	Import Invoice D...	Cloud Flow	N
✓	Invoice	cr61e_invoice	Table	N
🔗	Microsoft Dataverse Inven...	cr3ab_sharedco...	Connection Refe...	N

Invoice columns and data

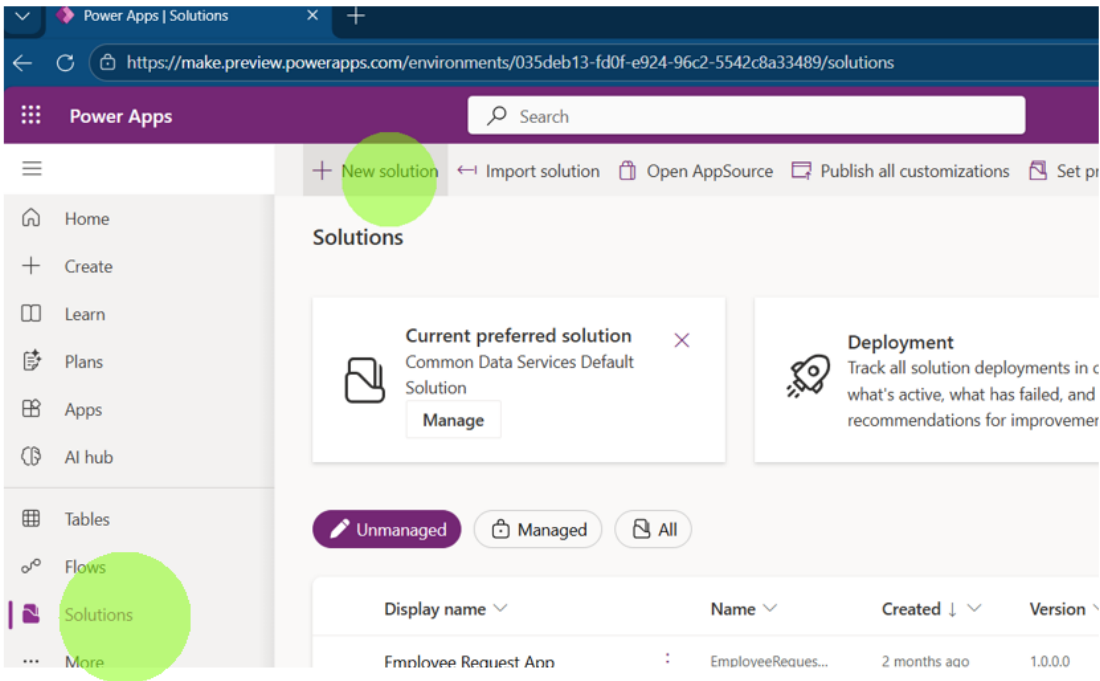
Update forms and views Edit ▾

ApprovalRa... ▾	ApprovalRe... ▾	ApprovalStatus ▾	Amount ▾	Amount (... ▾ %	InvoiceDate	+22 more ▾ +
8/11/2025		Pending	1,544.50	1,544.50	8/11/2025	
8/14/2025	8/16/2025	Rejected	1,126.81	1,126.81	8/12/2025	
8/16/2025		Exception	10,267.26	10,267.26	8/14/2025	
8/17/2025	8/19/2025	Rejected	7,753.84	7,753.84	8/14/2025	
8/21/2025		Exception	10,247.33	10,247.33	8/20/2025	
8/26/2025		Pending	12,585.17	12,585.17	8/25/2025	

----- End of Task -----

## Task 2: Creating a Solution

1. Go to [make.powerapps.com](https://make.powerapps.com).



Power Apps | Solutions

https://make.preview.powerapps.com/environments/035deb13-fd0f-e924-96c2-5542c8a33489/solutions

Power Apps

+ New solution ← Import solution Open AppSource Publish all customizations Set pr

Solutions

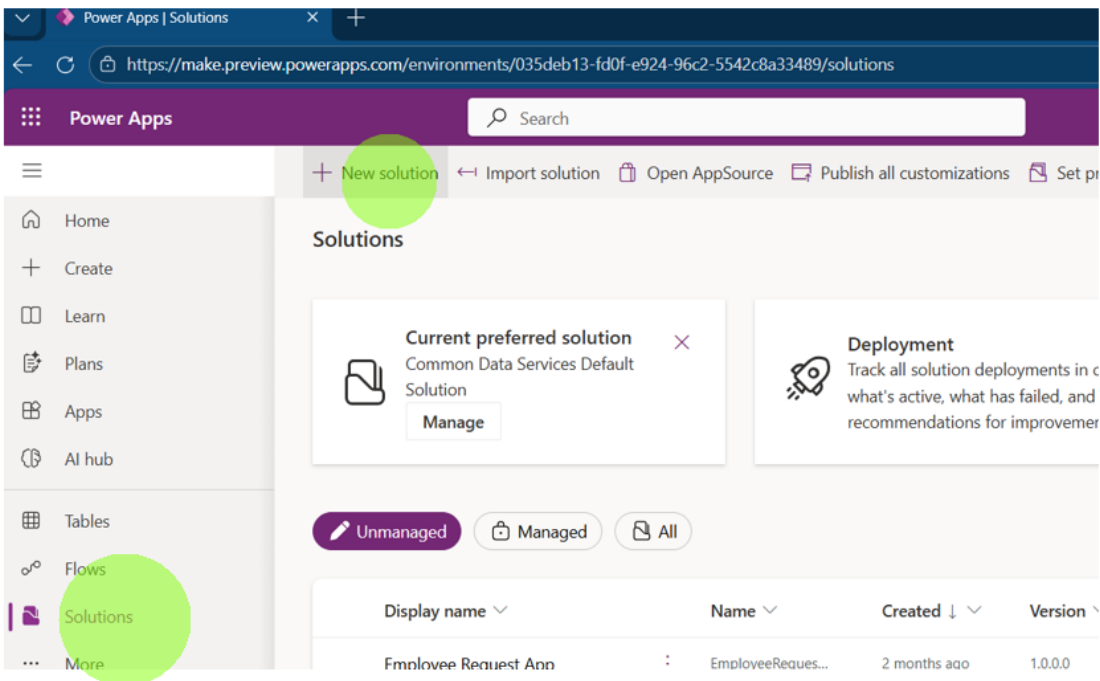
Current preferred solution  
Common Data Services Default Solution  
Manage

Deployment  
Track all solution deployments in c  
what's active, what has failed, and  
recommendations for improveme

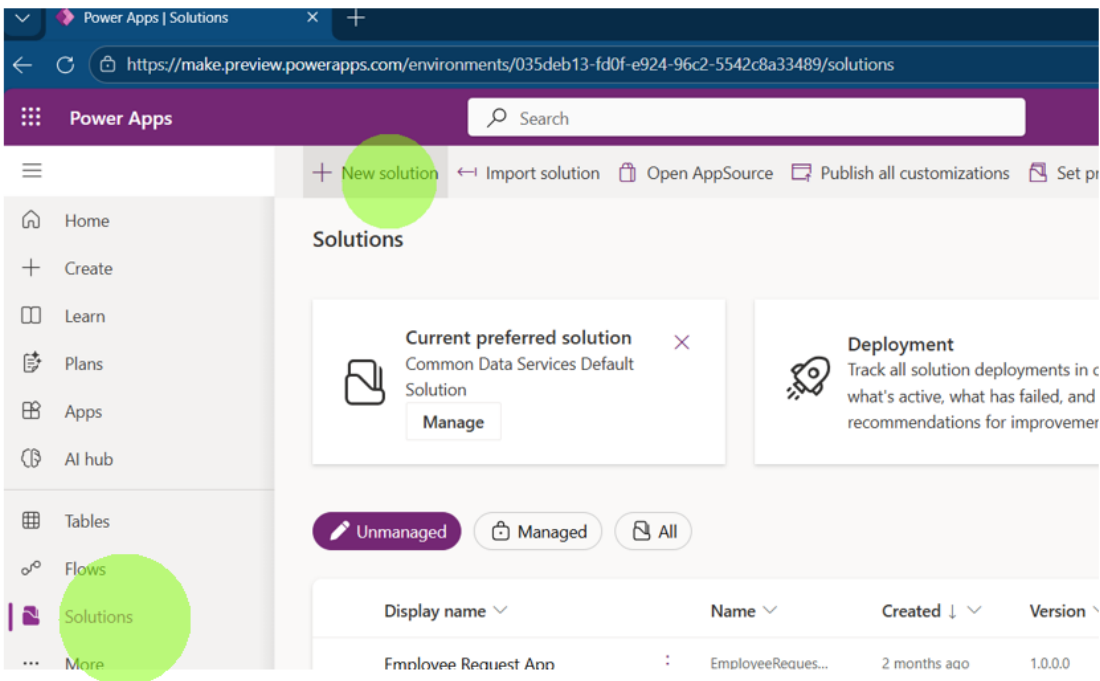
Unmanaged Managed All

Display name ▾	Name ▾	Created ↓ ▾	Version ▾
Employee Request App	EmployeeReques...	2 months ago	1.0.0.0

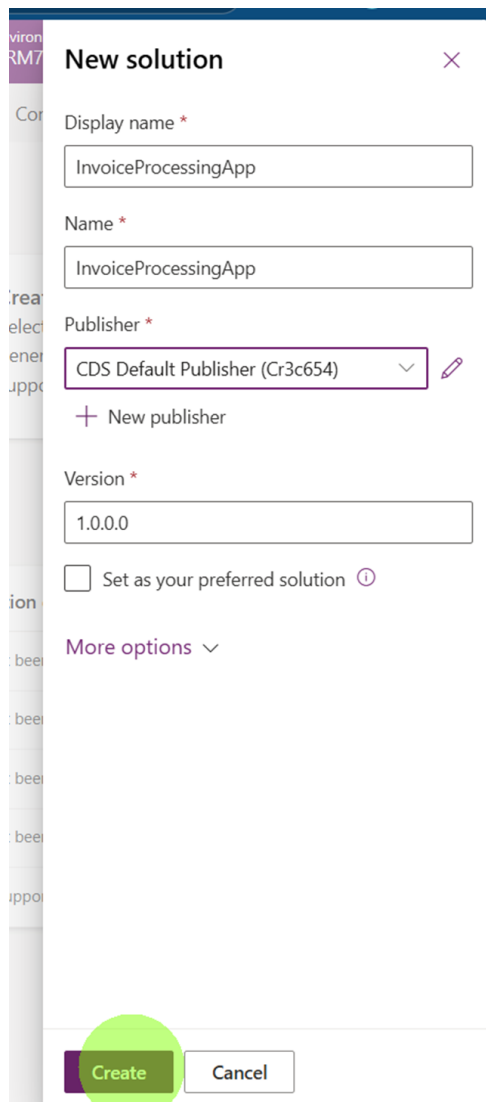
2. Click Solutions.



3. Click + New Solution.




4. Enter Solution Details as below. For Publisher, you will find a default publisher with a different ID to select.



**New solution** ✕

Display name \*  
InvoiceProcessingApp

Name \*  
InvoiceProcessingApp

Publisher \*  
CDS Default Publisher (Cr3c654) ▼   
+ New publisher

Version \*  
1.0.0.0

☐ Set as your preferred solution ⓘ

More options ▼

**Create** Cancel

5. You will be navigated to the empty Solution page.

----- **End of Task** -----

# Exercise 2: Creating the Canvas App and Designing the Home Screen

In this next exercise, we will create the canvas app from the solution and design a home screen for user navigation.

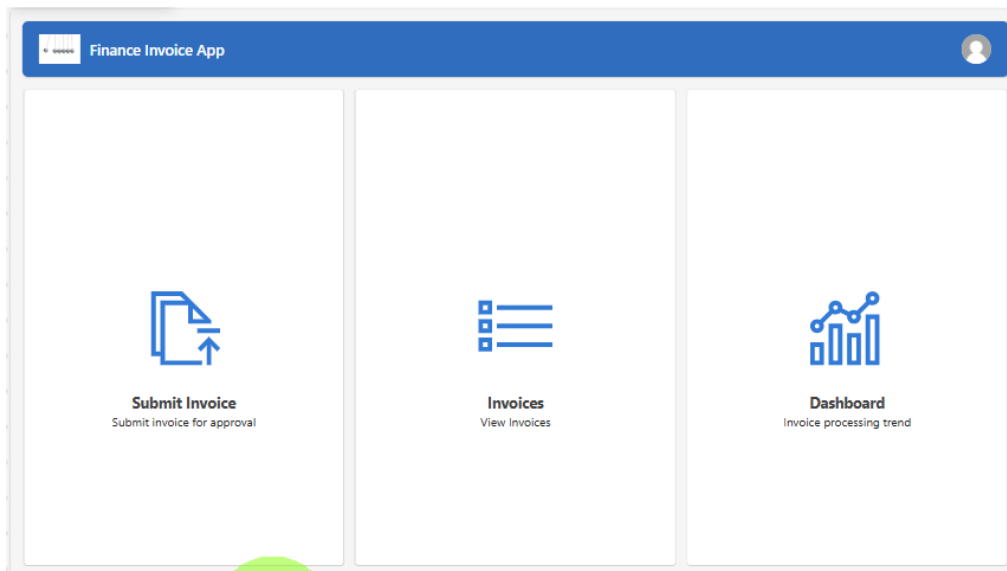
## Objectives

After completing this lab, you will be able to:

- Use built-in responsive layout templates
- Change the app settings to be more responsive

## Requires

- App Icons



## Estimated time to complete this lab

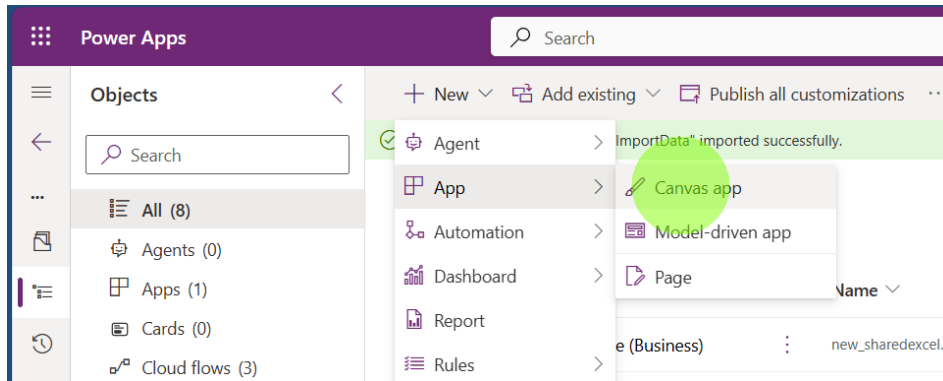
20 mins



## Task 1: Creating the responsive app

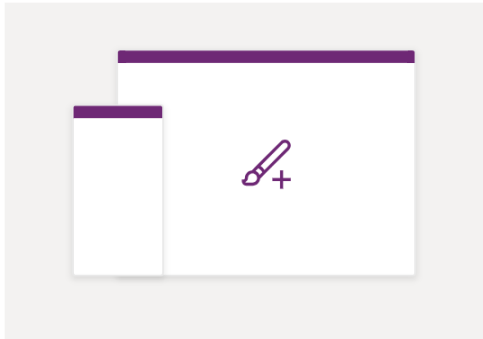
In this Task you will learn how to create a new app from scratch and how to make it responsive.

1. In solution, click +New > App > Canvas app




Provide a name and click Create.

### Canvas app from blank



Design the app you want, and connect it to hundreds of data sources.

 Canvas app

App name \*

Format

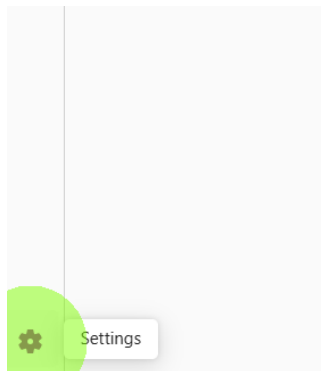
☒ Tablet

☐ Phone

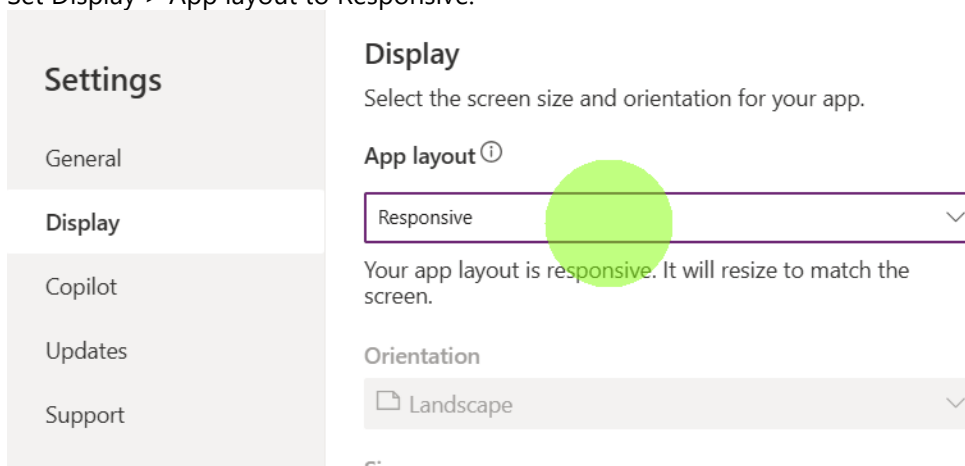
Create

Cancel

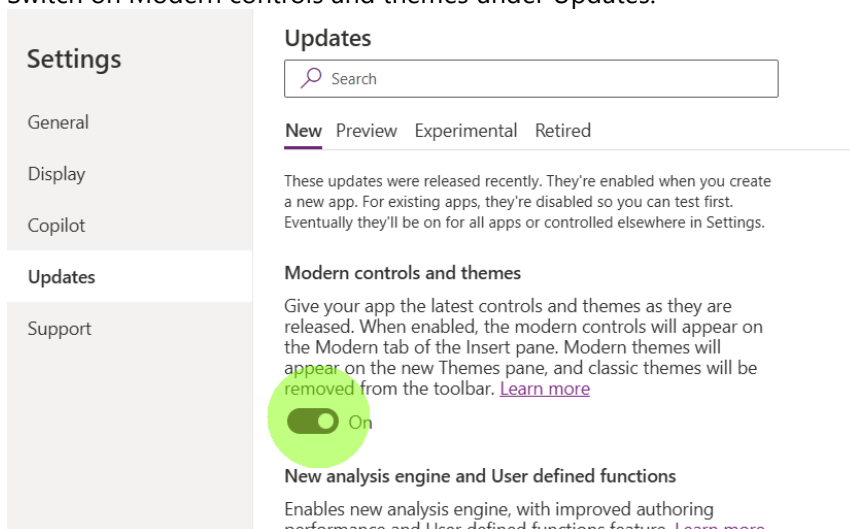
2. Once in the app designer, click setting icon at the bottom right corner.



Set Display > App layout to Responsive.



Switch on Modern controls and themes under Updates.



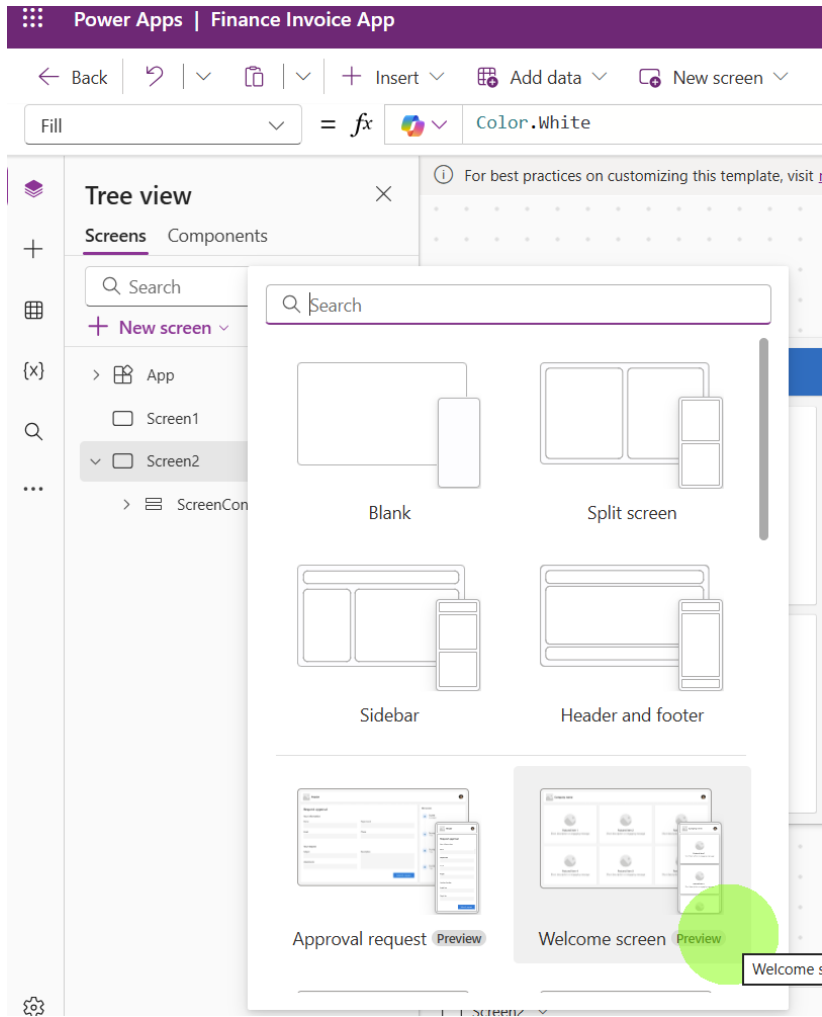
Close the Settings window.

----- End of Task -----

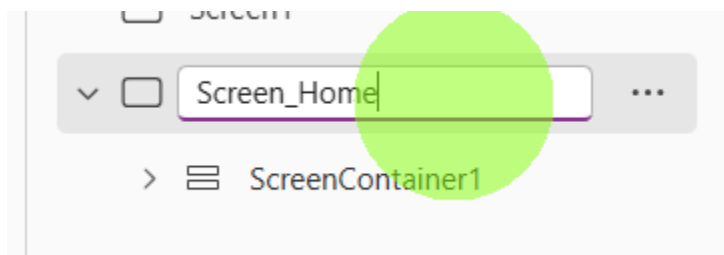
## Task 2: Editing Home Screen

In this Task you will learn how to use built-in Power App screens.

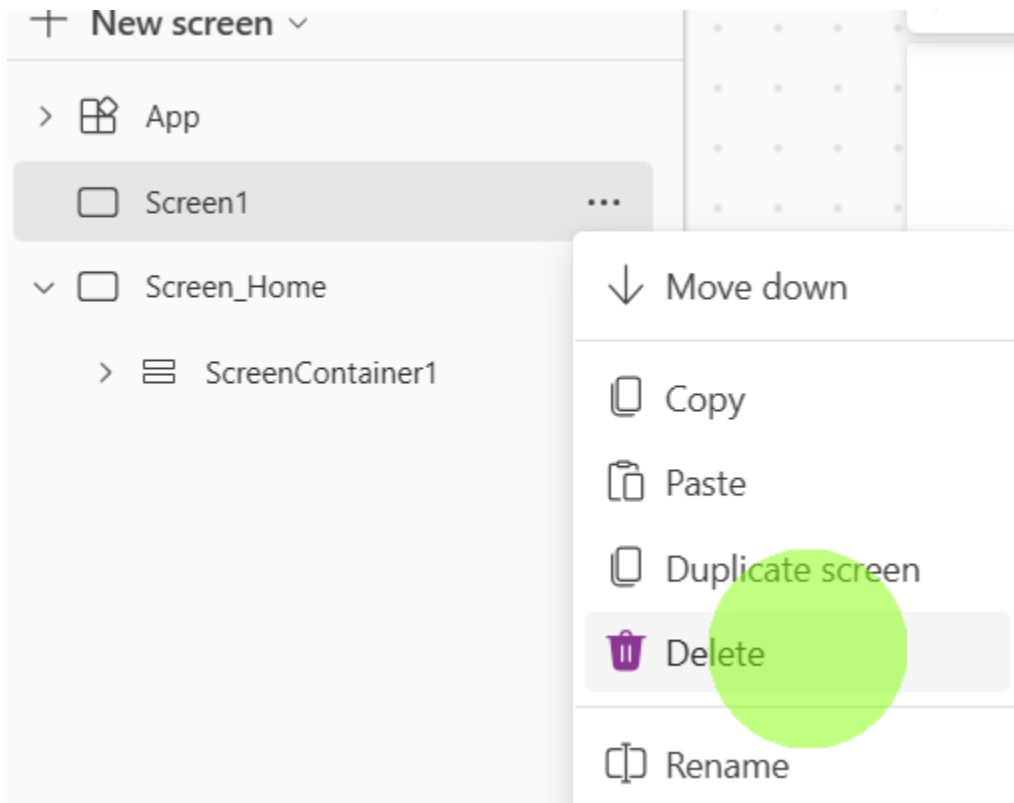
1. Click + New screen > Welcome screen.



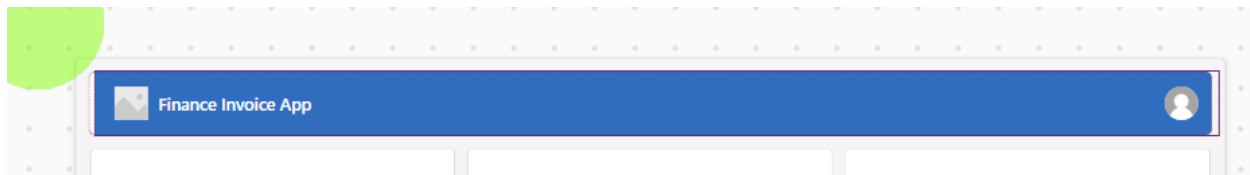
Rename the screen to Screen\_Home.



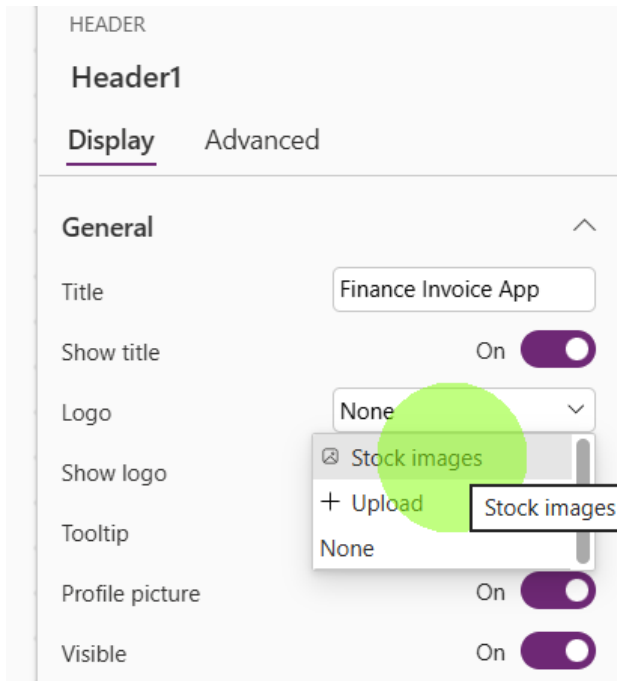
Delete Screen1.



2. Select Header and change Title to Finance Invoice App.

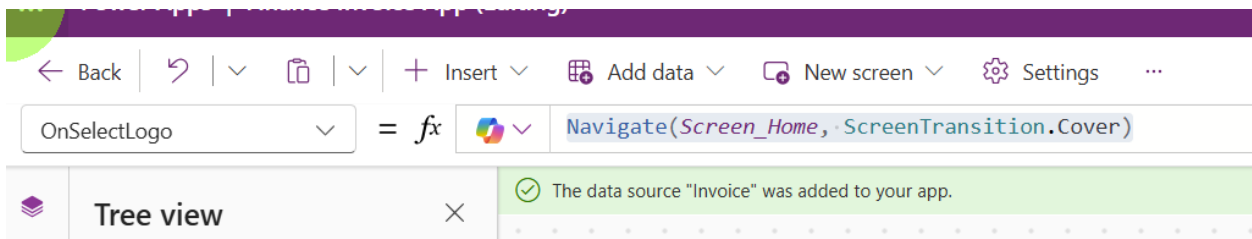


Click Logo and select any stock image that you like.

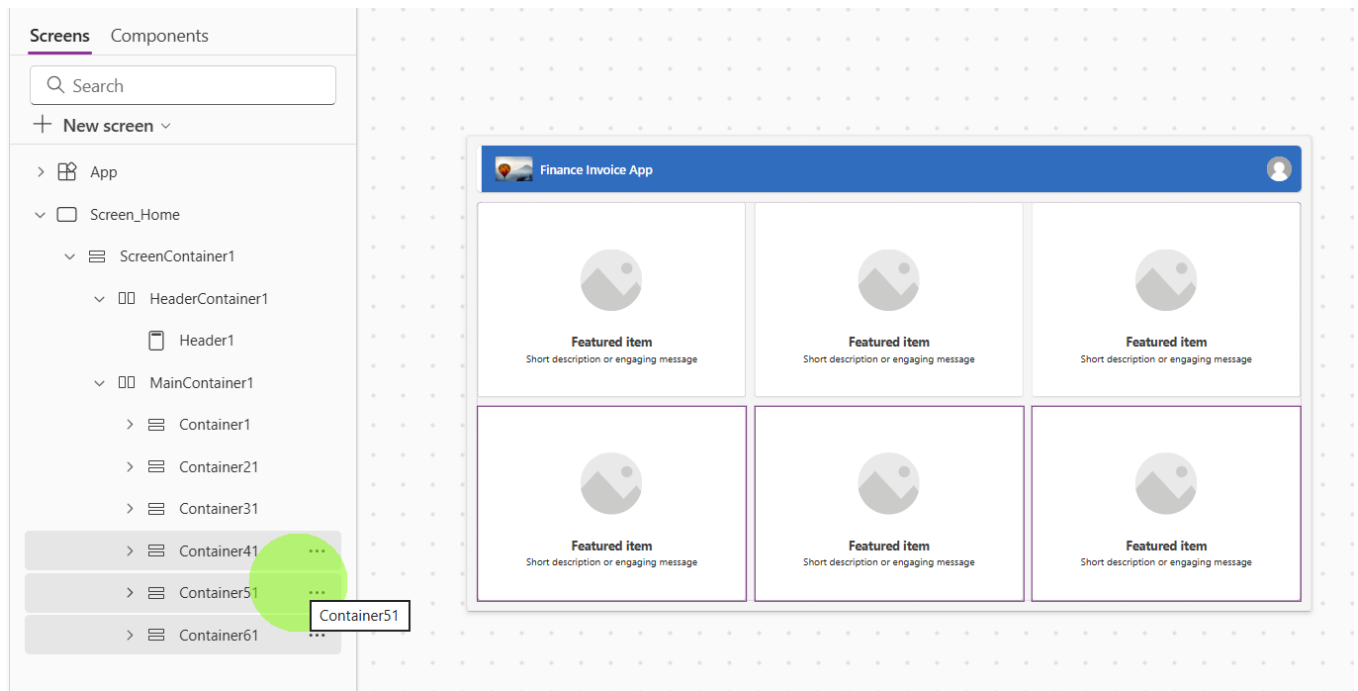


3. Change the OnSelectLogo property of Header to below.

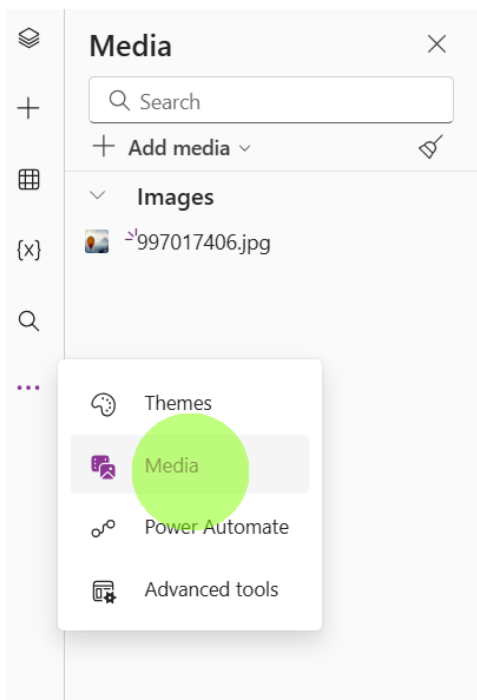
`Navigate(Screen_Home, ScreenTransition.Cover)`






4. Expand Main Container from Tree View. Select and delete the bottom 3 containers.



5. Upload Icons to the app by clicking Media > + Add media



Upload these 3 icons from Upload Icons folder.

Name	Status	Date modified	Type	Size
 Dashboard.svg	✓	18/1/2026 5:47 pm	Microsoft Edge HT...	2 KB
 List.svg	✓	18/1/2026 5:48 pm	Microsoft Edge HT...	1 KB
 Upload.svg	✓	18/1/2026 5:45 pm	Microsoft Edge HT...	1 KB

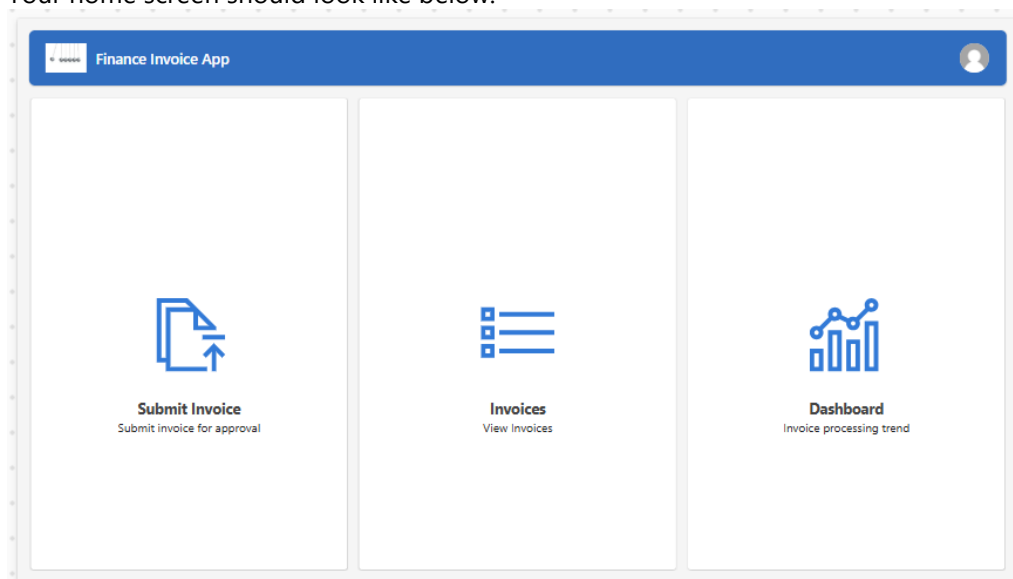
6. Select each icon on Home screen and update Image properties accordingly.

	Image Property	Border Radius
<b>First Icon</b>	Upload	0
<b>Second Icon</b>	List	0
<b>Third Icon</b>	Dashboard	0

7. Select the Text controls under each image and update accordingly.

	Title	Description
<b>First Icon</b>	Submit Invoice	Submit invoice for approval
<b>Second Icon</b>	Invoices	View Invoices
<b>Third Icon</b>	Dashboard	Invoice processing trend

Your home screen should look like below.

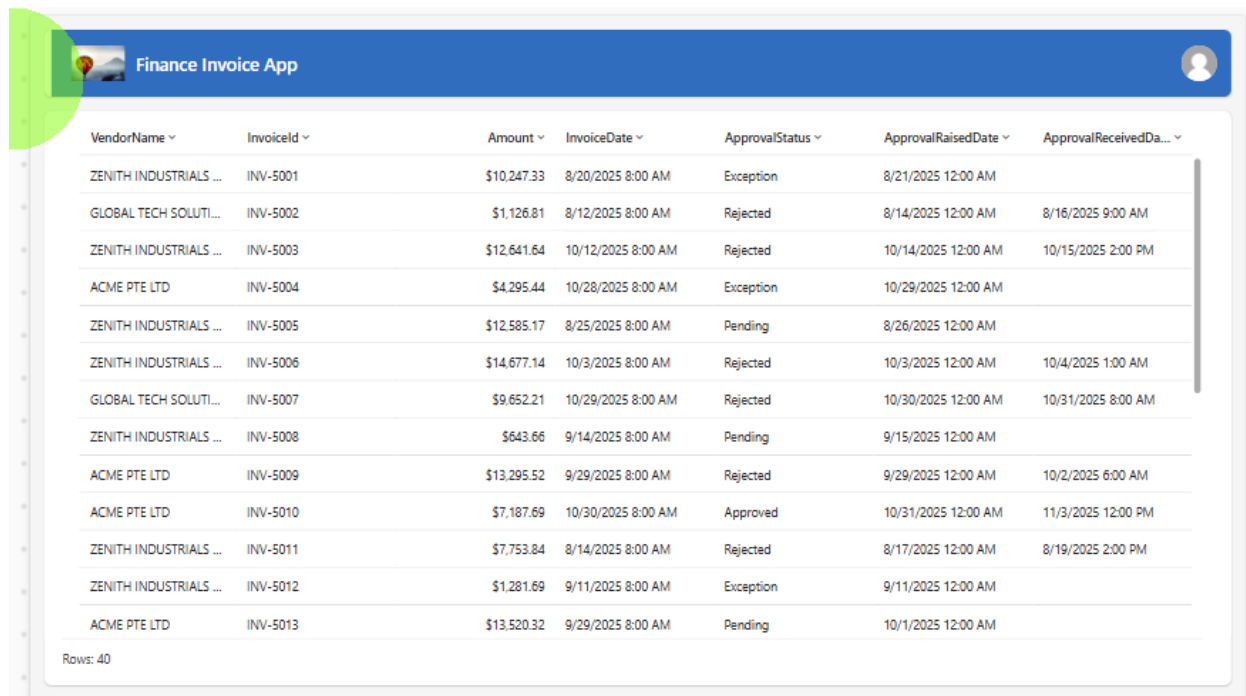


**Important! Remember to Save the App.**

**End of Task**

# Exercise 3: Connecting to Data and Displaying it

In this exercise, we will be displaying data in the app from the Invoices table that we created in exercise 1. We will then customize the look and feel of the data being displayed to create a nice UI format.



The screenshot shows a mobile app interface titled "Finance Invoice App". It features a table with 7 columns: VendorName, InvoiceId, Amount, InvoiceDate, ApprovalStatus, ApprovalRaisedDate, and ApprovalReceivedDate. The table contains 13 rows of data. A green circular highlight is on the left side of the app interface. At the bottom left, it says "Rows: 40".

VendorName	InvoiceId	Amount	InvoiceDate	ApprovalStatus	ApprovalRaisedDate	ApprovalReceivedDate
ZENITH INDUSTRIALS ...	INV-5001	\$10,247.33	8/20/2025 8:00 AM	Exception	8/21/2025 12:00 AM	
GLOBAL TECH SOLUTI...	INV-5002	\$1,126.81	8/12/2025 8:00 AM	Rejected	8/14/2025 12:00 AM	8/16/2025 9:00 AM
ZENITH INDUSTRIALS ...	INV-5003	\$12,641.64	10/12/2025 8:00 AM	Rejected	10/14/2025 12:00 AM	10/15/2025 2:00 PM
ACME PTE LTD	INV-5004	\$4,295.44	10/28/2025 8:00 AM	Exception	10/29/2025 12:00 AM	
ZENITH INDUSTRIALS ...	INV-5005	\$12,585.17	8/25/2025 8:00 AM	Pending	8/26/2025 12:00 AM	
ZENITH INDUSTRIALS ...	INV-5006	\$14,677.14	10/3/2025 8:00 AM	Rejected	10/3/2025 12:00 AM	10/4/2025 1:00 AM
GLOBAL TECH SOLUTI...	INV-5007	\$9,652.21	10/29/2025 8:00 AM	Rejected	10/30/2025 12:00 AM	10/31/2025 8:00 AM
ZENITH INDUSTRIALS ...	INV-5008	\$643.66	9/14/2025 8:00 AM	Pending	9/15/2025 12:00 AM	
ACME PTE LTD	INV-5009	\$13,295.52	9/29/2025 8:00 AM	Rejected	9/29/2025 12:00 AM	10/2/2025 6:00 AM
ACME PTE LTD	INV-5010	\$7,187.69	10/30/2025 8:00 AM	Approved	10/31/2025 12:00 AM	11/3/2025 12:00 PM
ZENITH INDUSTRIALS ...	INV-5011	\$7,753.84	8/14/2025 8:00 AM	Rejected	8/17/2025 12:00 AM	8/19/2025 2:00 PM
ZENITH INDUSTRIALS ...	INV-5012	\$1,281.69	9/11/2025 8:00 AM	Exception	9/11/2025 12:00 AM	
ACME PTE LTD	INV-5013	\$13,520.32	9/29/2025 8:00 AM	Pending	10/1/2025 12:00 AM	

## Objectives

The exercise will achieve the following objectives.

- Learn how to connect to a Dataverse table
- Display data in a Table control
- Filter data displayed in Table control

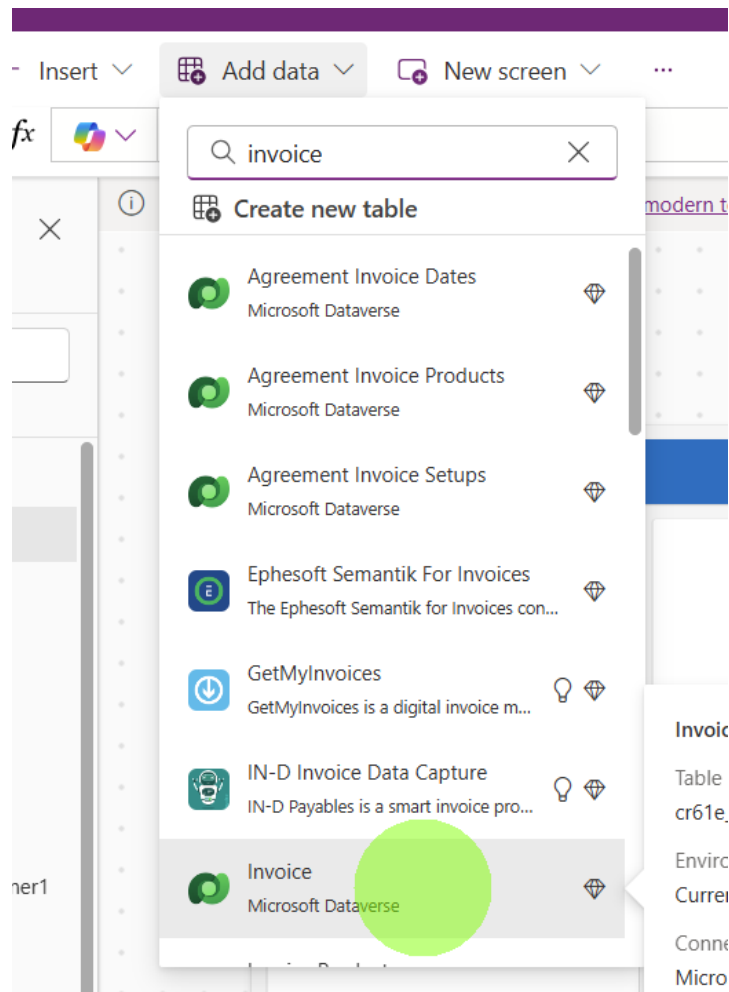
## Estimated time to complete this lab

20 mins

## Task 1: Connecting to the Dataverse Table



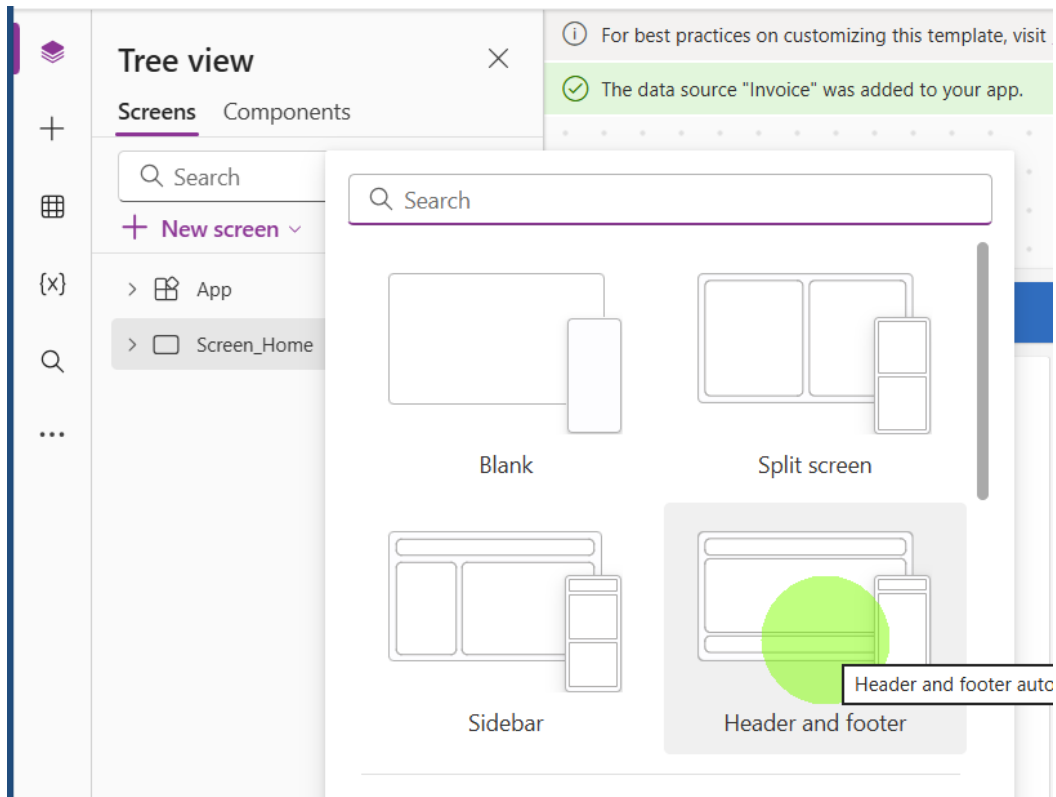
1. Click on Add data and locate the Invoice table and click.



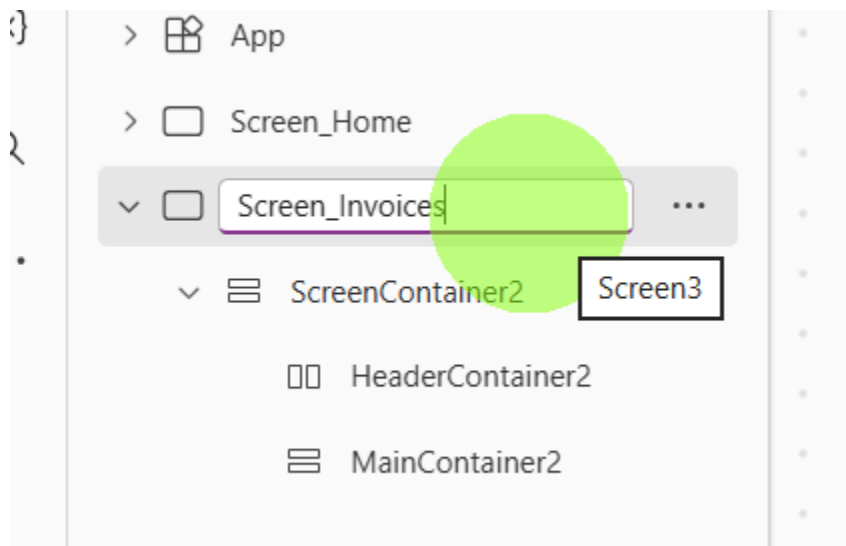
----- End of Task -----

## Task 2: Displaying the Data in a Table Control

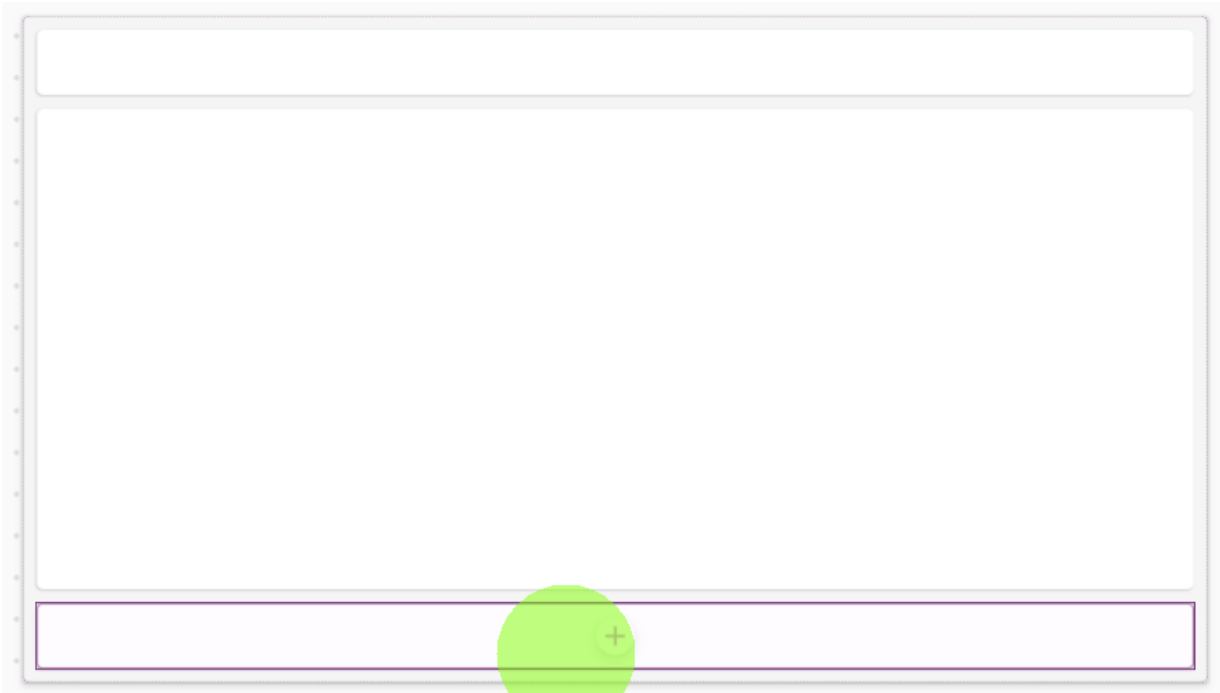
1. Click New screen > Header and footer to add a new screen.



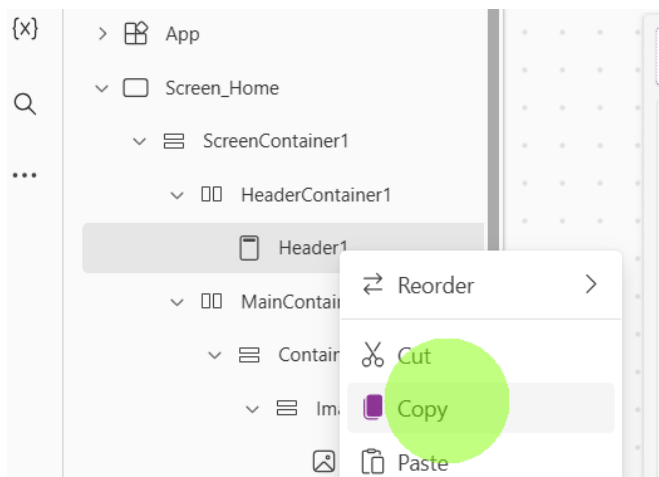
2. Rename the screen to Screen\_Invoices.

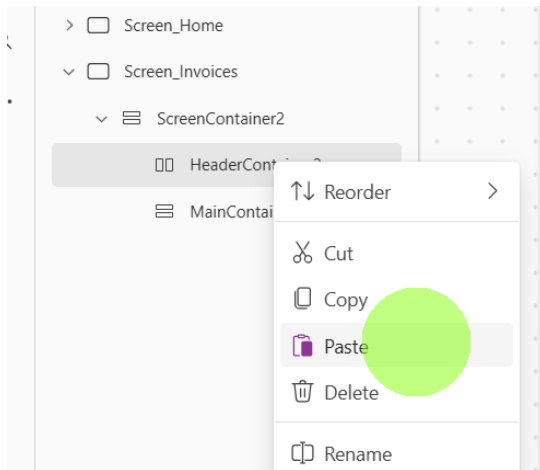


3. Select the bottom container and delete.

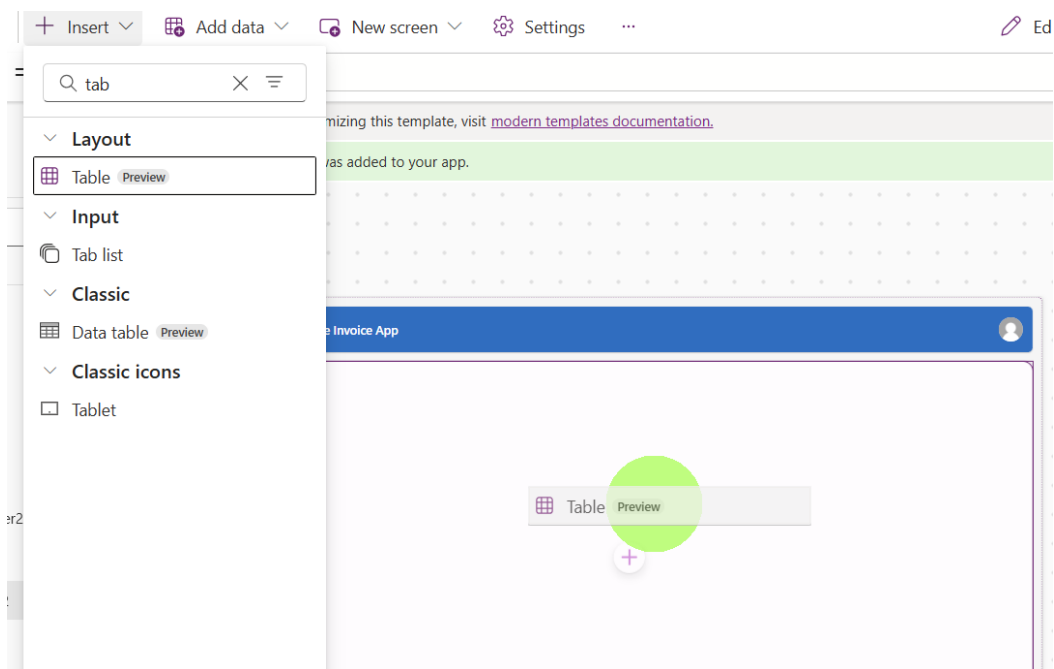


4. Copy Header1 from Screen\_Home and paste in Screen\_Invoices's Header Container.

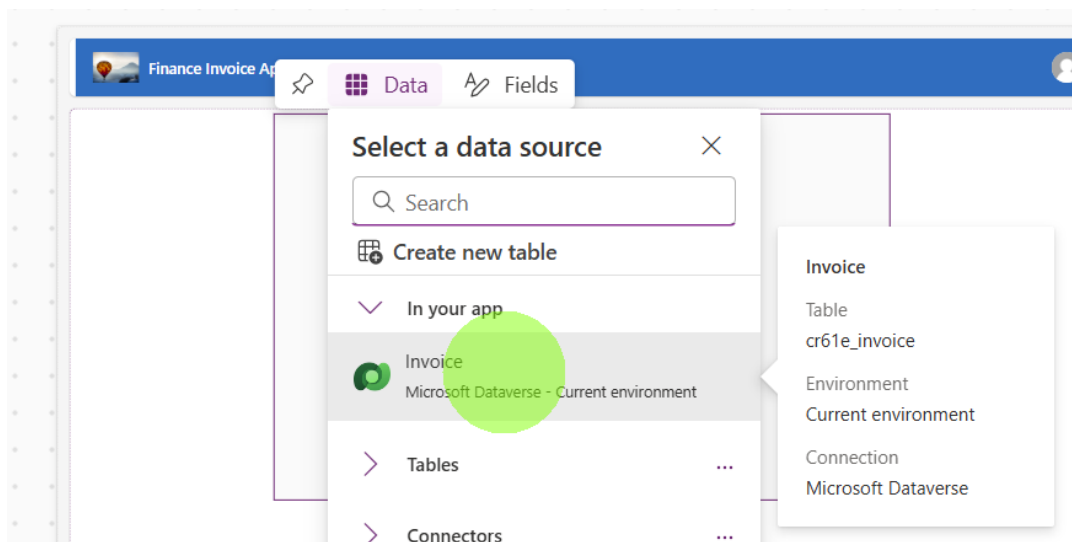




8. Click Insert > Table into the MainContainer.

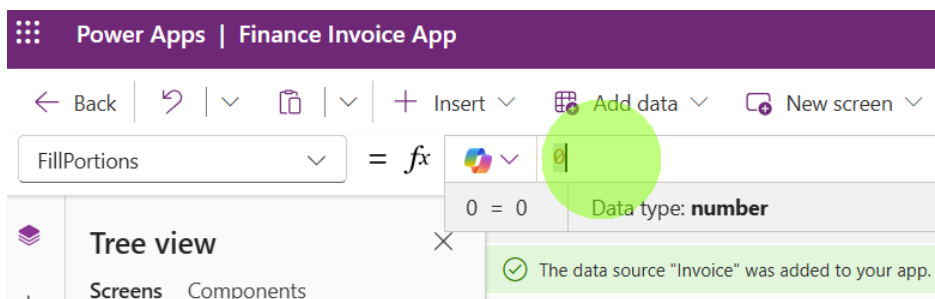
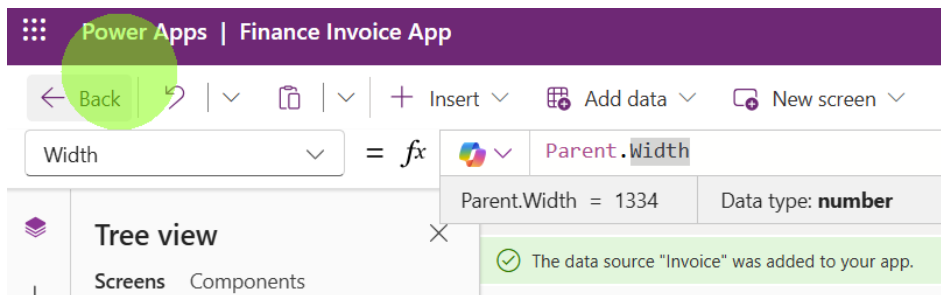


Select Invoice table as data source for the table.



9. Change below properties of table.

Width: **Parent.Width**  
FillPortions: 1



10. Change the fields displayed by clicking on Edit beside Fields.

Properties

TABLE

Table1

Display Advanced

General

Items(Items) Invoice

Views None

Fields

No data text No value

Visible On

Date time format Short date time

Date only format Short date

Edit

Edit the fields list to only keep these fields.

Data

+ Add field

Suggested by AI. Please review. [See terms](#)

VendorName

Invoiceld

Amount

InvoiceDate

ApprovalStatus

ApprovalRaisedDate

ApprovalReceivedDate

More actions...

----- End of Task -----

# Exercise 4: Filtering the Table

In this exercise, you will learn how to perform filtering and on the table to quickly find information when needed.

ScreenContainer7\_1

Finance Invoice App

Vendor Name All Invoice Date (From) Select a date... Invoice Date (To) Select a date...

VendorName	InvoiceId	InvoiceDate	Amount	ApprovalStatus	ApprovalRaisedDate	ApprovalReceivedDa...
ZENITH INDUSTRIALS ...	INV-5001	8/20/2025 8:00 AM	\$10,247.33	Exception	8/21/2025 12:00 AM	
GLOBAL TECH SOLUTI...	INV-5002	8/12/2025 8:00 AM	\$1,126.81	Rejected	8/14/2025 12:00 AM	8/16/2025 9:00 AM
ZENITH INDUSTRIALS ...	INV-5003	10/12/2025 8:00 AM	\$12,641.64	Rejected	10/14/2025 12:00 AM	10/15/2025 2:00 PM
ACME PTE LTD	INV-5004	10/28/2025 8:00 AM	\$4,295.44	Exception	10/29/2025 12:00 AM	
ZENITH INDUSTRIALS ...	INV-5005	8/25/2025 8:00 AM	\$12,585.17	Pending	8/26/2025 12:00 AM	
ZENITH INDUSTRIALS ...	INV-5006	10/3/2025 8:00 AM	\$14,677.14	Rejected	10/3/2025 12:00 AM	10/4/2025 1:00 AM
GLOBAL TECH SOLUTI...	INV-5007	10/29/2025 8:00 AM	\$9,652.21	Rejected	10/30/2025 12:00 AM	10/31/2025 8:00 AM
ZENITH INDUSTRIALS ...	INV-5008	9/14/2025 8:00 AM	\$643.66	Pending	9/15/2025 12:00 AM	
ACME PTE LTD	INV-5009	9/29/2025 8:00 AM	\$13,295.52	Rejected	9/29/2025 12:00 AM	10/2/2025 6:00 AM
ACME PTE LTD	INV-5010	10/30/2025 8:00 AM	\$7,187.69	Approved	10/31/2025 12:00 AM	11/3/2025 12:00 PM
ZENITH INDUSTRIALS ...	INV-5011	8/14/2025 8:00 AM	\$7,753.84	Rejected	8/17/2025 12:00 AM	8/19/2025 2:00 PM
ZENITH INDUSTRIALS ...	INV-5012	9/11/2025 8:00 AM	\$1,281.69	Exception	9/11/2025 12:00 AM	

Rows: 40

## Objectives

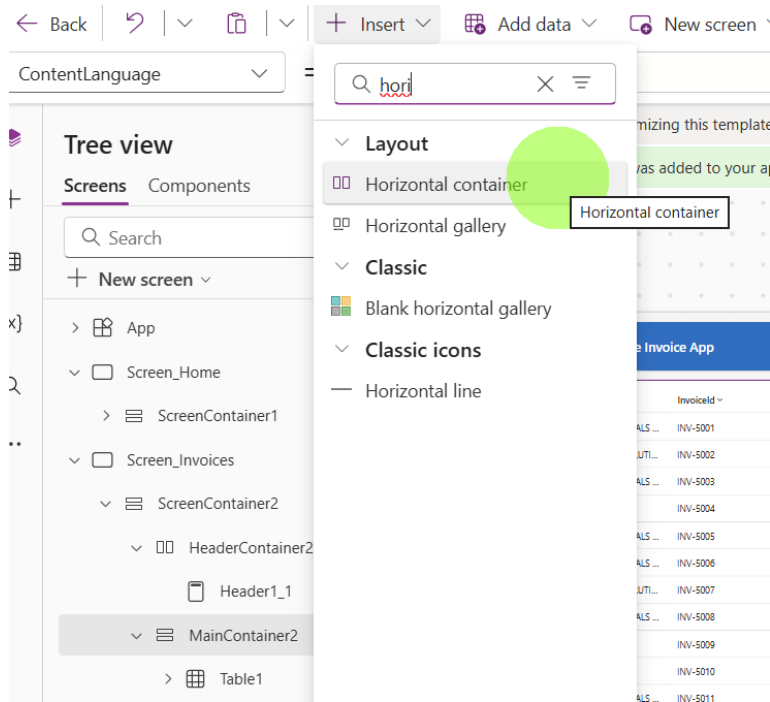
- Filter a table by Vendor Name
- Filter a table by Date Range

## Estimated time to complete this lab

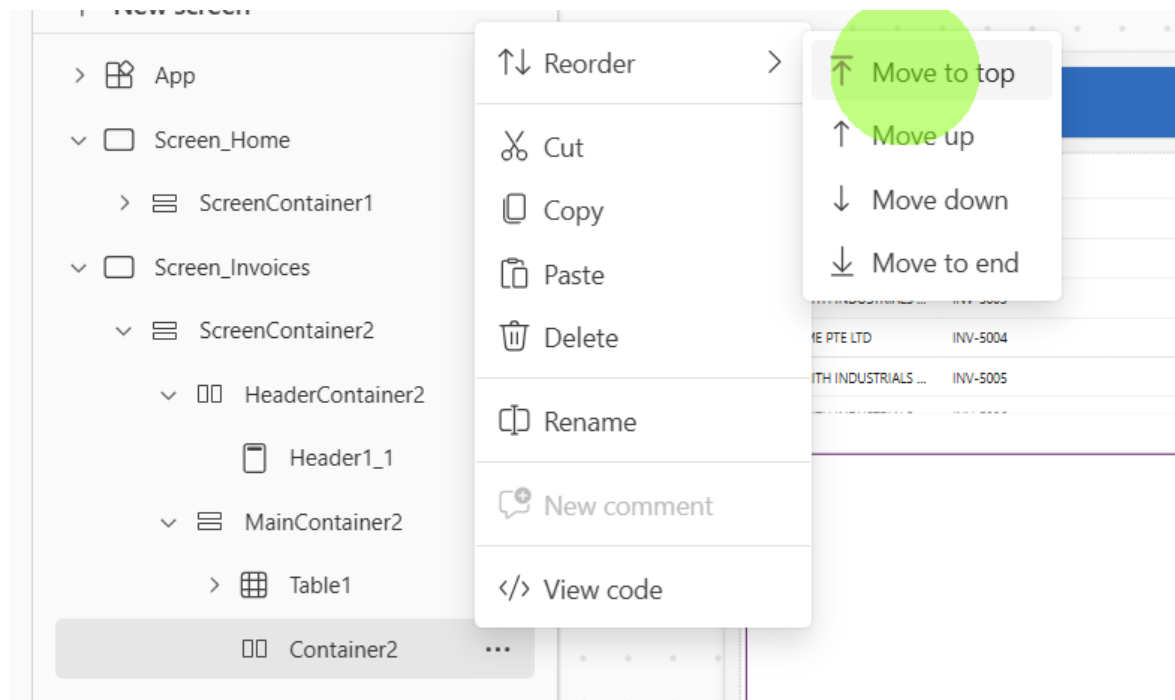
20 mins

## Task 1: Creating Filter Controls

1. Insert a new Horizontal Container to Main Container.

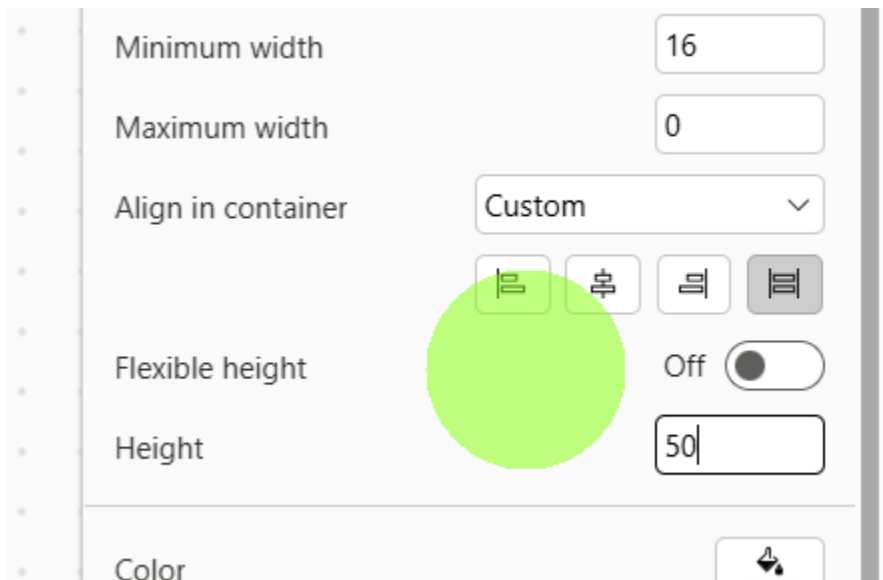


Reorder the new Container to top.



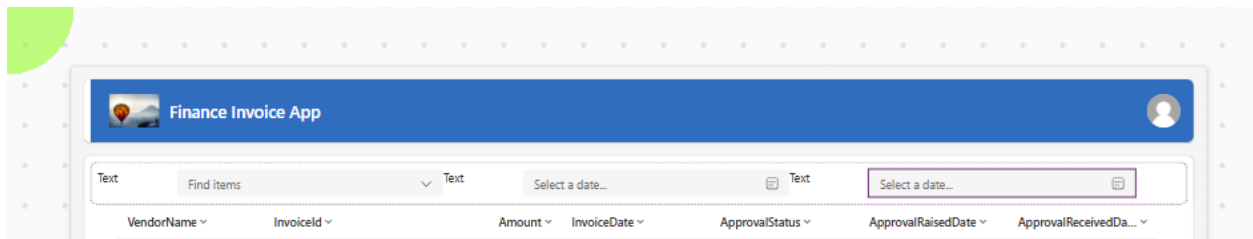


Switch Off Flexible height and set Height to 50.



2. Add 6 controls below to the container.

- Text Control
- ComboBox Control
- Text Control
- Date Picker Control
- Text Control
- Date Picker Control

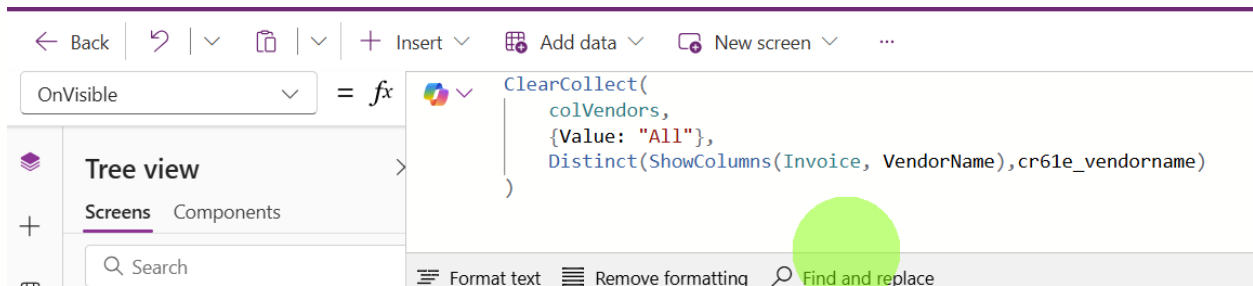


3. Change properties of those controls to below.

	Text	ControlName	Width
<b>Text Control</b>	Vendor Name	<i>No change</i>	<i>No change</i>
<b>ComboBox Control</b>	-	<i>Combo_VendorName</i>	<i>300</i>
<b>Text Control</b>	Invoice Date (From)	<i>No change</i>	<i>140</i>
<b>Date Picker Control</b>	-	<i>Date_From</i>	<i>300</i>
<b>Text Control</b>	Invoice Date (To)	<i>No change</i>	<i>140</i>
<b>Date Picker Control</b>	-	<i>Date_To</i>	<i>300</i>

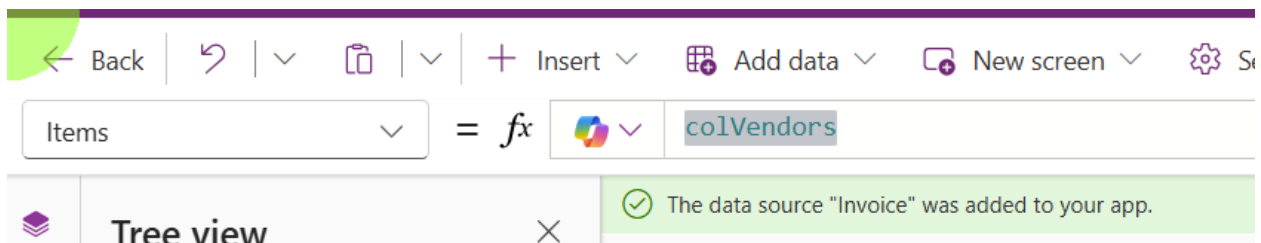
4. Insert this formula into OnVisible property of Screen\_Invoices.

```
ClearCollect(
    colVendors,
    {Value: "All"},
    Distinct(ShowColumns(Invoice, VendorName), cr61e_vendorname)
)
```



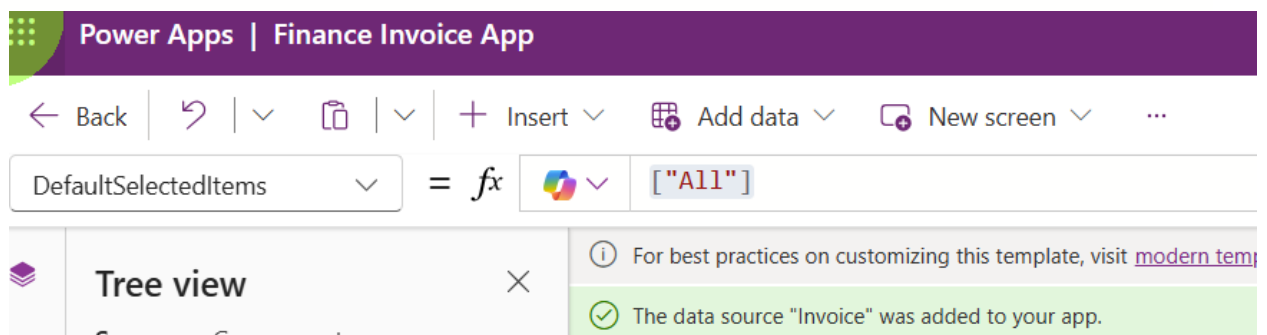
5. Change Items property of Combo\_VendorName to below.

colVendors



11. Change DefaultSelectedItems property of Combo\_VendorName to below.

["All"]

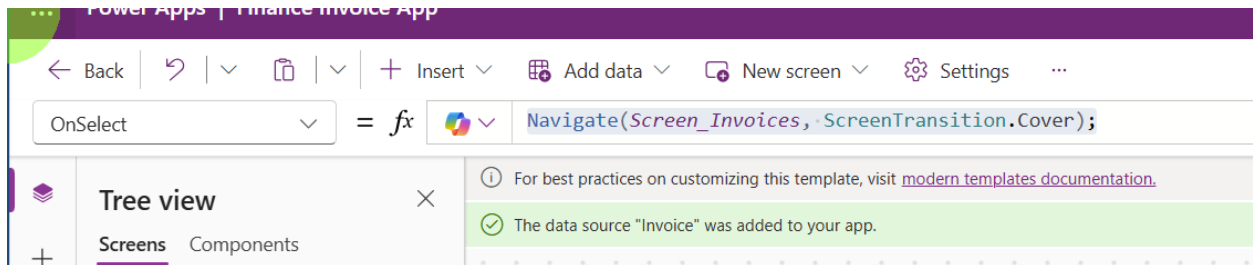


----- End of Task -----

## Task 2: Applying Filter

1. Click Screen\_Home. And set OnSelect property of Invoices Icon to below.

```
Navigate(Screen_Invoices, ScreenTransition.Cover);
```



2. Set Items property of Table to below.

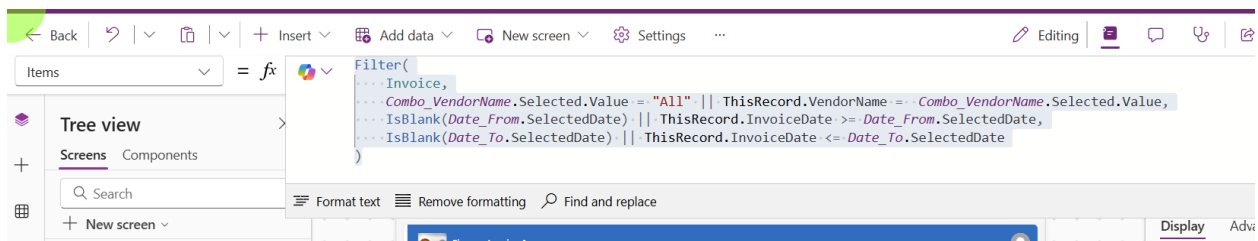
```
Filter(
    Invoice,

    Combo_VendorName.Selected.Value = "All" || ThisRecord.VendorName
= Combo_VendorName.Selected.Value,

    IsBlank(Date_From.SelectedDate) || ThisRecord.InvoiceDate >=
Date_From.SelectedDate,

    IsBlank(Date_To.SelectedDate) || ThisRecord.InvoiceDate <=
Date_To.SelectedDate

)
```



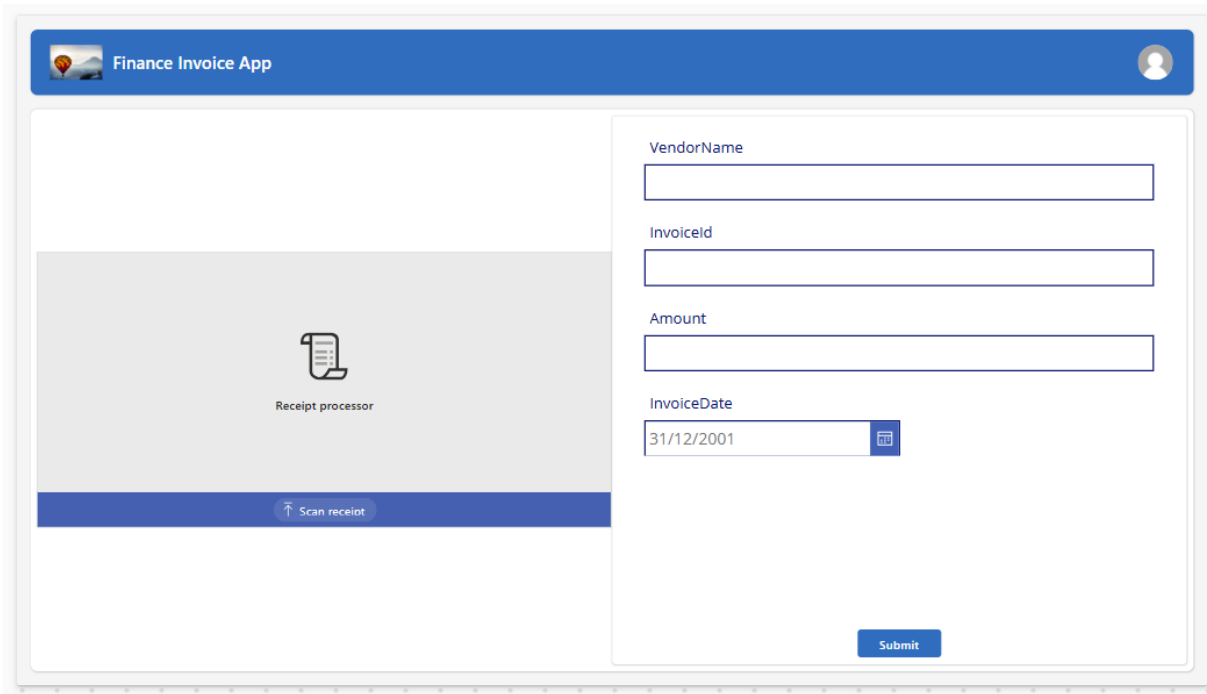
You should see the vendor list populated and be able to filter the table by vendor name, date range.

Sorting is automatically done by the Table control.

----- End of Task -----

# Exercise 5: Allow Users to Submit New Invoices

Users need to be able to submit new invoices to be processed using Receipt Processor. This exercise will get users to create a new screen, allow users upload invoices and automatically extract information.



The screenshot shows the 'Finance Invoice App' interface. On the left, there is a large grey area with a receipt icon and the text 'Receipt processor'. Below this is a blue button labeled 'Scan receipt'. On the right, there is a form with the following fields: 'VendorName' (text input), 'InvoiceId' (text input), 'Amount' (text input), and 'InvoiceDate' (date picker showing '31/12/2001'). A blue 'Submit' button is located at the bottom right of the form.

## Objectives

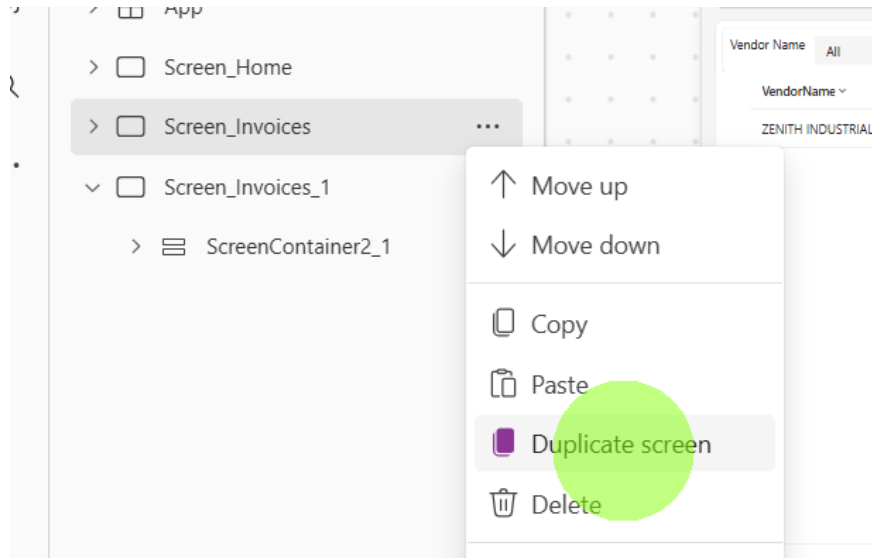
- User Receipt Processor to extract values from PDF
- Add in form that allows users to create a new request

## Estimated time to complete this lab

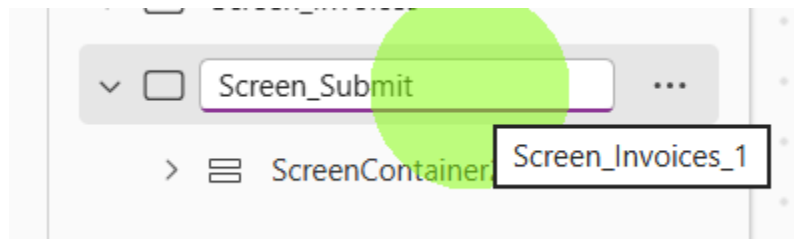
30 mins

## Task 1: Duplicate the existing Screen

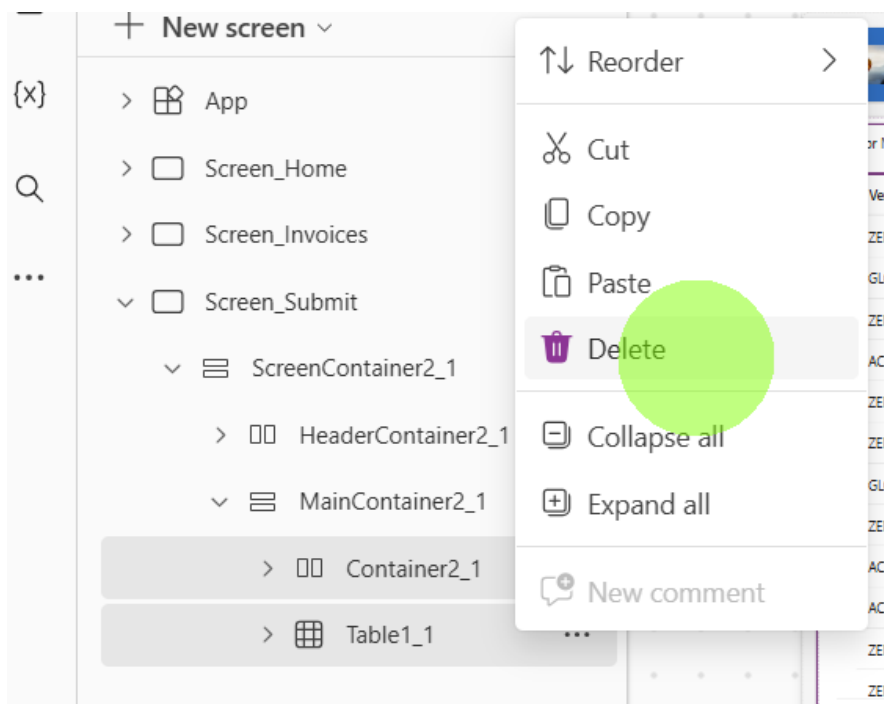
1. Duplicate Screen\_Invoices by following selecting the screen, right clicking it, and then selecting the 'Duplicate Screen' option.



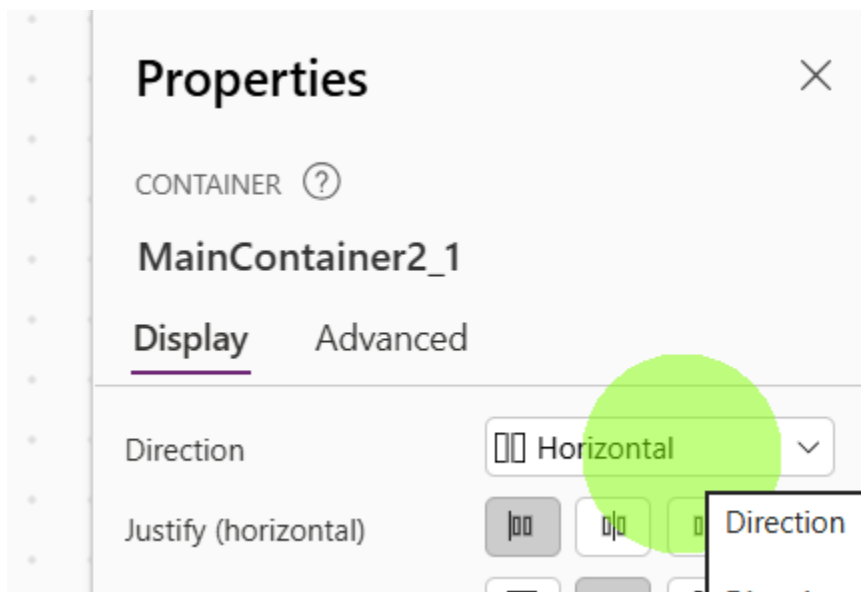
Rename the new screen to Screen\_Submit.



2. Expand the Main Container and delete the controls inside.



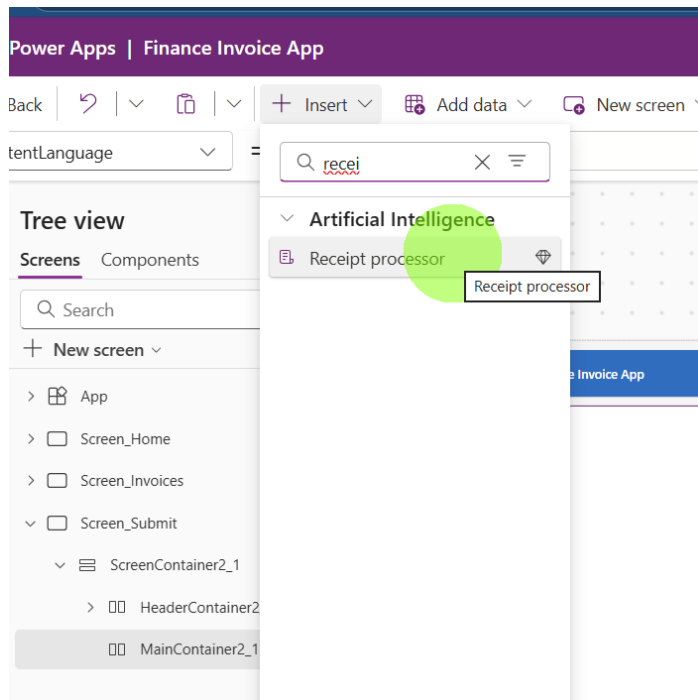
3. Change Main Container's Direction to Horizontal.



----- End of Task -----

## Task 2: Adding Controls to the Screen

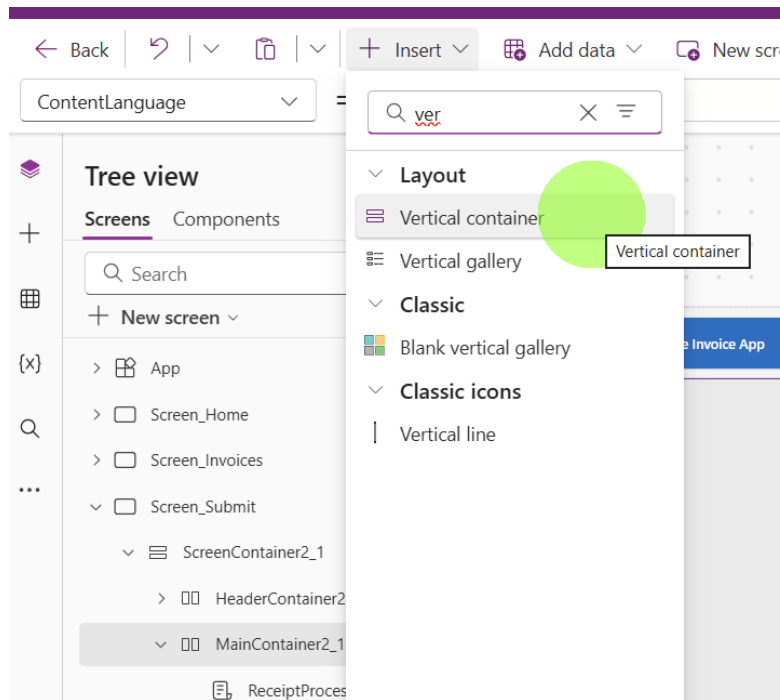
1. Select Main Container and add Receipt Processor control.



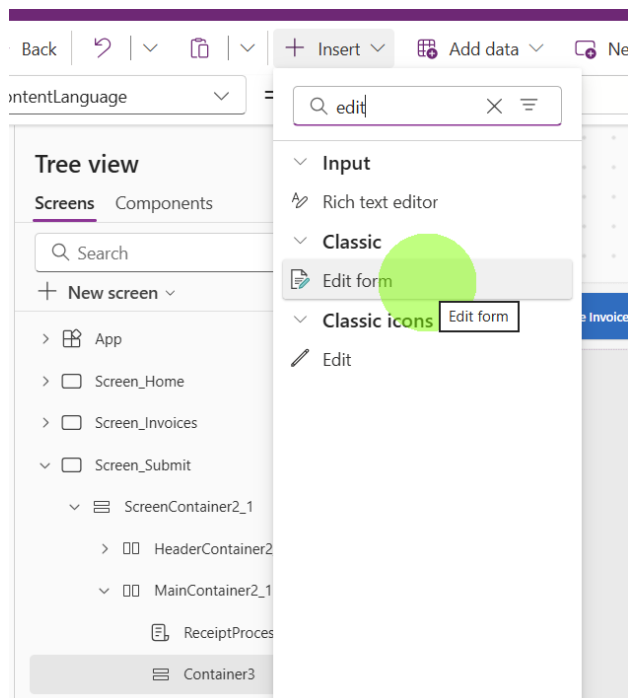
Change below properties of the Receipt Processor.

Height: **Parent**.Height  
FillPortions: 1

2. Add a new Vertical Container into the Main Container.

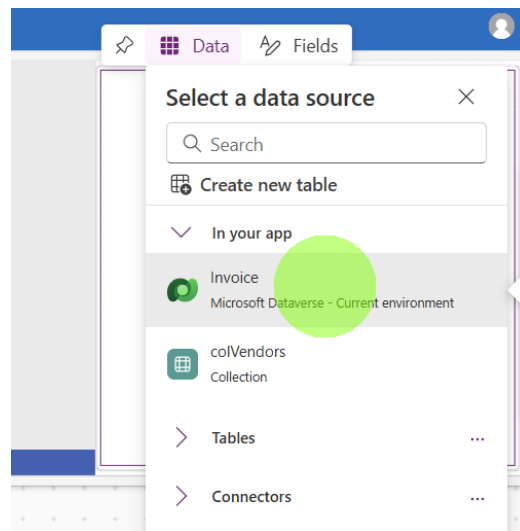


3. Add a new Form into the Vertical Container.

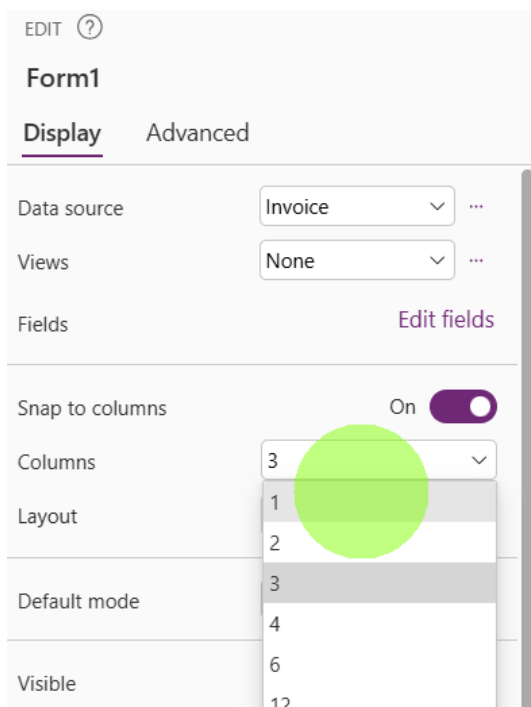




4. Point the Data Source for the Form to Invoice.



5. Change the number of Columns to 1

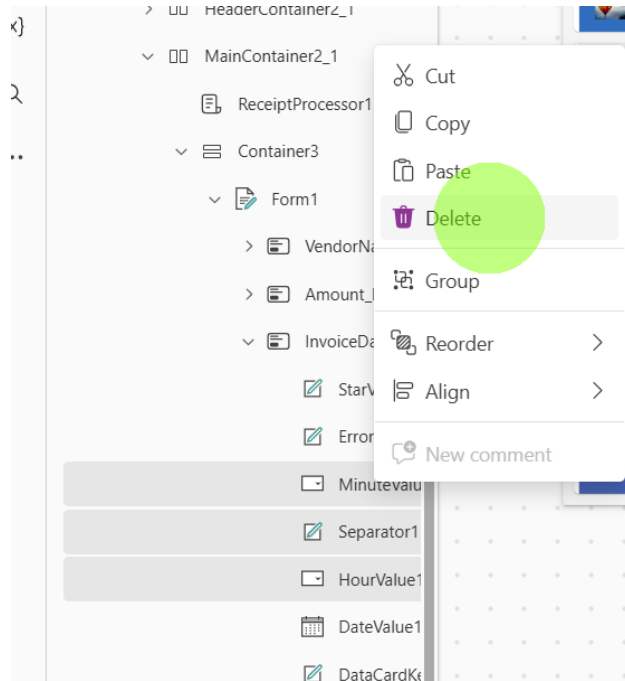


6. Select Fields to edit the fields displayed and only keep these 4 fields below.

The screenshot shows the Microsoft Power Platform form editor interface. On the left, a preview of the form is visible with fields for VendorName, InvoiceId, Amount, and InvoiceDate. The InvoiceDate field contains the value '31/12/2001'. In the center, the 'Fields' pane lists four fields: VendorName, InvoiceId, Amount, and InvoiceDate, each with a dropdown arrow and a small icon. Above the list, there is a '+ Add field' button and a note 'Suggested by AI. Please review. [See terms](#)'. On the right, the 'Properties' pane is open, showing settings for 'Form1'. The 'Display' tab is selected. The 'Data source' is set to 'Invoice'. The 'Views' dropdown is set to 'None', with a green circle highlighting it and a tooltip that says '4 selected'. The 'Fields' section also has a green circle highlighting it with a tooltip that says '4 selected'. Other settings include 'Snap to columns' (On), 'Columns' (1), 'Layout' (Vertical), 'Default mode' (Edit), and 'Visible' (On).

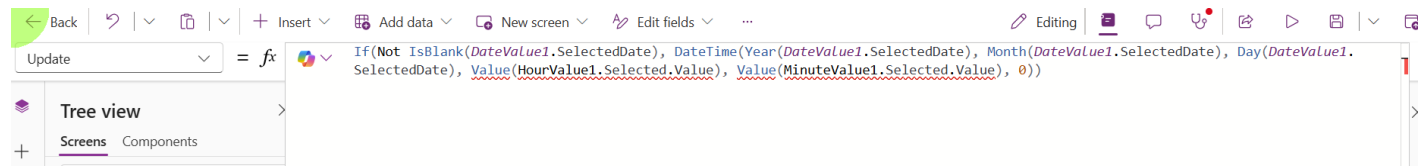
7. Unlock Invoice Date Field and delete Hours, Minutes and Separator controls.

The screenshot shows a close-up of the 'InvoiceDate' field in the form editor. The field is currently locked, indicated by a padlock icon. A green circle highlights the 'Unlock' button, which is located next to the field. Below the 'Unlock' button, there are options for 'Paste' and 'New comment'. The field itself contains the date '31/12/2001'.



There will be errors generated. To fix the error, replace the Error Values with Zeros.

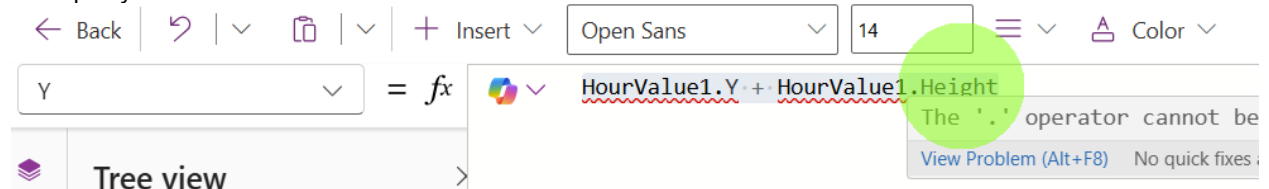
## Update Property



Value(HourValue1.Selected.Value) → 0

Value(MinuteValue1.Selected.Value) → 0

## Y Property



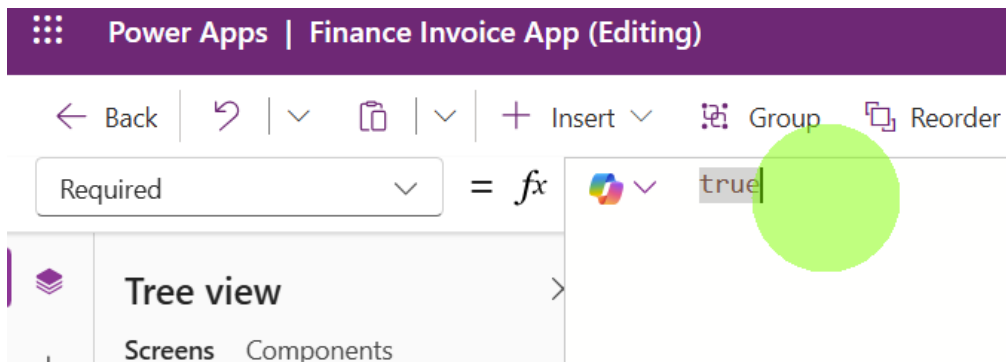
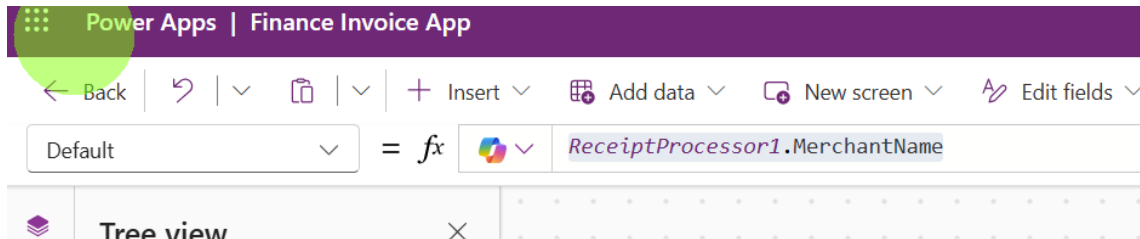
HourValue1.Y + HourValue1.Height → 0

----- End of Task -----

## Task 3: Extracting Values from Receipt Processor

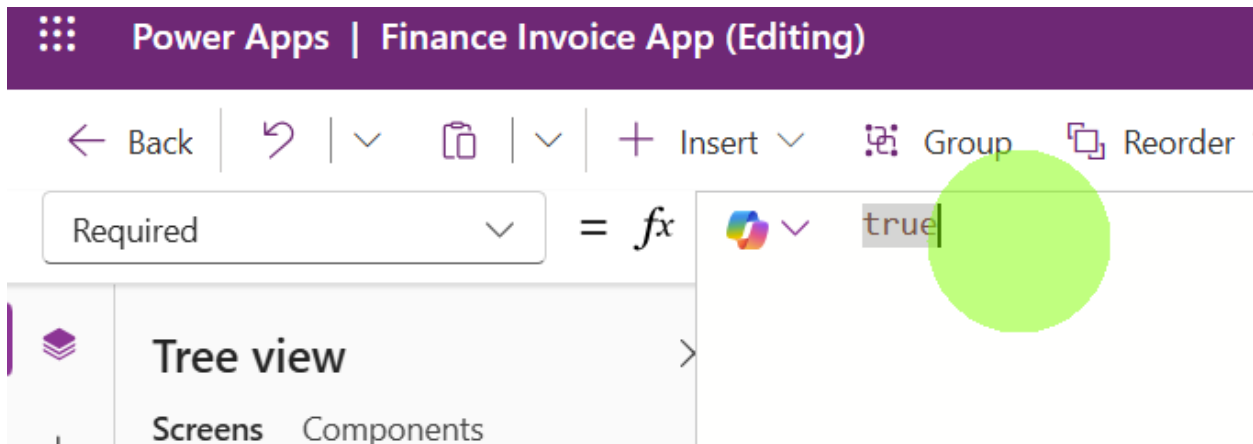
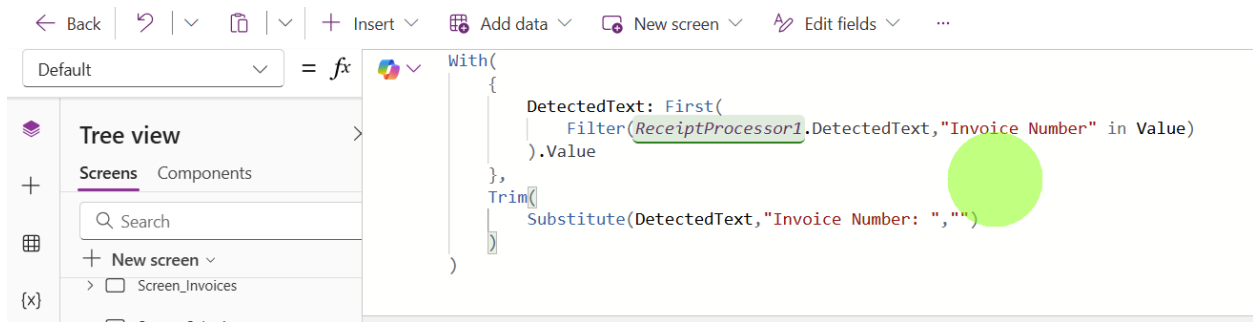
1. Unlock VendorName Card and set Default property to below and Required to true.

*ReceiptProcessor1*.MerchantName



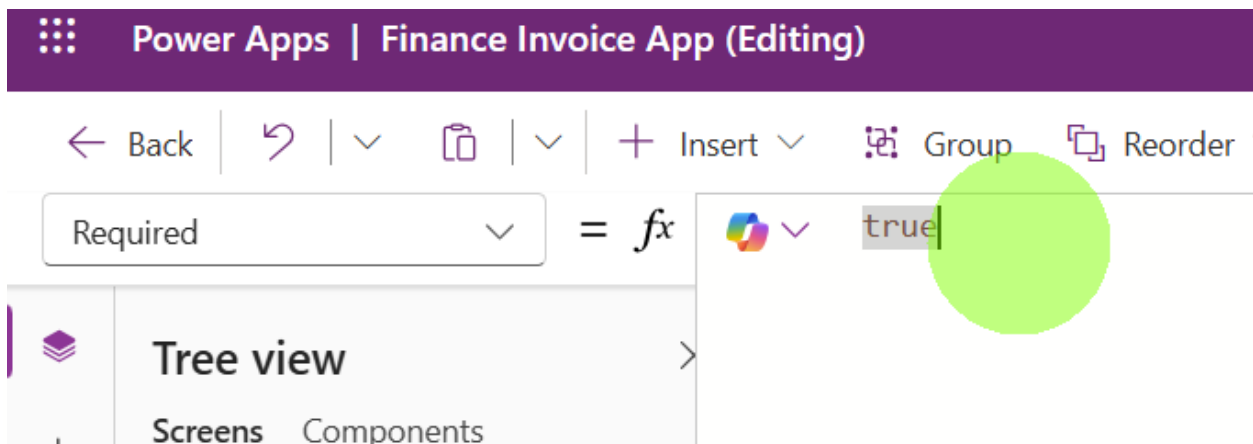
2. Unlock InvoiceId Card and set Default property to below and Required to true.

```
With(  
  {  
    DetectedText: First(  
      Filter(ReceiptProcessor1.DetectedText,"Invoice Number" in  
Value)  
    ).Value  
  },  
  Trim(  
    Substitute(DetectedText,"Invoice Number: ","")  
  )  
)
```



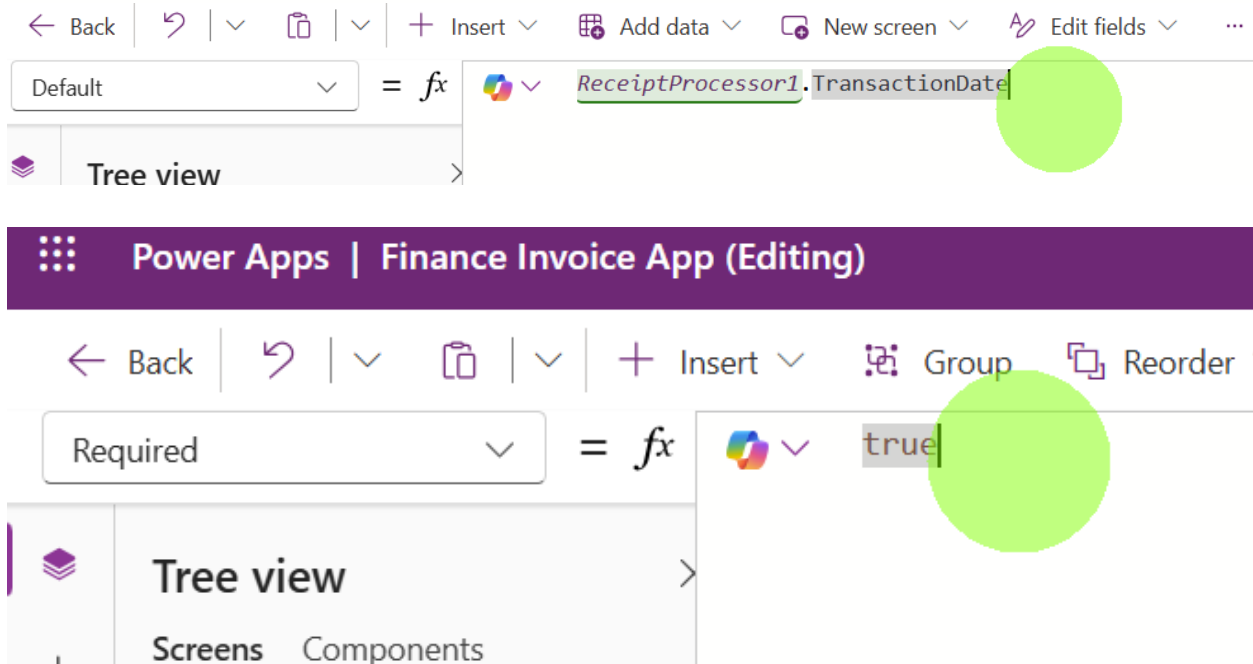
3. Unlock Amount Card and set Default property to below and Required to true.

*ReceiptProcessor1.Total*



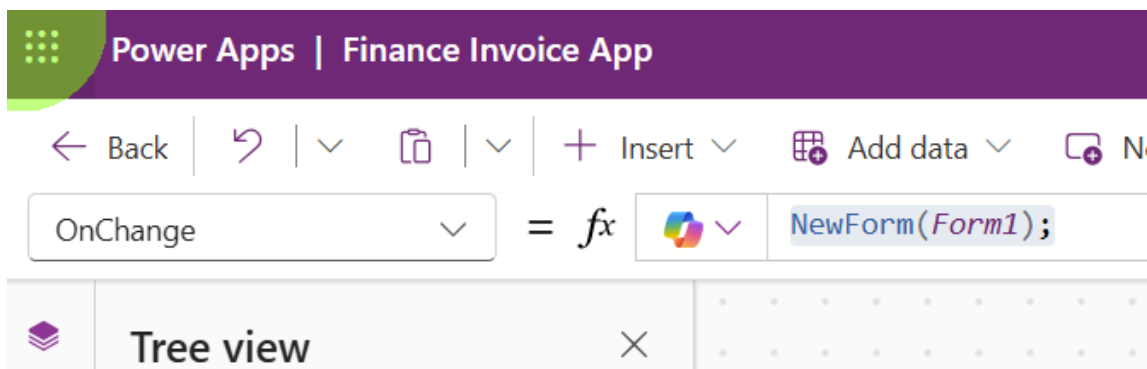
4. Unlock InvoiceDate Card and set Default property to below and Required to true.

*ReceiptProcessor1.TransactionDate*





5. Change OnChange properties of the Receipt Processor to below.

`NewForm(Form1);`



You can test with the sample invoices provided to you and the form should auto-populate the values based on invoice uploaded.

**Finance Invoice App**

GLOBAL TECH SOLUTIONS PTE LTD

100 Robinson Road, #10-01, Robinson 110211

050 1234 5678 | finance@globaltech.sg

**INVOICE**

Invoice Number: INV-3001

Invoice Date: 23/01/2025

Bill To: "Global Tech Pte Ltd" Group (Singapore)

Description	Qty	Unit Price (\$SGD)	Line Total (\$SGD)
Supply & Installation of Servers	1	1500.00	1500.00
Network Storage Setup	1	2000.00	2000.00
Cloud Storage Subscription (1yr)	1	400.00	400.00
Software Licenses (Microsoft)	1	1000.00	1000.00
Professional Services - Project Mgmt	1	1500.00	1500.00
Software License - Enterprise	1	1500.00	1500.00
Subtotal:			7900.00
GST (9%):			706.50
Total (\$SGD):			8606.50

Payment due within 30 days of invoice date. Please email for your business.

VendorName

GLOBAL TECH SOLUTIONS PTE LTD

InvoiceId

INV-3001

Amount

15685.1

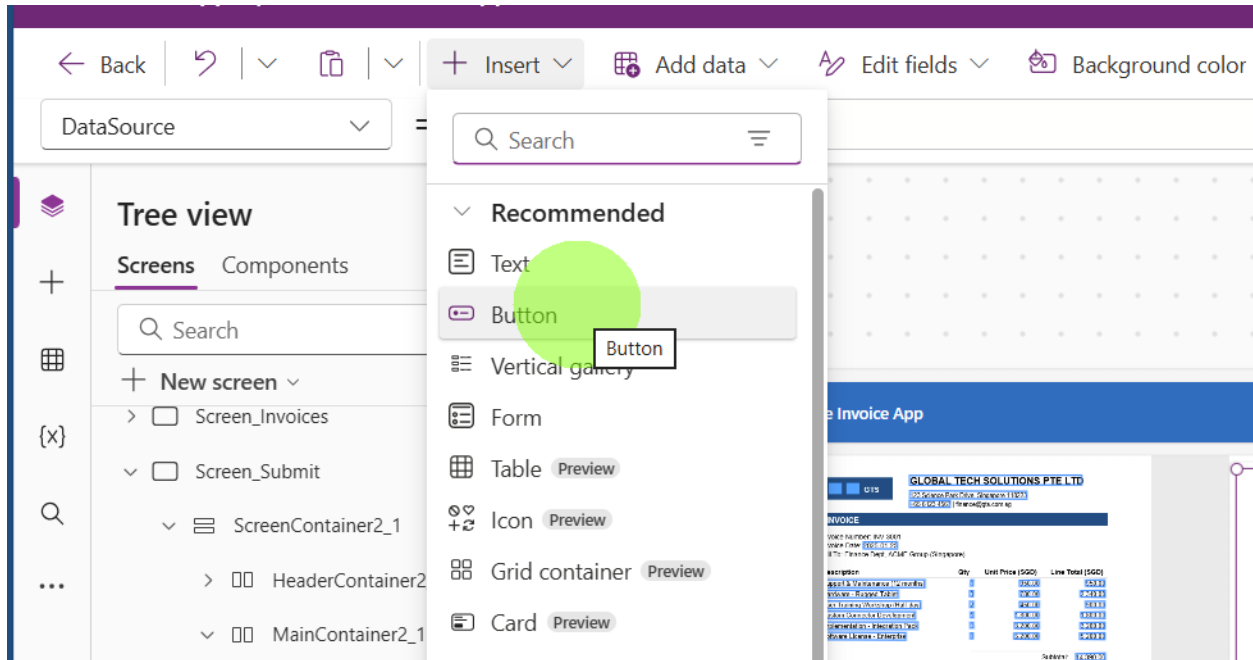
InvoiceDate

23/01/2025

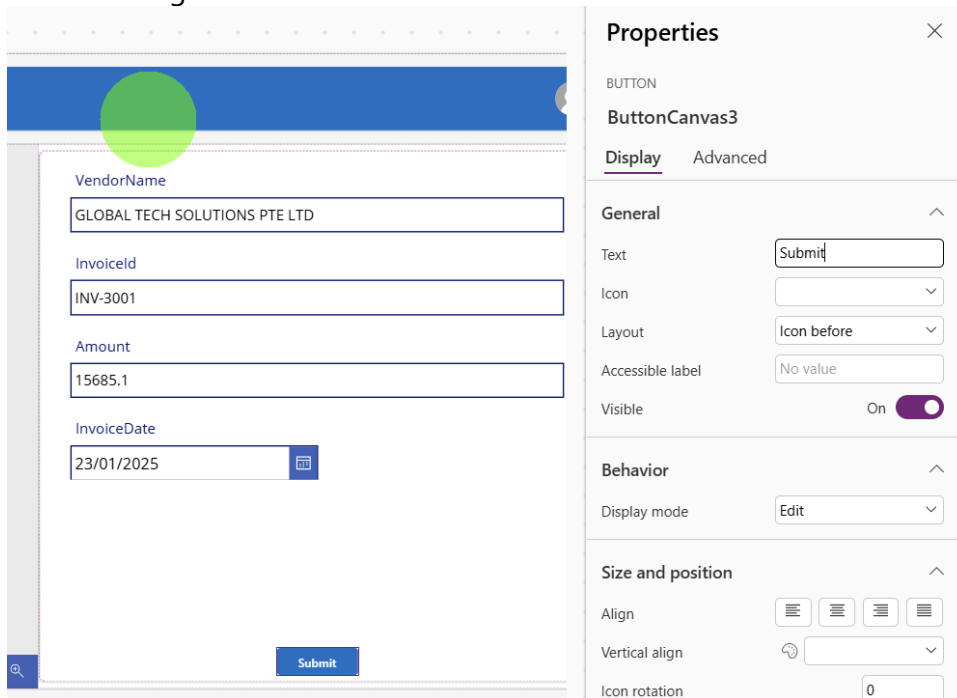
----- End of Task -----

## Task 4: Submitting Extracted Values

1. Add a button below the form.



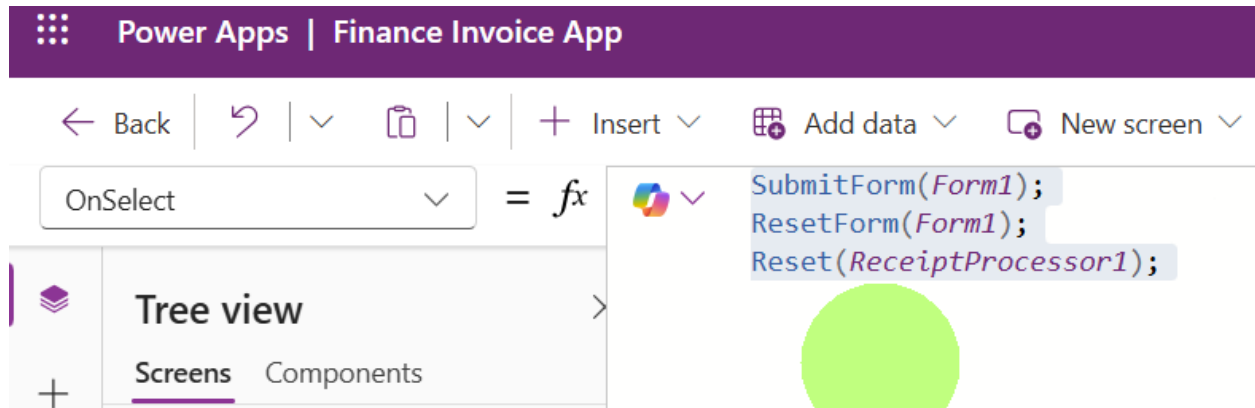
2. Change Text to Submit





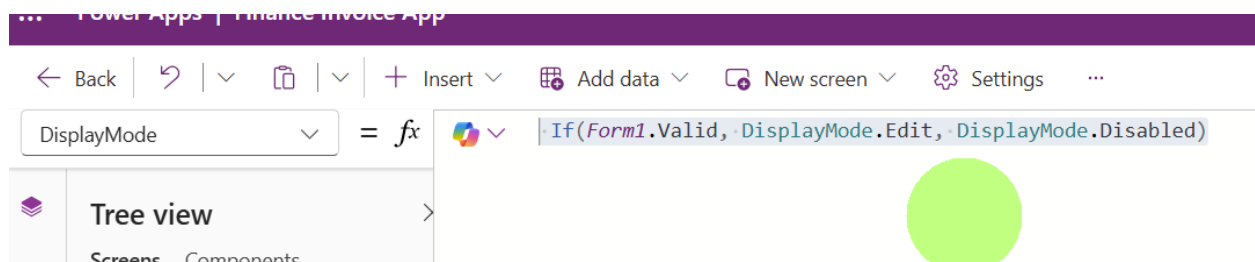
3. Set OnSelect property of Submit button to this.

```
SubmitForm(Form1);  
ResetForm(Form1);  
Reset(ReceiptProcessor1);
```



4. Set DisplayMode property of Submit button to this.

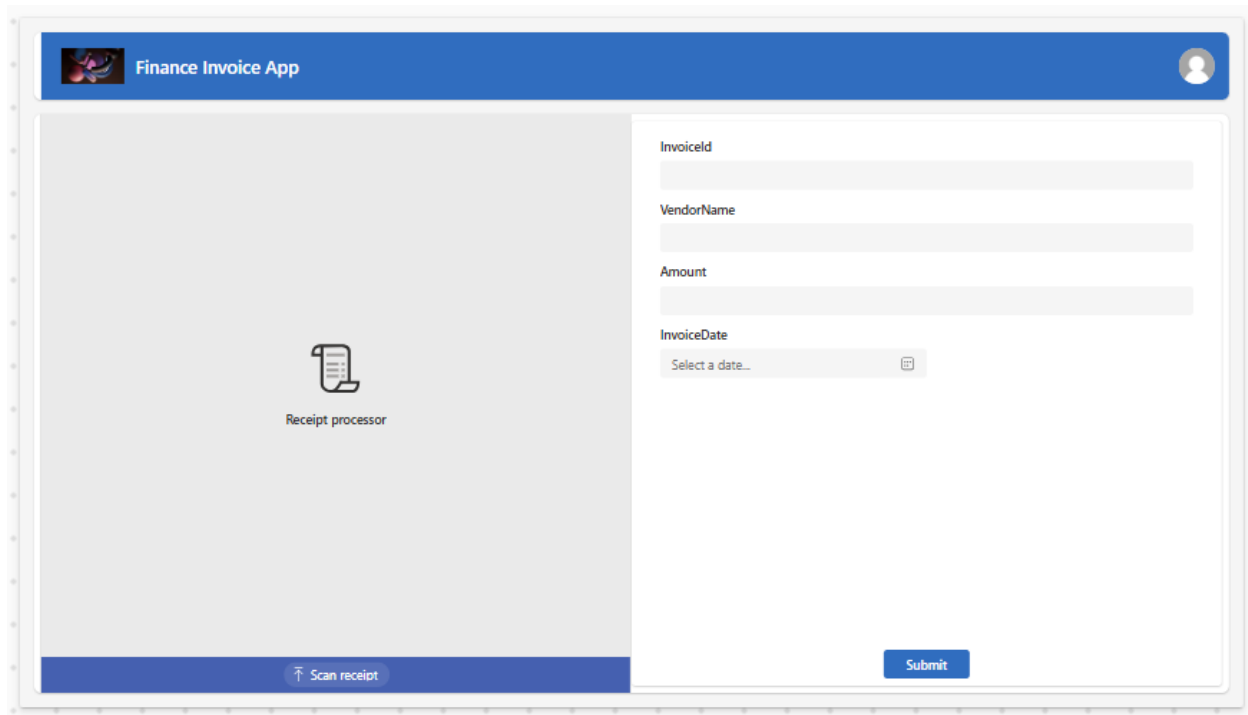
```
If(Form1.Valid, DisplayMode.Edit, DisplayMode.Disabled)
```



----- End of Task -----

# Exercise 6: Creating Message and Loading Popups using Components

Utilising components to create reusable message and loading popups.



## Objectives

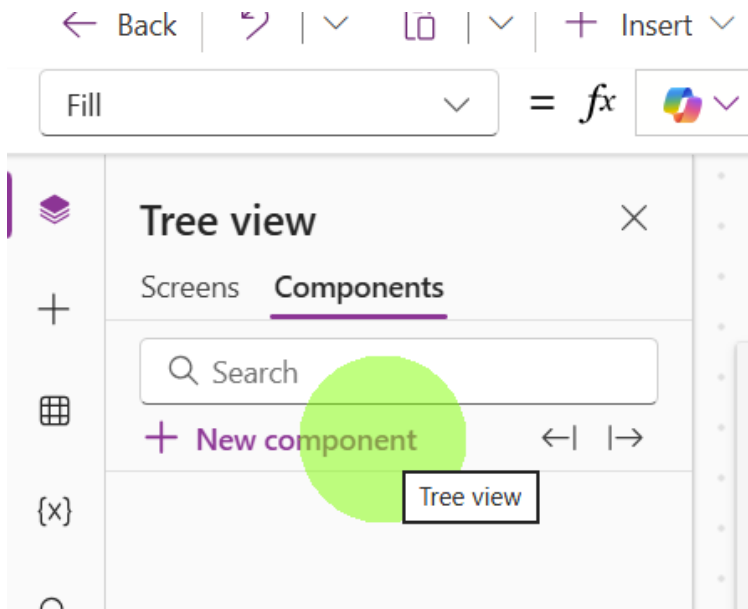
- Understand the use of components and custom properties

## Estimated time to complete this lab

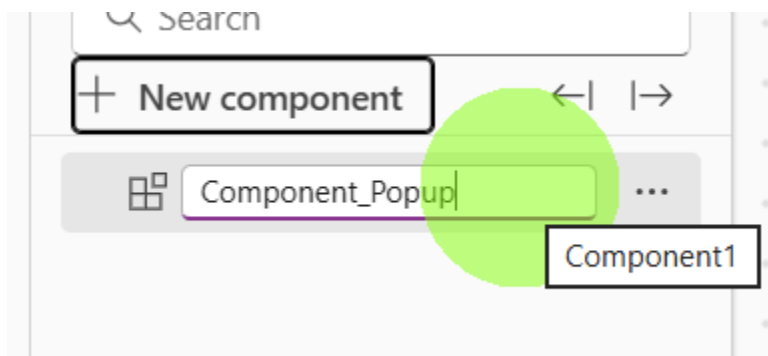
20 mins

## Task 1: Adding a New Message Popup Component with Custom Properties

1. Switch to Components tab and click +New Component.



Rename it to Component\_Popup.



Set Width and Height of the component as below.

Width: 800

Height: 600

**Component\_Popup**

Display   Advanced

Description

Access app scope Off ☐

Visible On ☒

Size    
Width   Height

Fill

Set Fill Property to the component as below.

`RGBA(255, 255, 255, 0.9)`

The screenshot shows the Power Platform interface. At the top, there is a purple header bar. Below it is a navigation bar with icons for Back, Undo, Redo, and Insert, followed by text labels "Add data" and "New screen". The main area displays the "Fill" property set to `RGBA(255, 255, 255, 0.9)`. Below this is a "Tree view" panel with a close button.

2. Add a new Custom Property (MessageType) to the Component as below.

New custom property

Display name \*

MessageType

Name \*

MessageType

Description \*

MessageType

Property type

Data

Property definition

Input

Data type

Text

☐ Raise OnReset when value changes

Create

Cancel

Properties

COMPONENT

Component\_Popup

Display Advanced

Description

No value

Access app scope

Off

Visible

On

Size

800

600

Width

Height

Fill

Custom properties

+ New custom property

3. Add one more Custom Property (Message) as below.

New custom property

Display name \*

Message

Name \*

Message

Description \*

Message

Property type

Data

Property definition

Input

Data type

Text

☐ Raise OnReset when value changes

Create

Cancel

Properties

COMPONENT

Component\_Popup

Display

Advanced

Description

No value

Access app scope

Off

Visible

On

Size

800

600

Width

Height

Fill

Custom properties

+ New custom property

MessageType

Text

4. Add one more Custom Property (ClickEvent) as below.

New custom property

Display name \*

ClickEvent

Name \*

ClickEvent

Description \*

ClickEvent

Property type

Event

Return data type

None

Parameters

+ New parameter

Create

Cancel

Properties

COMPONENT

Component\_Popup

Display Advanced

Description

No value

Access app scope

Off

Visible

On

Size

800600

WidthHeight

Fill

Custom properties

+ New custom property

MessageType

Text

...

Message

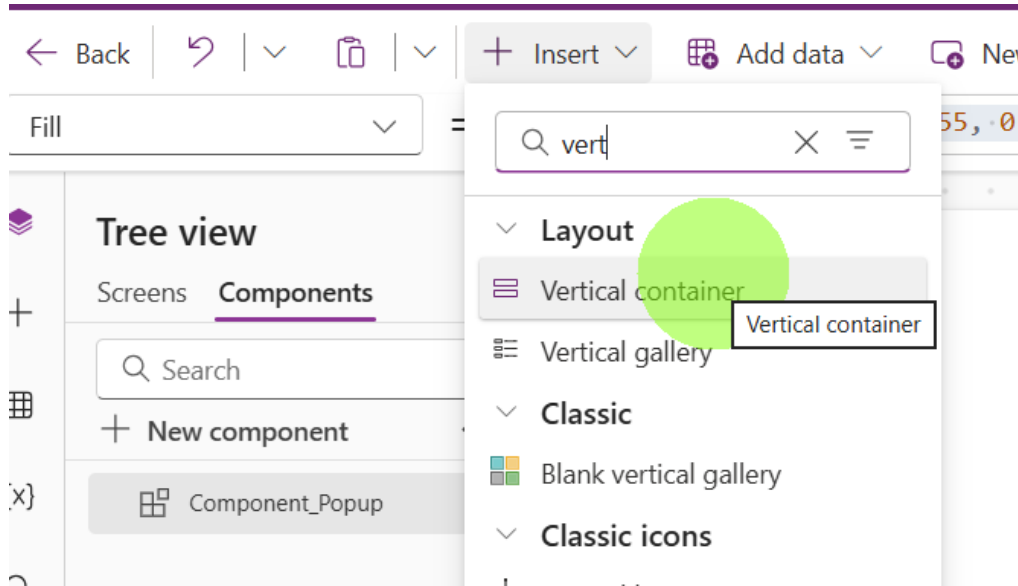
Text

...

End of Task

## Task 2: Adding Controls into the Component

1. Add a new Vertical Container inside the component



Change below properties of the Vertical Container.

Width: 500

Height: 400

LayoutGap: 30

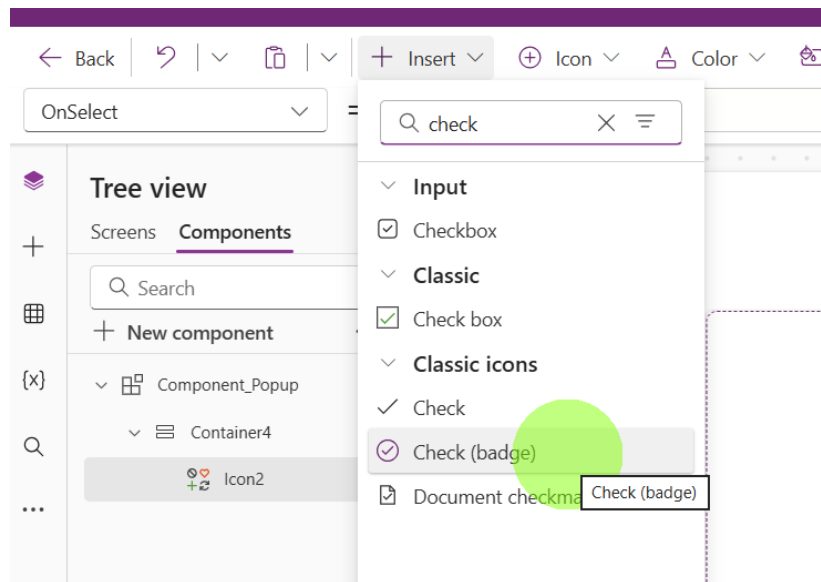
LayoutJustifyContent: `LayoutJustifyContent.Center`

X: `Parent.Width/2 - Self.Width/2`

Y: `Parent.Height/2 - Self.Height/2`



2. Insert a Check (badge) icon into the container.



Change below properties of the Icon.

Width: 100

Height: 100

Color: `If(Component_Popup.MessageType = "Success", Color.DarkGreen, Color.DarkRed)`

Icon: `If(Component_Popup.MessageType = "Success", Icon.CheckBadge, Icon.Error)`

Property

Top Bottom

0 0

Left Right

Width 100

Align in container Set by container

Flexible height Off

Height 100

Back

Icon

Color

Background color

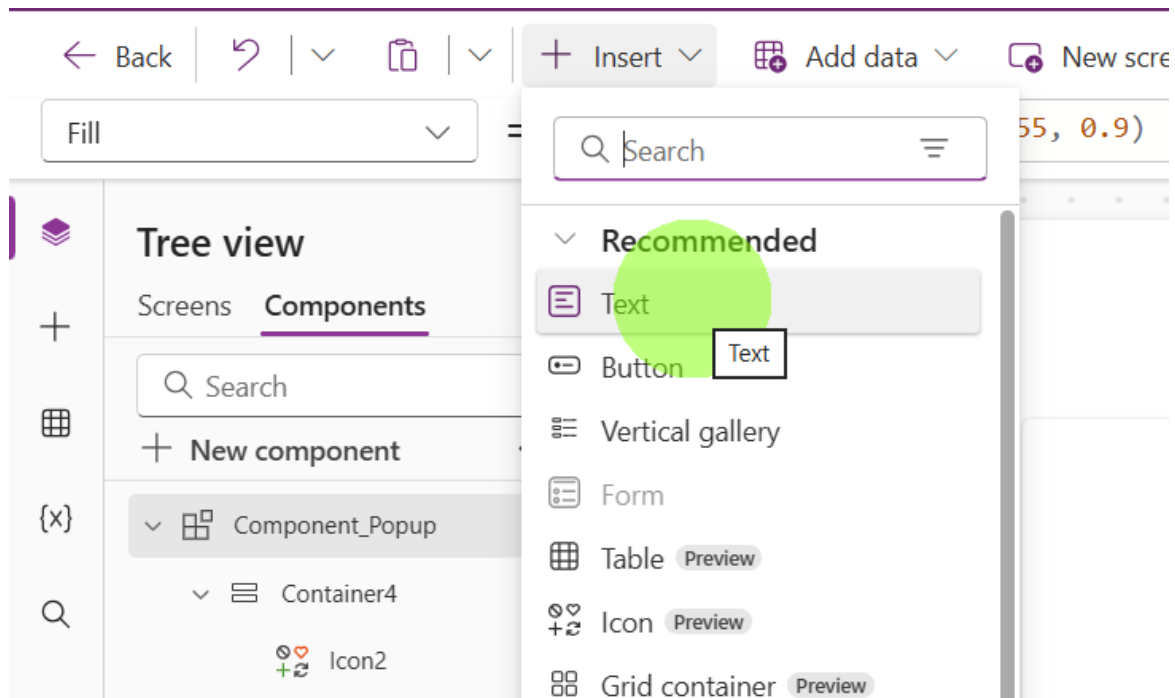
Icon

= fx

If(Component\_Popup.MessageType = "Success", Icon.CheckBadge, Icon.Error)

Tree view

3. Insert a Text control into the container.



Change below properties of the Text.

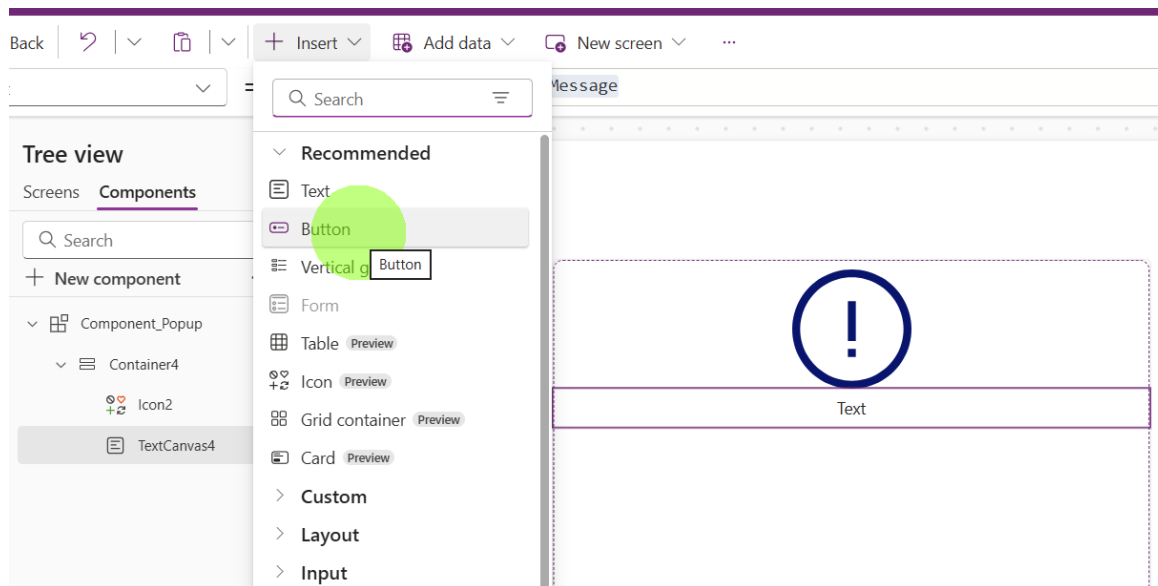
Width: `Parent.Width`

Align: `'TextCanvas.Align'.Center`

VerticalAlign: `VerticalAlign.Middle`

Text: `Component_Popup.Message`

### 3. Insert a Button control into the container.

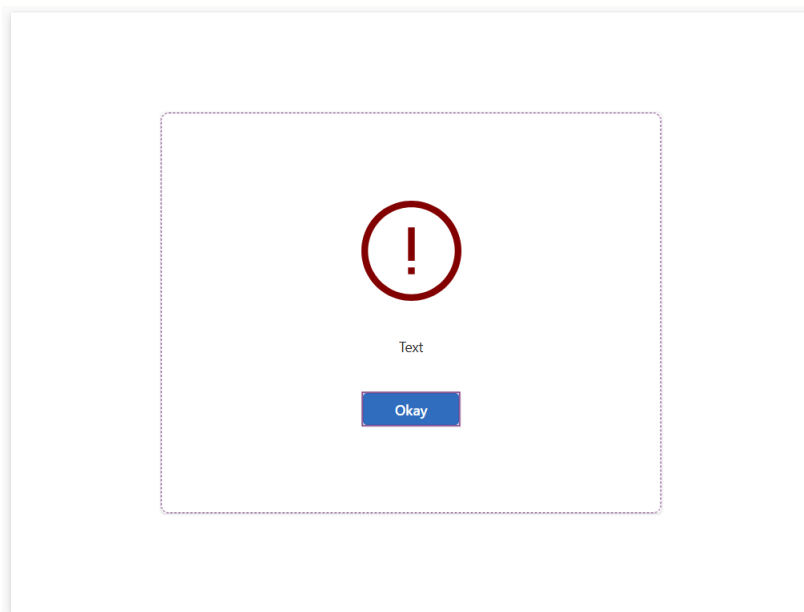


Change below properties of the Button.

Text: "Okay"

OnSelect: `Component_Popup.ClickEvent()`

Your Component should look something like below.



----- End of Task -----

## Task 3: Adding Message Popup Component to Submission Screen

1. Select Form control and change OnSuccess proper to below.

`UpdateContext({MessageType: "Success", Message: "Invoice data submitted successfully", ShowPopup: true});`

The screenshot shows the Microsoft Power Apps Studio interface. At the top, the formula bar displays the following code for the `OnSuccess` property:

```
UpdateContext({MessageType: "Success", Message: "Invoice data submitted successfully", ShowPopup: true});
```

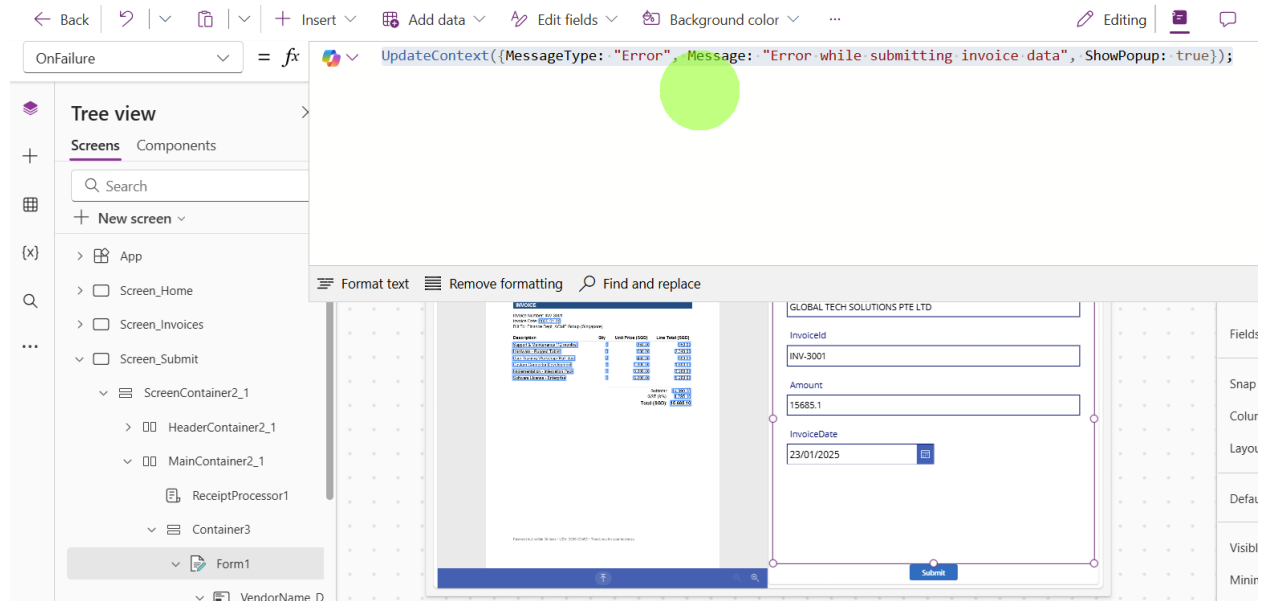
On the left, the **Tree view** pane shows the hierarchy of the app:

- App
  - Screen\_Home
  - Screen\_Invoices
  - Screen\_Submit
    - ScreenContainer2\_1
      - HeaderContainer2\_1
      - MainContainer2\_1
        - ReceiptProcessor1
        - Container3
          - Form1 (Selected)
          - VendorName\_D

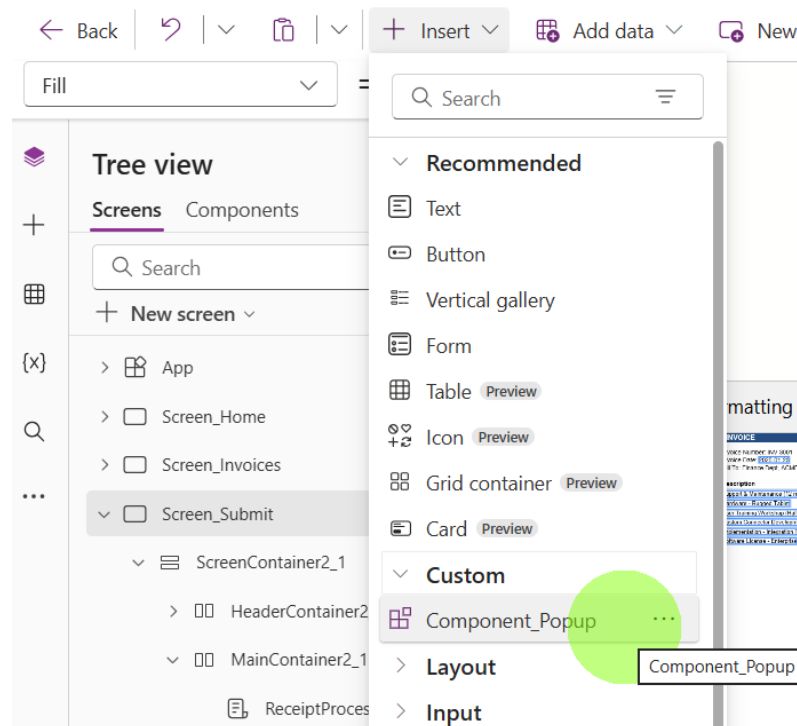
The main canvas shows a form titled "GLOBAL TECH SOLUTIONS PTE LTD" with fields for `InvoiceId` (value: INV-3001), `Amount` (value: 15685.1), and `InvoiceDate` (value: 23/01/2025). A `Submit` button is at the bottom right. A green circle is overlaid on the canvas, indicating the selected form control.

2. Select Form control and change OnFailure property to below.

```
UpdateContext({MessageType: "Error", Message: "Error while submitting invoice data", ShowPopup: true});
```



3. Insert Component\_Popup into Screen\_Submit.



Change below properties of Component.

Width: `Parent.Width`

Height: `Parent.Height`

Visible: `ShowPopup`

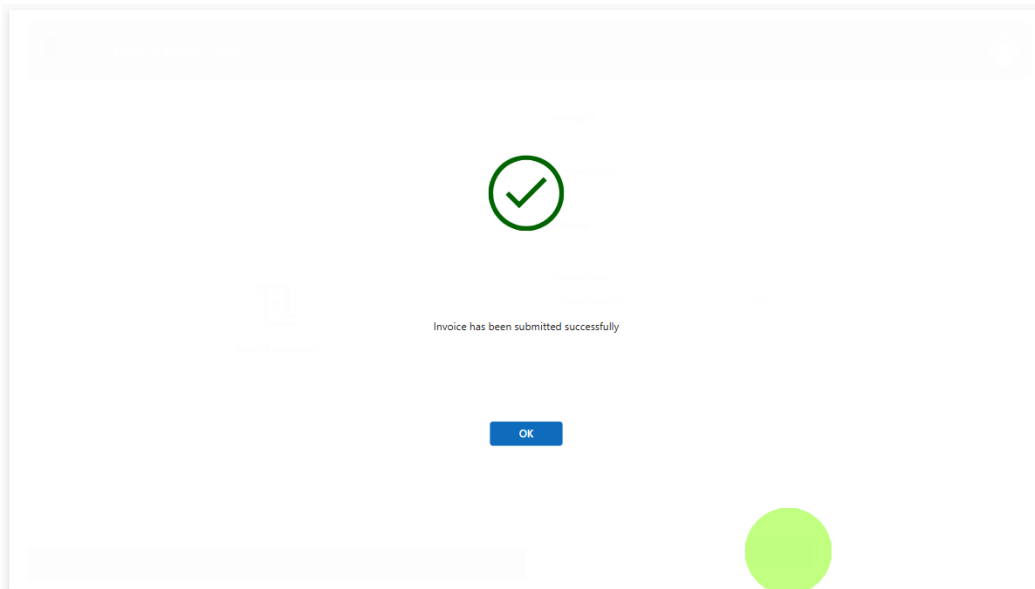
ClickEvent: `UpdateContext({ShowPopup:false});`

MessageType: `MessageType`

Message: `Message`

When you try to submit the invoice data, it will now show the Success Popup.

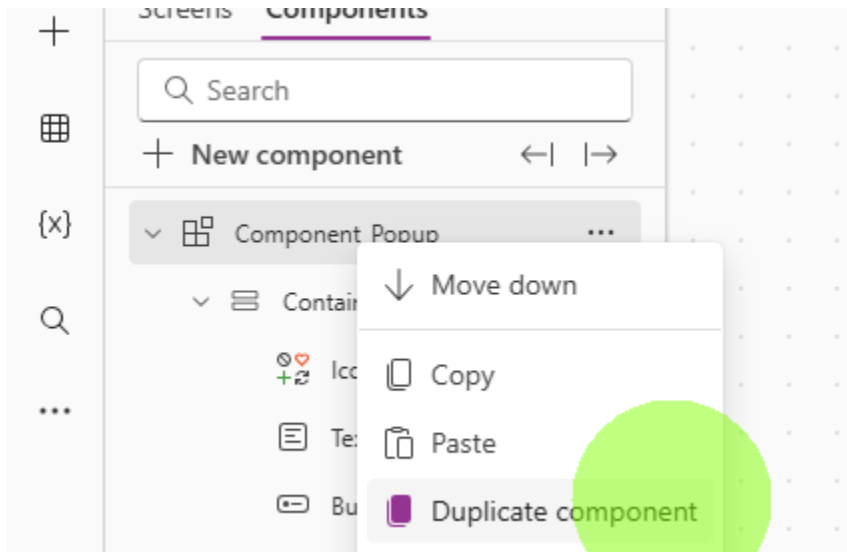
*(To test Error Popup, try putting text value to Invoice Amount field to generate the error.)*



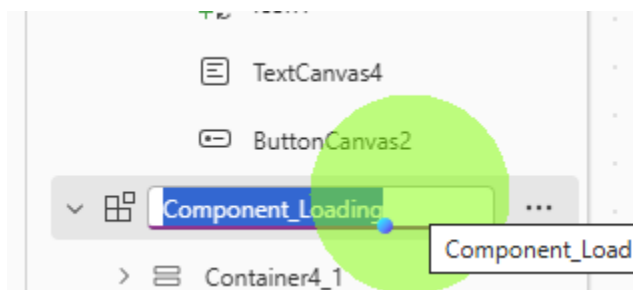
----- **End of Task** -----

## Task 4: Adding a New Loading Component

1. Duplicate Component Popup.

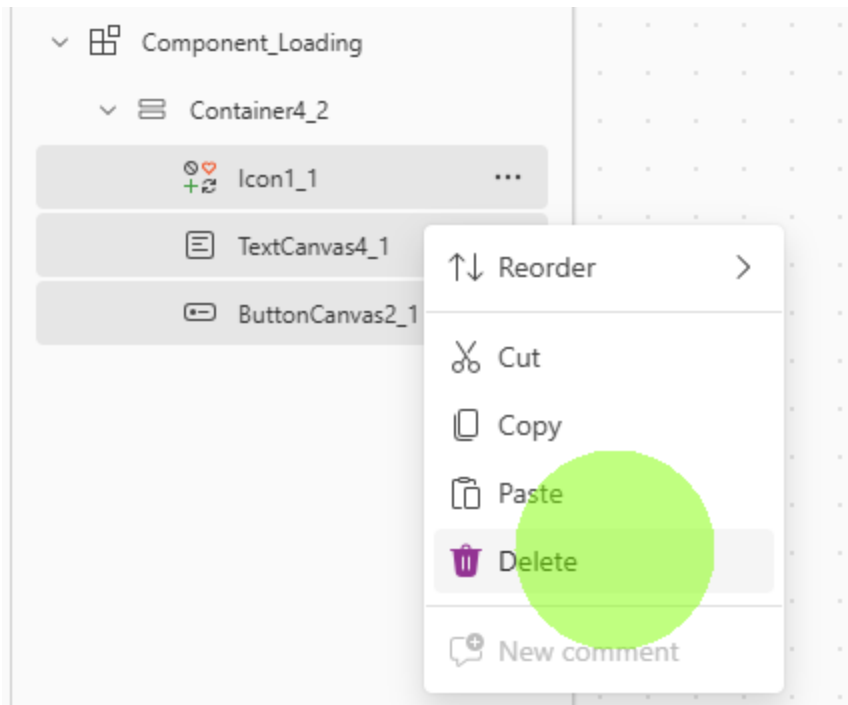


2. Rename it as Component\_Loading.

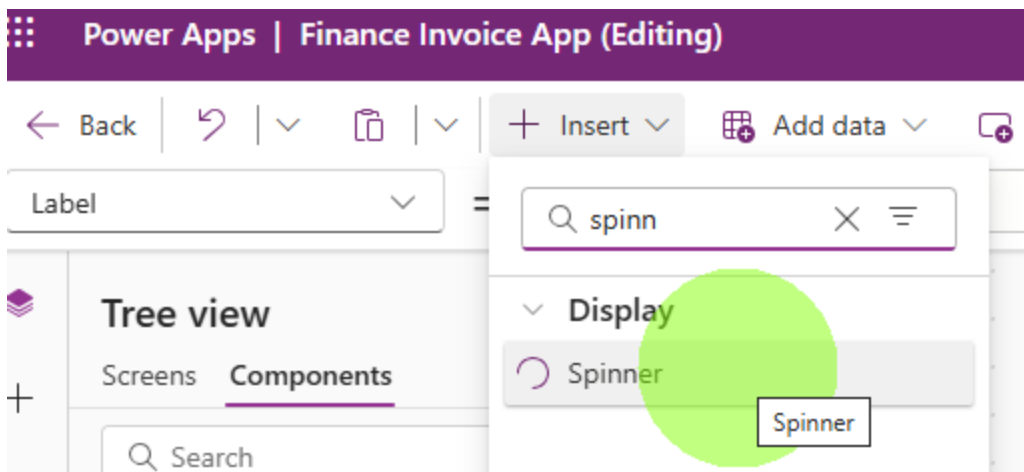


3. Delete all the child controls inside Component\_Loading's middle Container.





4. Add a Spinner control into the middle Container.



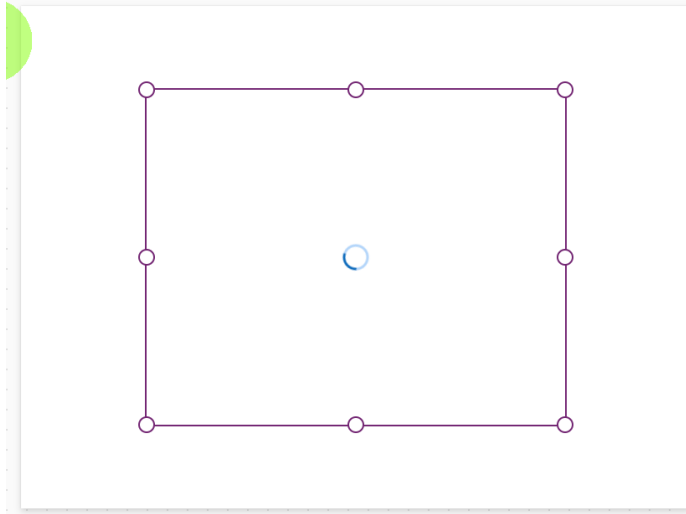
Change below properties of the Spinner control.

Width: **Parent.Width**

Label: **Component\_Loading.Message**

Change LayoutJustifyContent property of middle Container to below:

LayoutJustifyContent: **LayoutJustifyContent.Center**



5. Select Component\_Loading control, open Properties and delete MessageType and ClickEvent properties.

Edit custom property

Display name \* ⓘ

Name \* ⓘ

Description \* ⓘ

Property type ⓘ

Property definition ⓘ

Data type ⓘ

☐ Raise OnReset when value changes ⓘ

[Delete custom property](#)

Edit custom property

Display name \* ⓘ

Name \* ⓘ

Description \* ⓘ

Property type ⓘ

Return data type ⓘ

**Parameters**  

+ New parameter

[Delete custom property](#)

----- End of Task -----

## Task 5: Adding Loading Component to Submission Screen

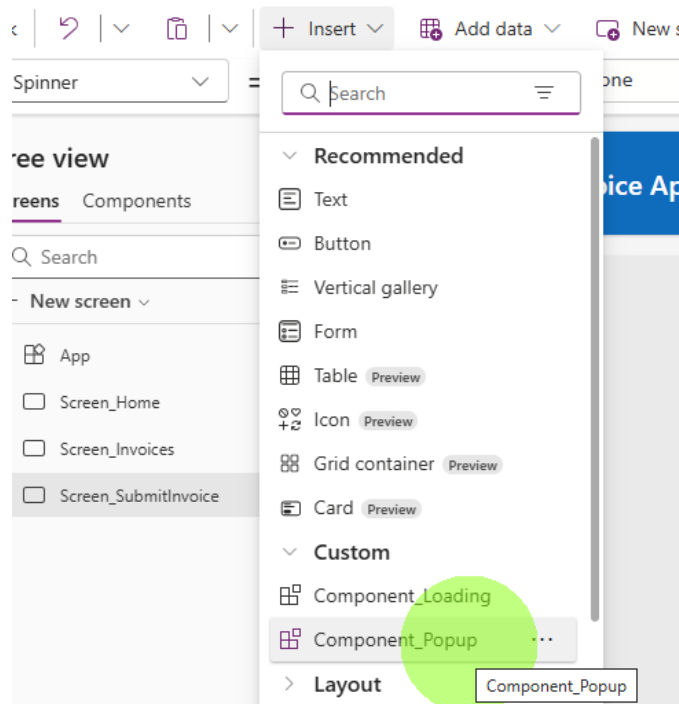
1. Update OnSelect property of Submit button to below.

```
UpdateContext({ ShowLoading: true, Message: "Submitting invoice values"});
```

```
SubmitForm(Form1);  
ResetForm(Form1);  
Reset(ReceiptProcessor1);
```

```
UpdateContext({ ShowLoading: false, Message: ""});
```

2. Insert Component\_ Loading into Screen\_Submit.

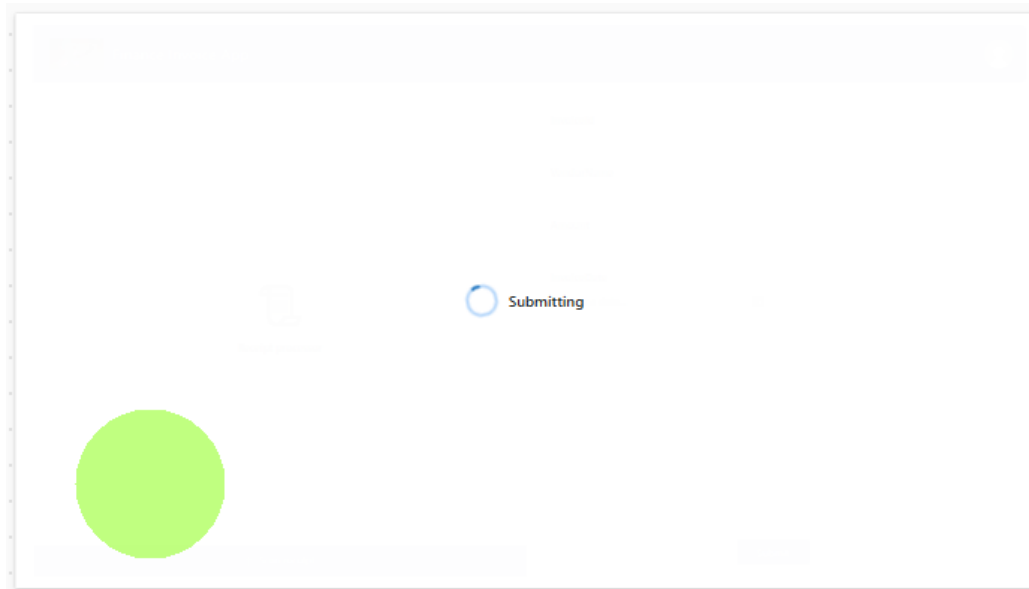


Change below properties of Component.

Width: **Parent.Width**  
Height: **Parent.Height**  
Visible: **ShowLoading**  
Message: **Message**

When you try to submit the invoice data, it will now show the Loading Popup before showing the Success Popup..

*(To test Error Popup, try putting text value to Invoice Amount field to generate the error.)*



----- **End of Task** -----