

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	43
Exercise 6: Creating Message Popups using Components	46
Objectives	46
Estimated time to complete this lab	46

Task 1: Adding a New Component with Custom Properties.....	47
Task 2: Adding Controls into the Component.....	52
Task 3: Adding Popup Component to Submission Screen	57

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

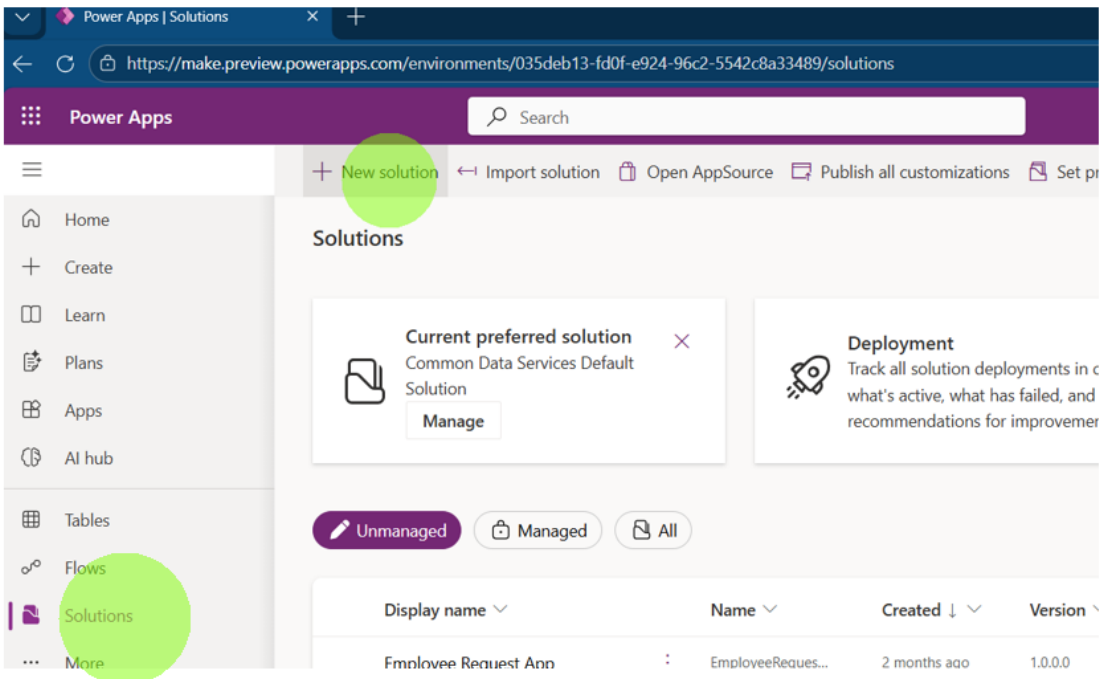
- InvoiceProcessingApplImportData_1_0_0_1.zip
- invoices_last3months_simple_approval.xlsx

Estimated time to complete this lab

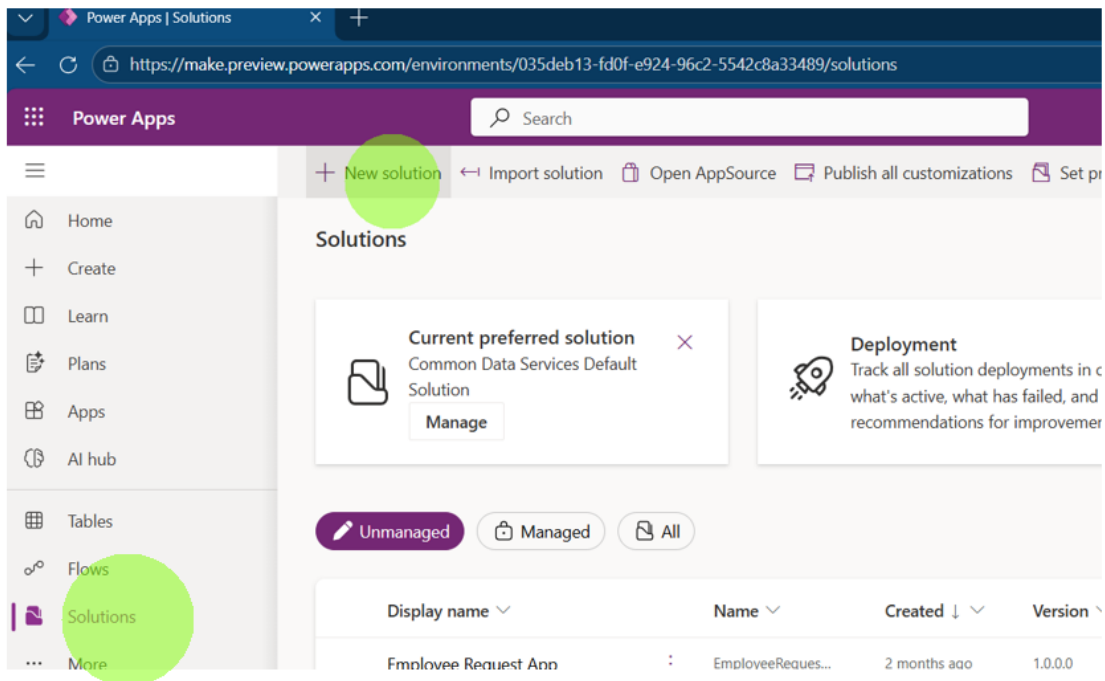
15 mins

Task 1: Importing Invoice Table Schema and Data

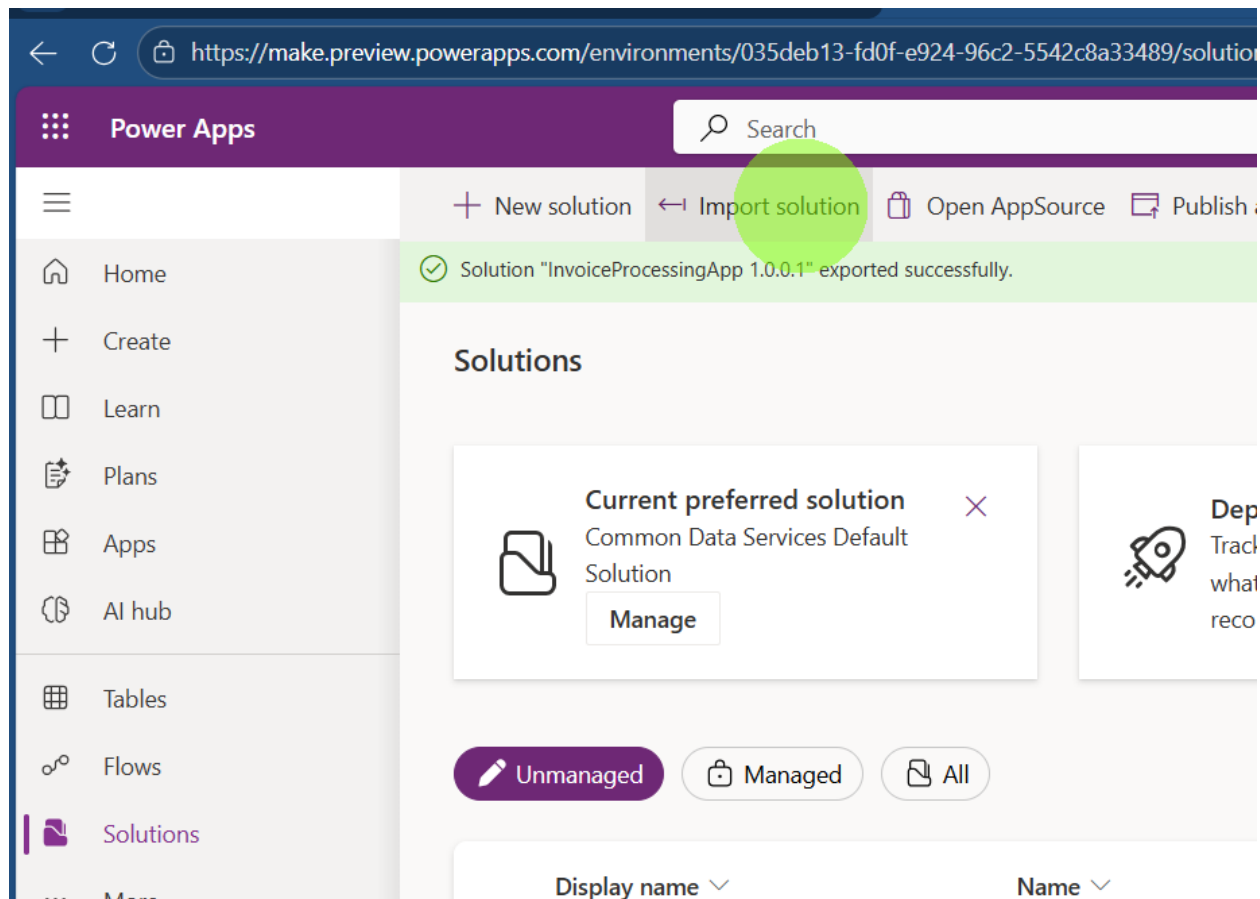
1. Go to make.powerapps.com.



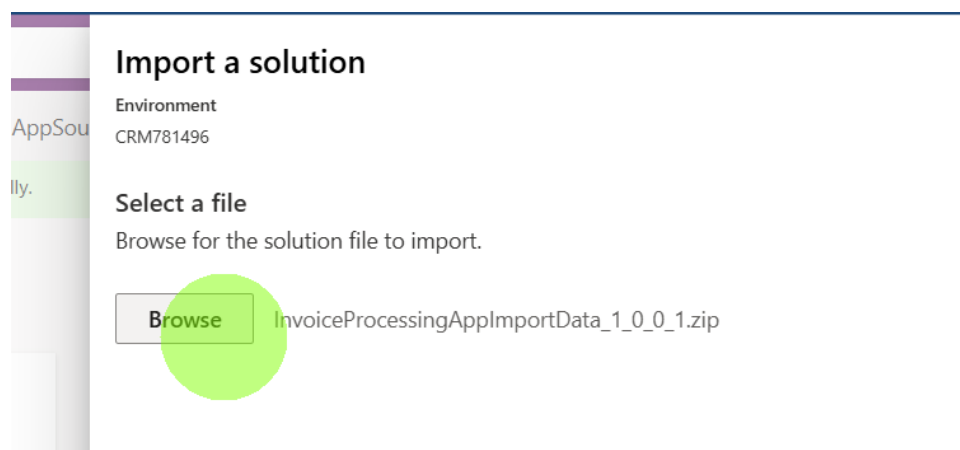
2. Click Solutions.



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.

InvoiceProcessingApplImportData_1_0_0_1.zip

Import a solution


Environment
CRM781496


Connections
Re-establish connections to activate your solution.


☒ Review and adjust all connections


0 updates needed. 2 updates optional.

6. Meanwhile upload invoices_last3months_simple_approval.xlsx to your OneDrive folder.

 invoices_last3months_simple_approval


 You frequently open
11m ago


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


 You recently opened
Jan 19


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	InvoiceProcessingAppImportData	14 minutes ago	1.0.0.1	CDS Default Publisher	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 Advanced

✓ 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

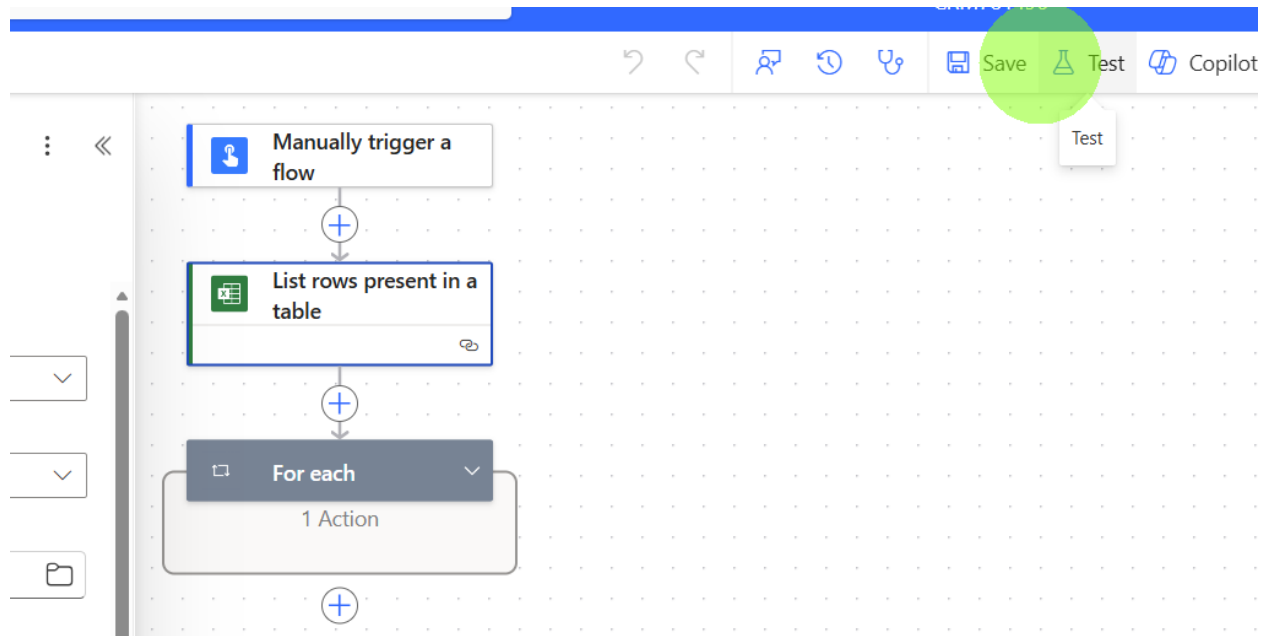
File *
/invoices_last3months_simple_approval.xlsx

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

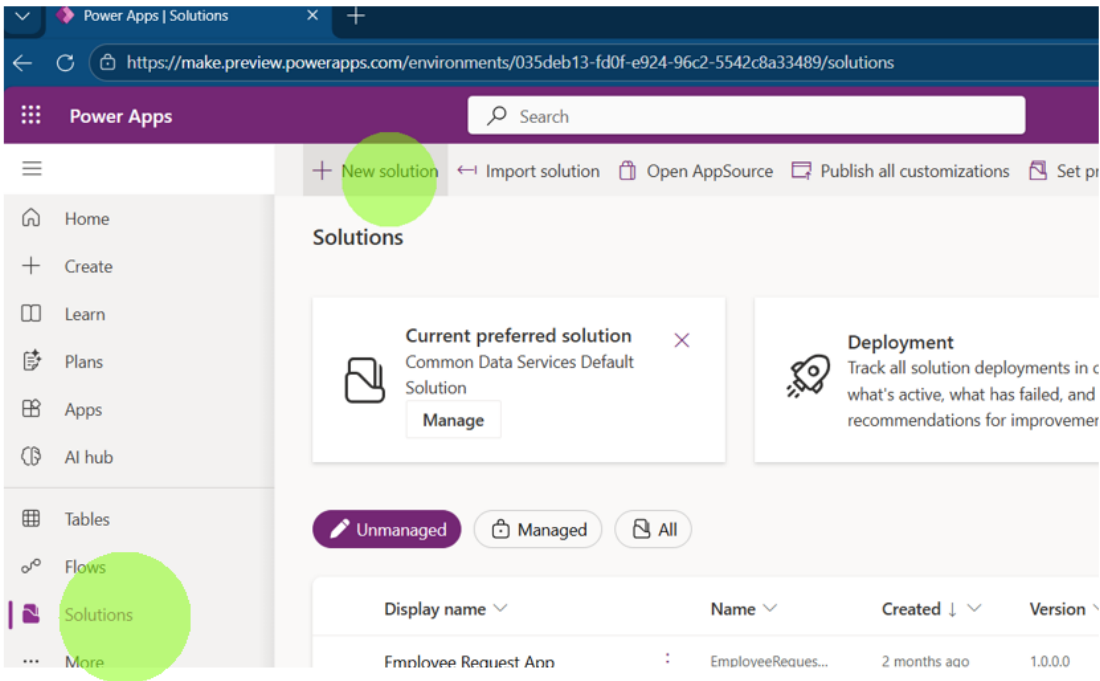
📊 Invoice columns and data Update forms and views Edit ▾

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

12. Go to 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

13. Click Solutions.

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

14. Click + New Solution.

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

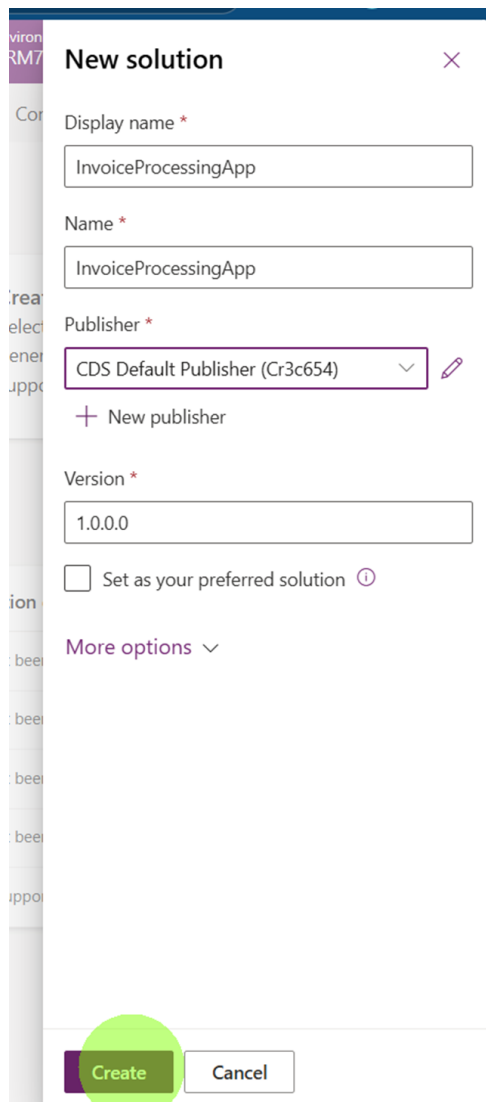
Current preferred solution
Common Data Services Default
Solution
Manage

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

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



16. 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.

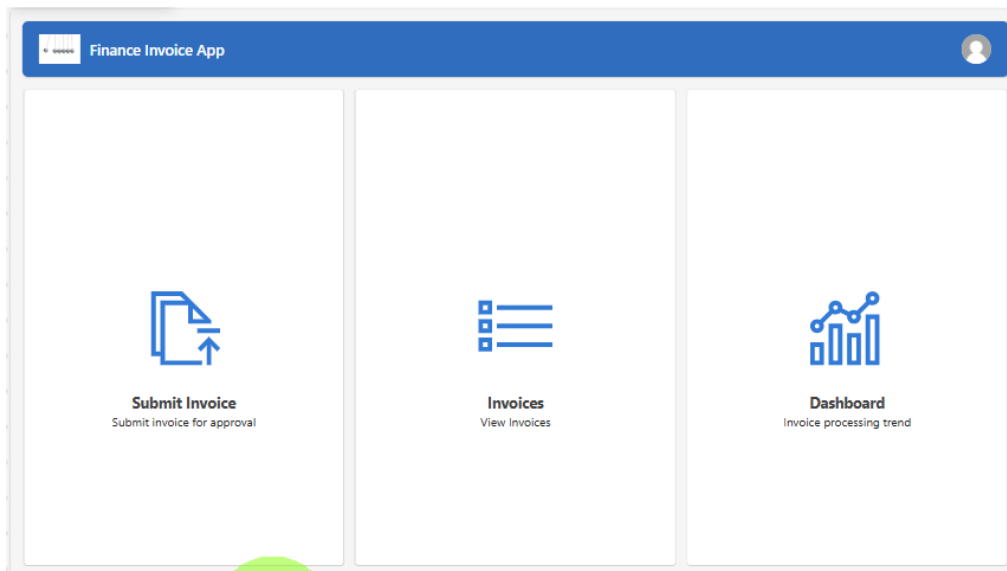
Objectives

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



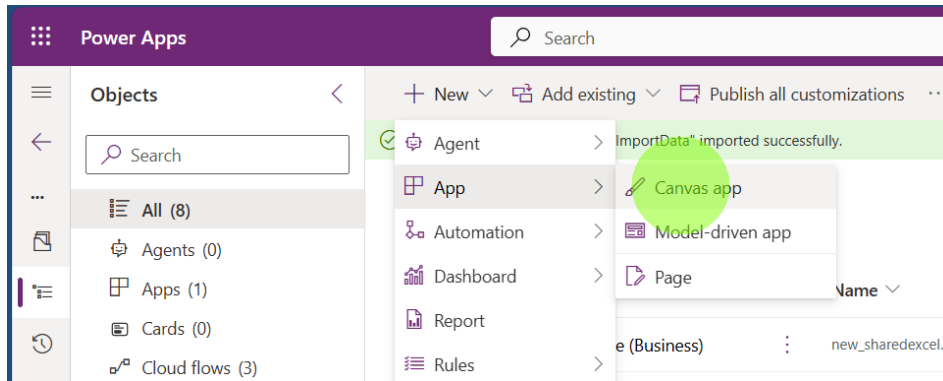
Estimated time to complete this lab

20 mins

Task 1: Creating the responsive app

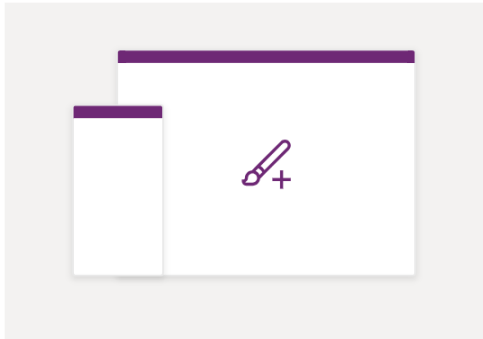
In this Task you will learn how to create a new app from scratch and how to make it responsive.

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 *

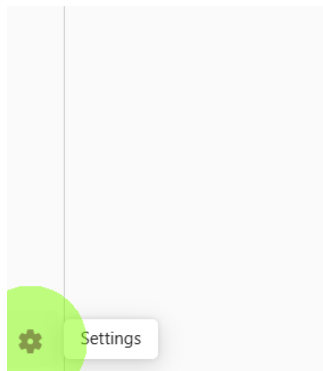
Format

☒ Tablet

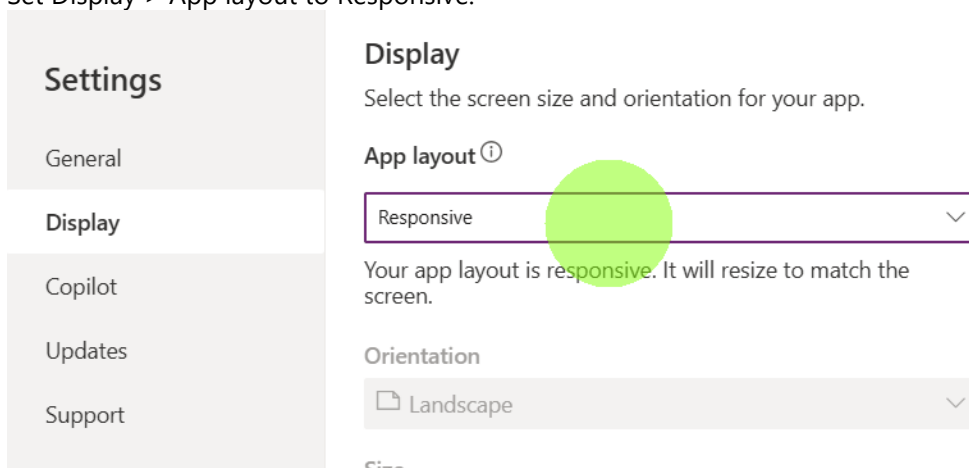
☐ Phone

 Create

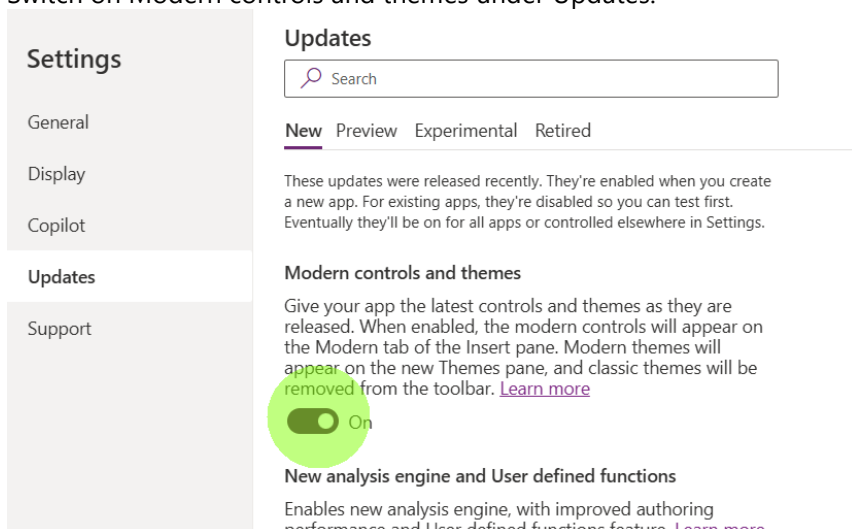
2. Once in the app designer, click setting icon at the bottom right corner.



Set Display > App layout to Responsive.



Switch on Modern controls and themes under Updates.



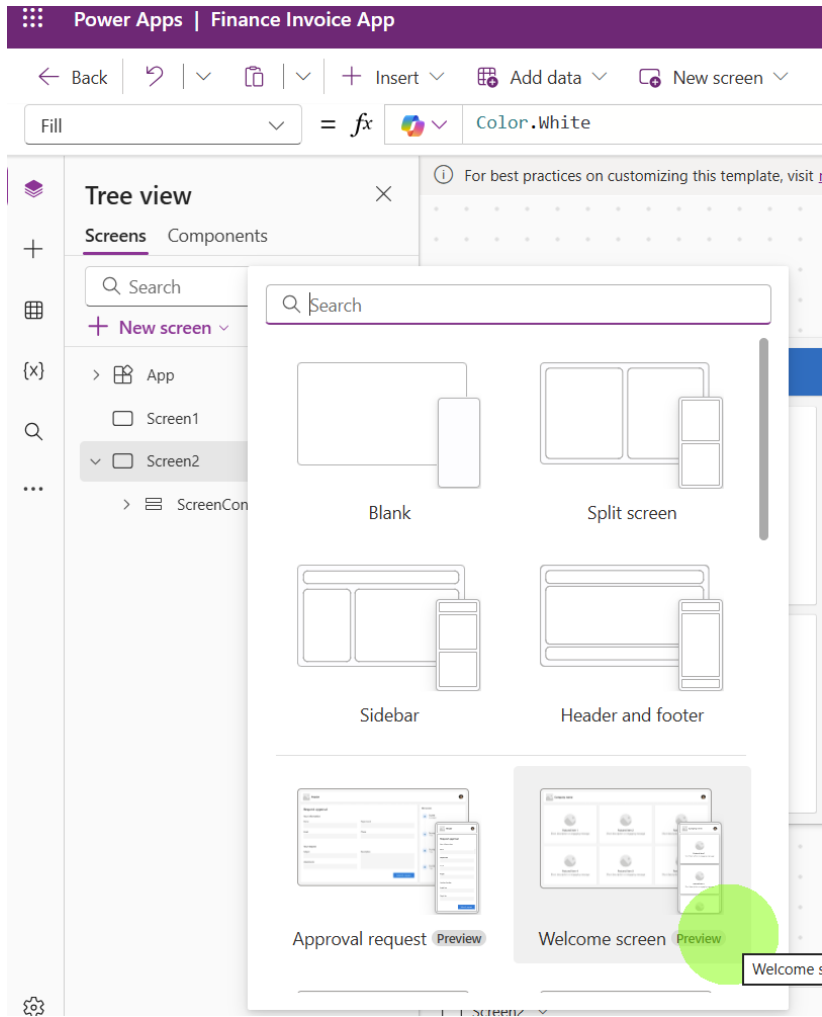
Close the Settings window.

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

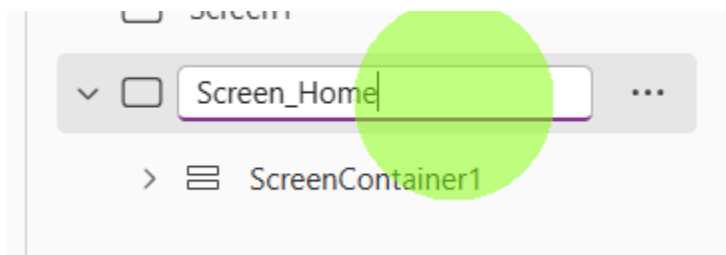
Task 2: Editing Home Screen

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

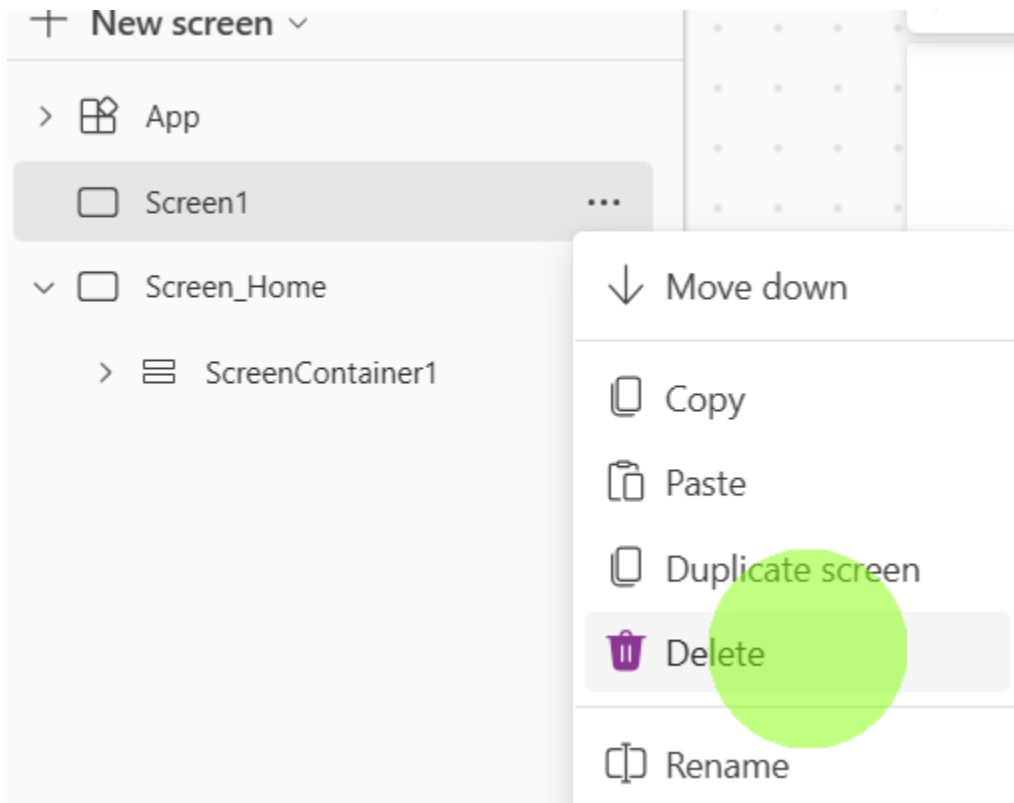
1. Click + New screen > Welcome screen.



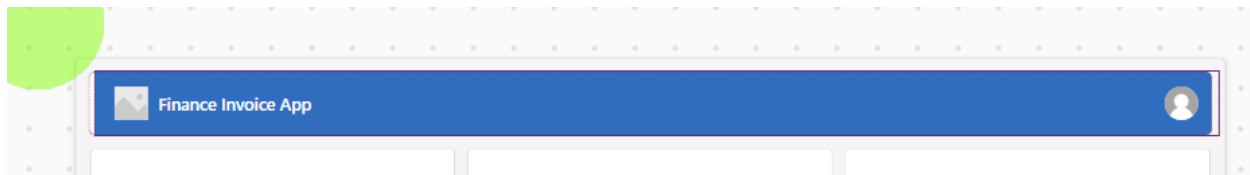
Rename the screen to Screen_Home.



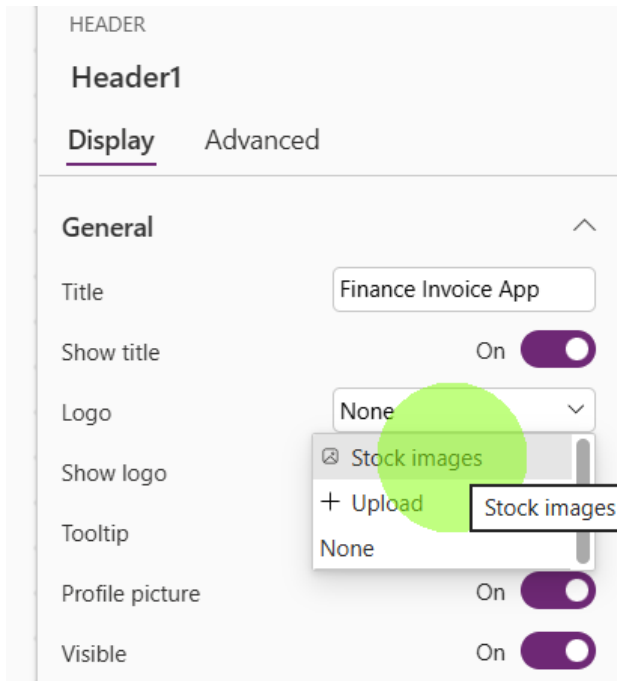
Delete Screen1.



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

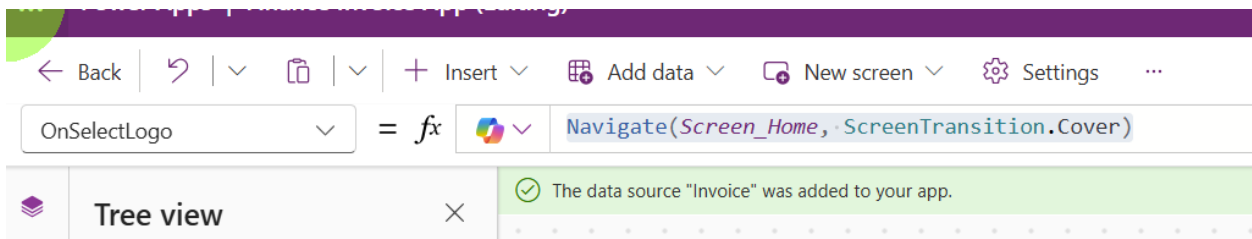


Click Logo and select any stock image that you like.

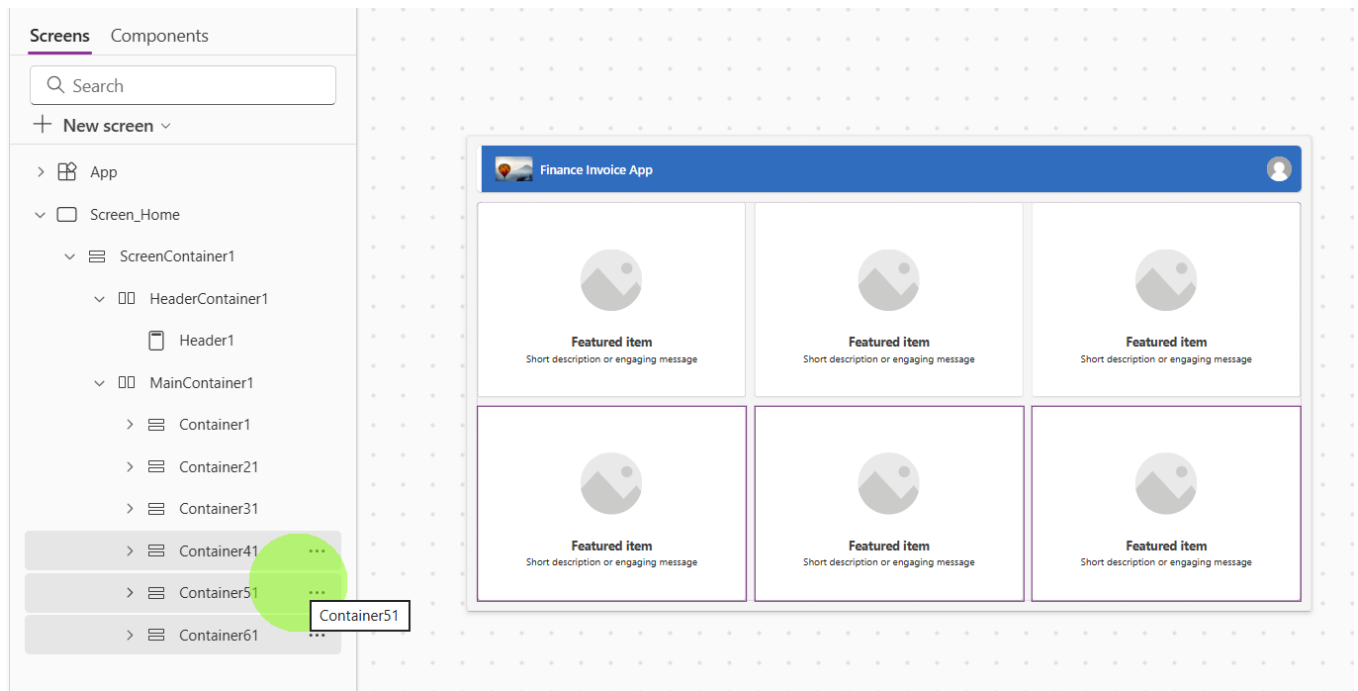


3. Change the OnSelectLogo property of Header to below.

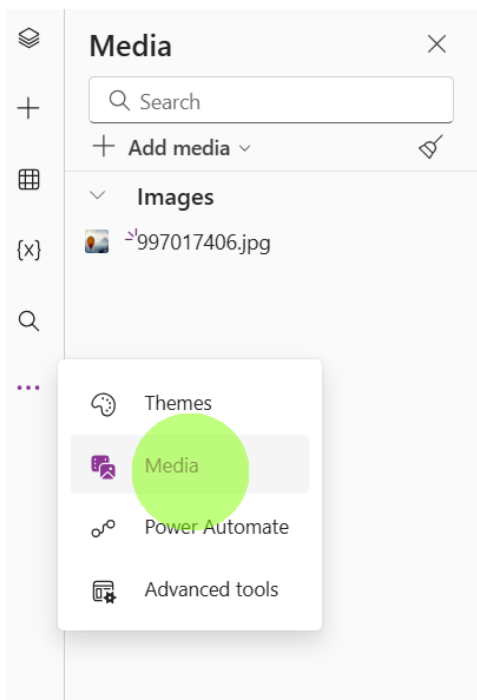
`Navigate(Screen_Home, ScreenTransition.Cover)`






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

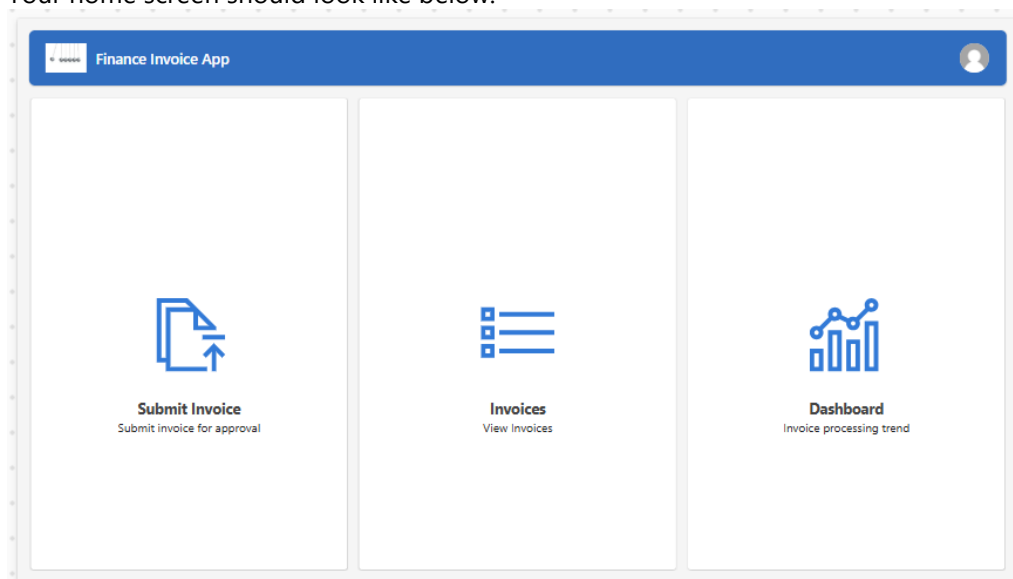
6. Select each icon on Home screen and update Image properties accordingly.

	Image Property	Border Radius
First Icon	Upload	0
Second Icon	List	0
Third Icon	Dashboard	0

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

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

Your home screen should look like below.

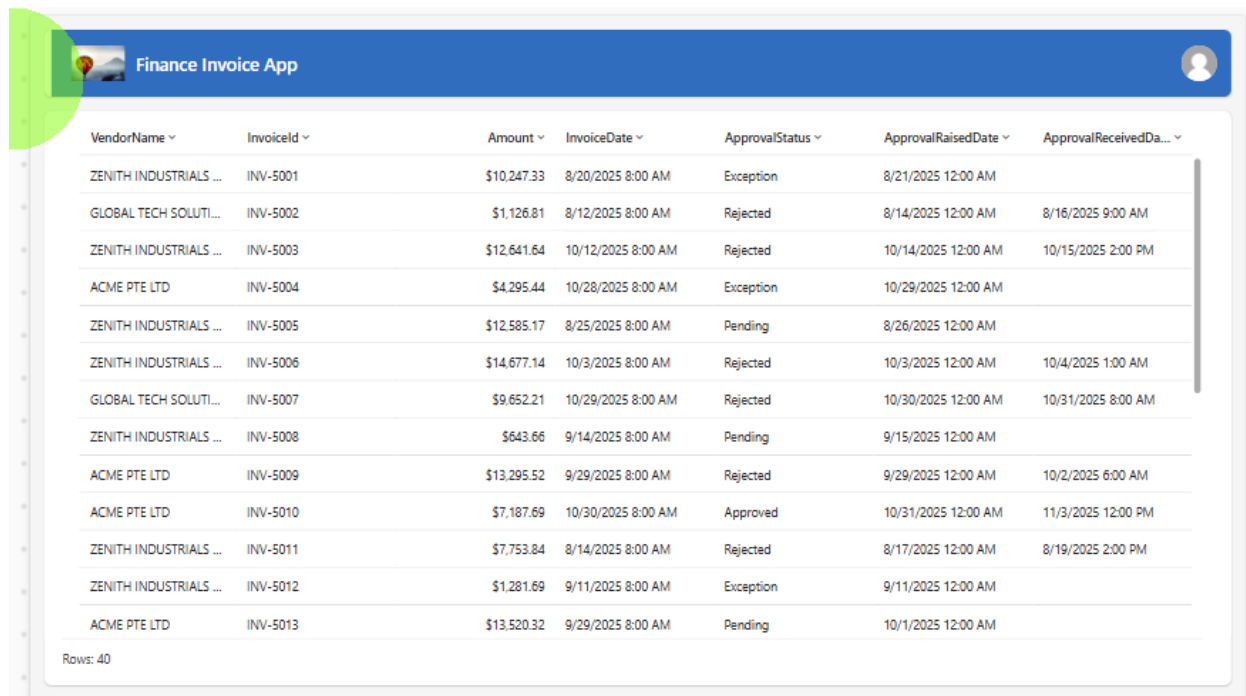


Important! Remember to Save the App.

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 table. 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.

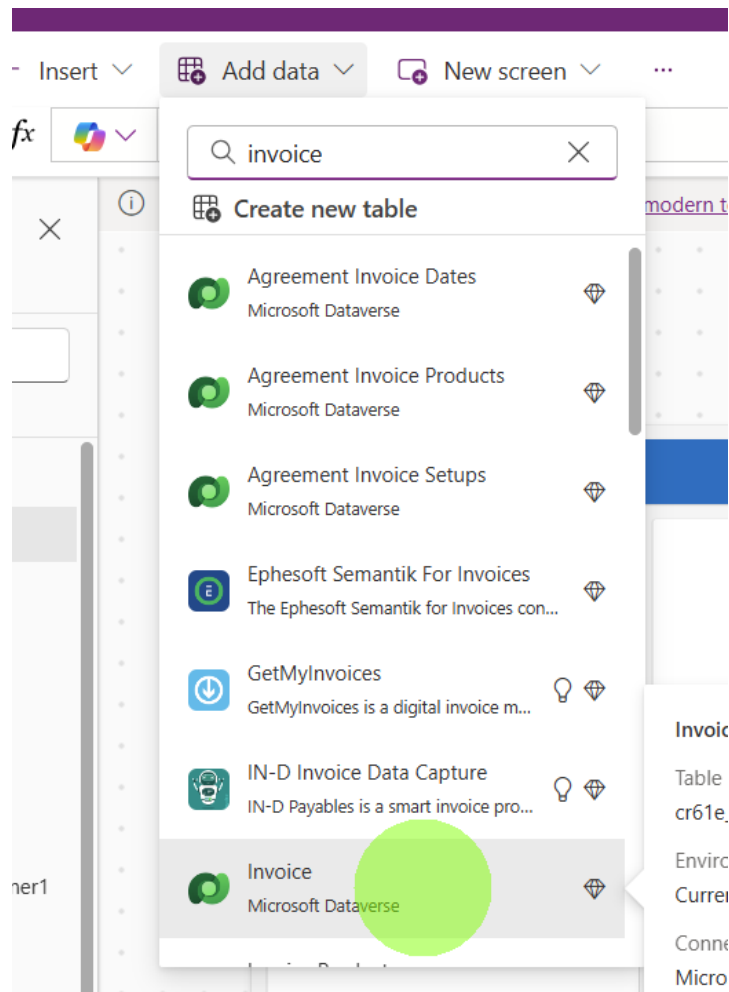
- Learn how to connect to a Dataverse table
- Display data in a Table control
- Filter data displayed in Table control

Estimated time to complete this lab

20 mins

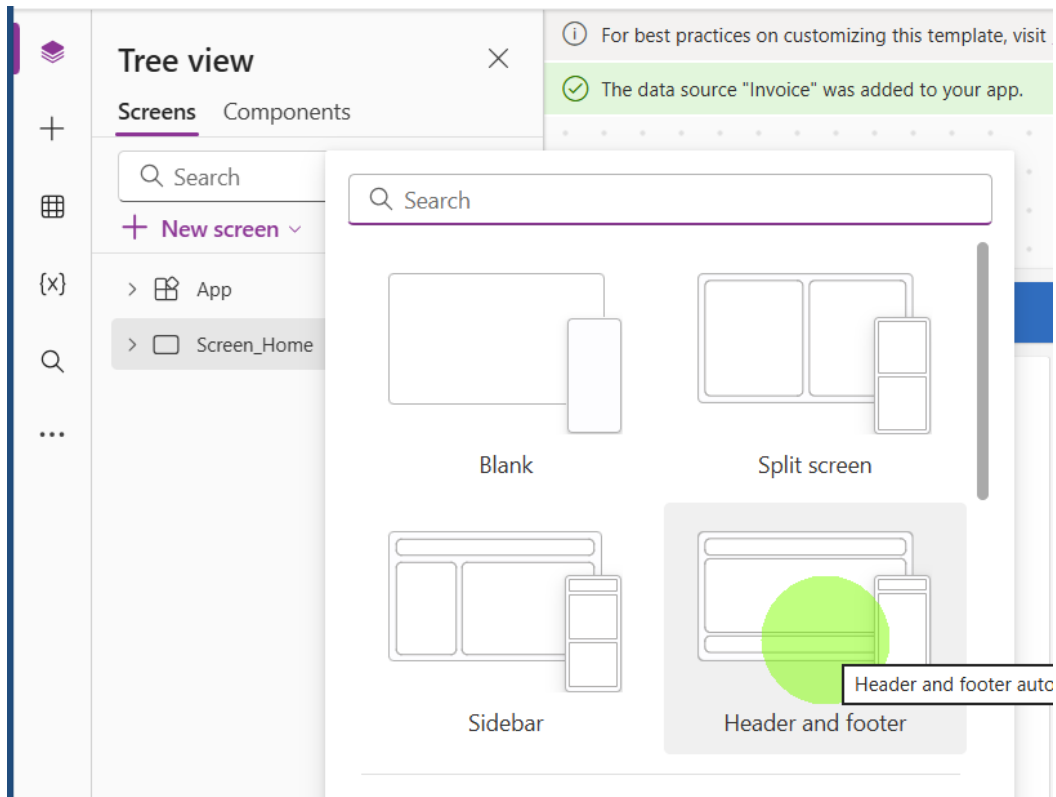
Task 1: Connecting to the Dataverse Table

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

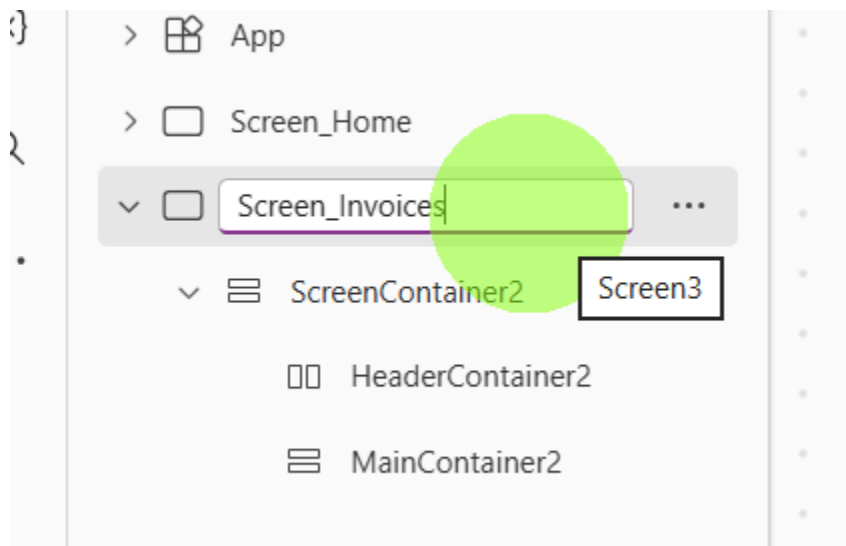


Task 2: Displaying the Data in a Table Control

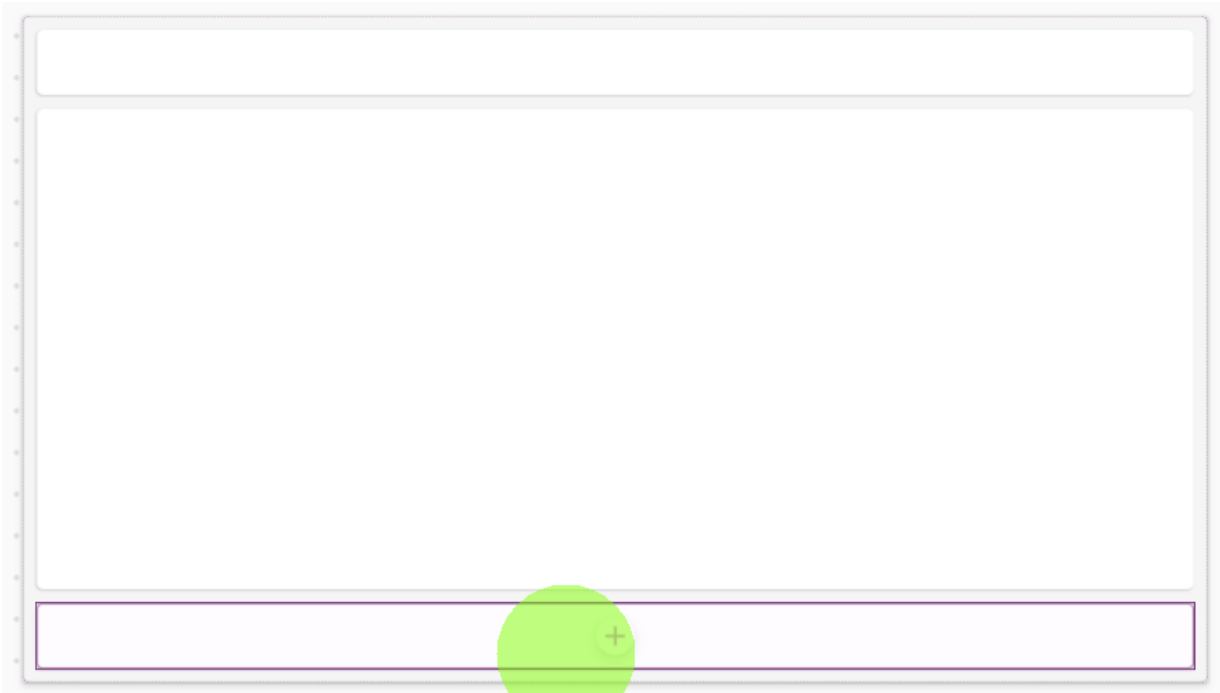
1. Click New screen > Header and footer to add a new screen.



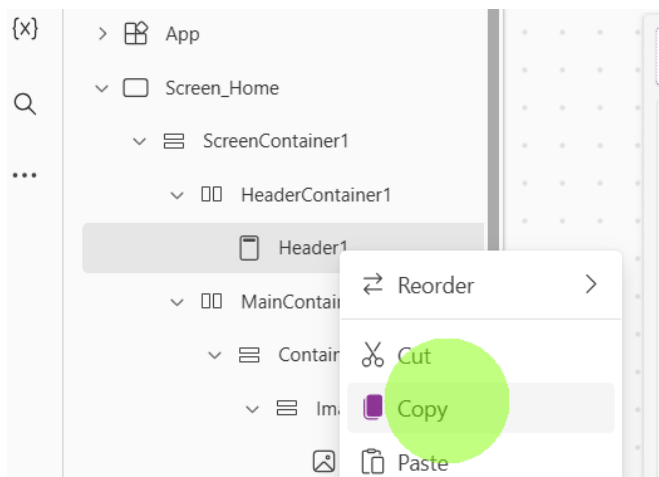
2. Rename the screen to Screen_Invoices.

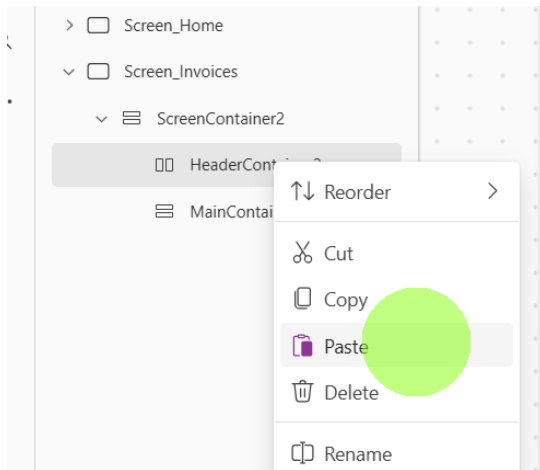


3. Select the bottom container and delete.

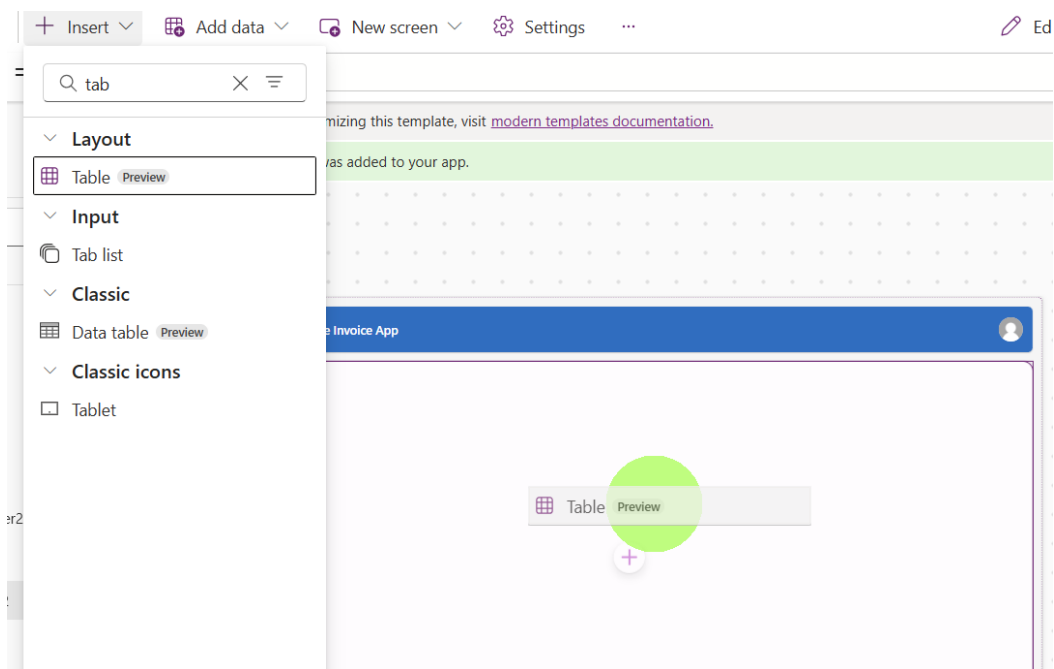


4. Copy Header1 from Screen_Home and paste in Screen_Invoices's Header Container.

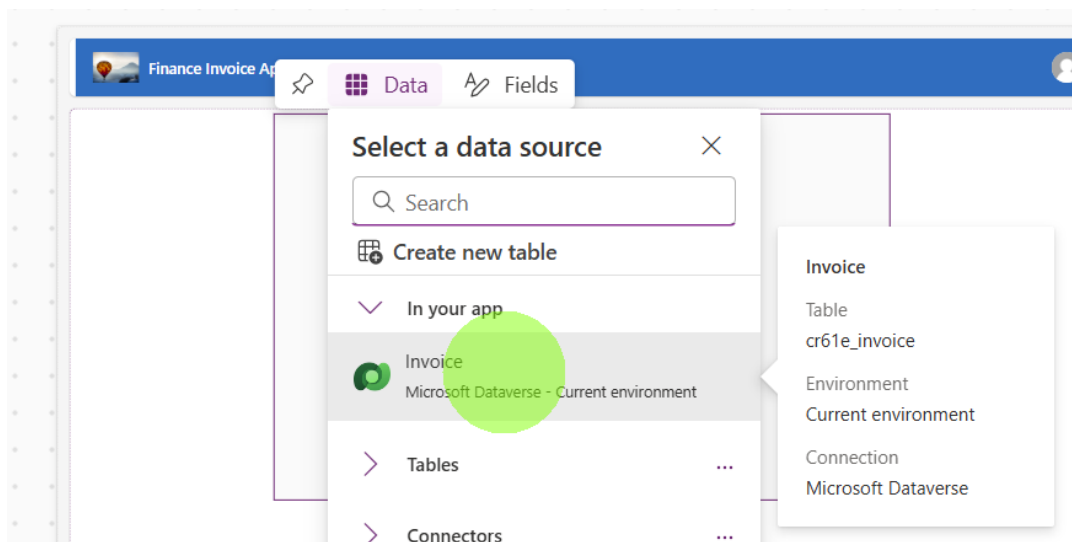




8. Click Insert > Table into the MainContainer.

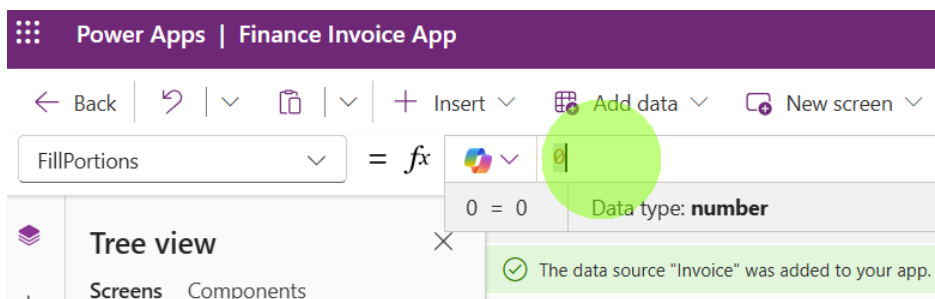
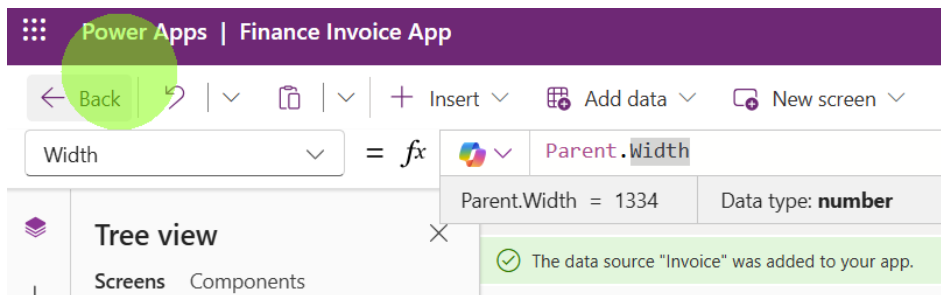


Select Invoice table as data source for the table.



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

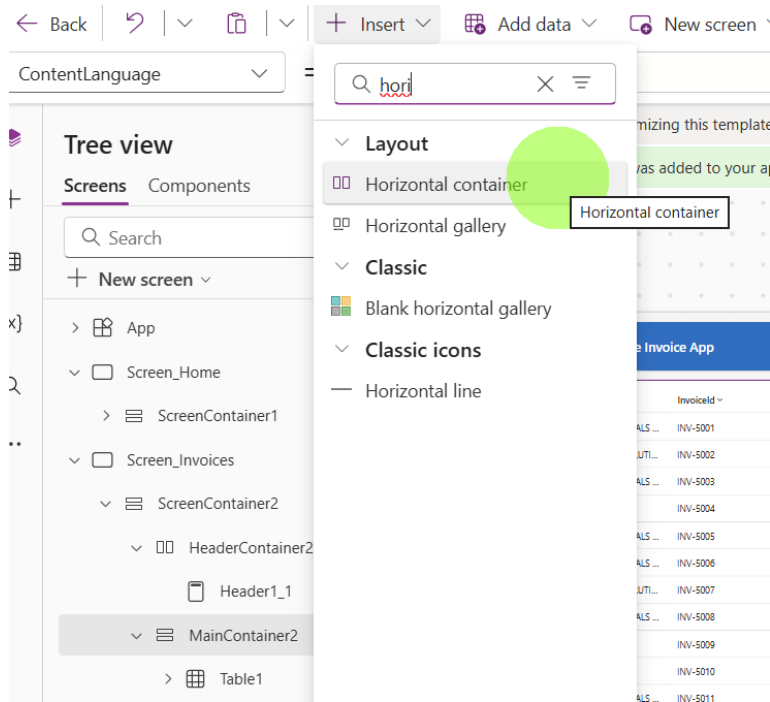
- Filter a table by Vendor Name
- Filter a table by Date Range

Estimated time to complete this lab

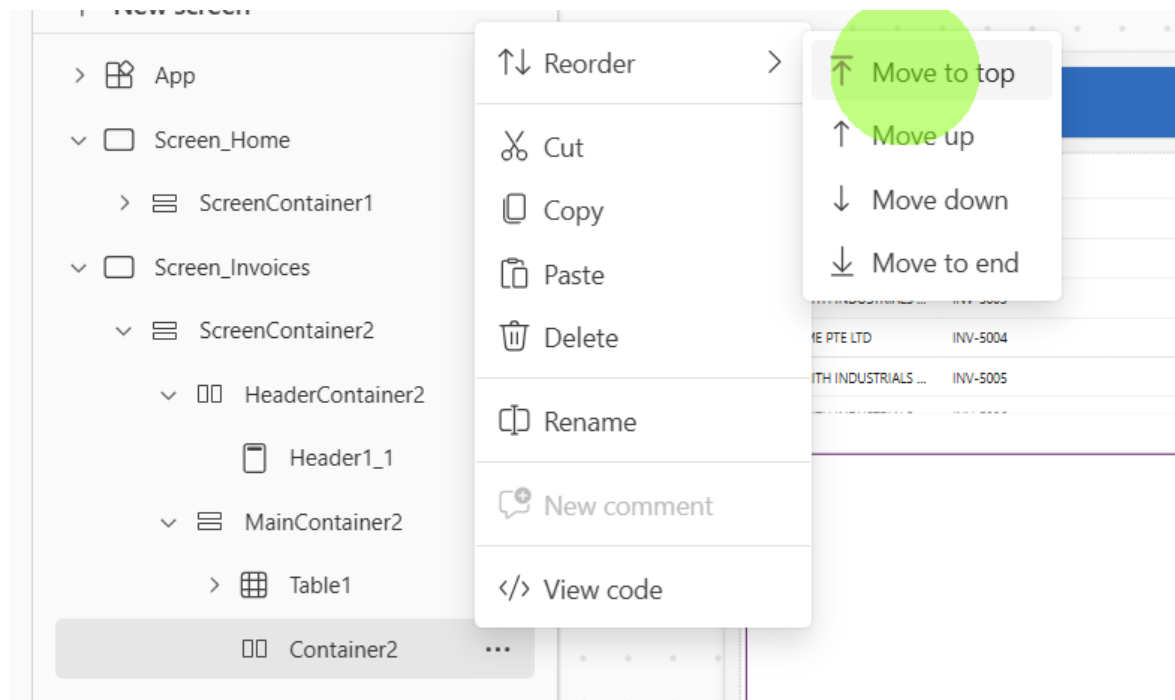
20 mins

Task 1: Creating Filter Controls

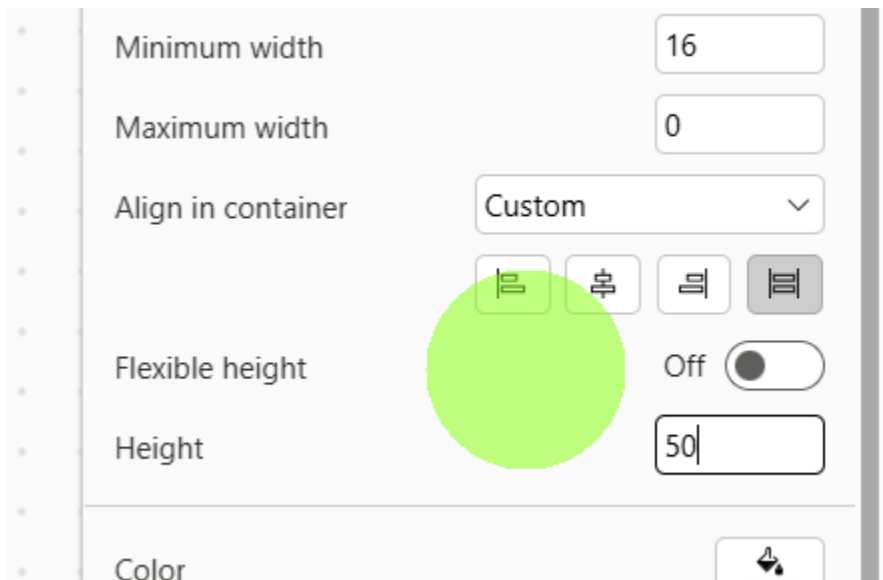
1. Insert a new Horizontal Container to Main Container.



Reorder the new Container to top.

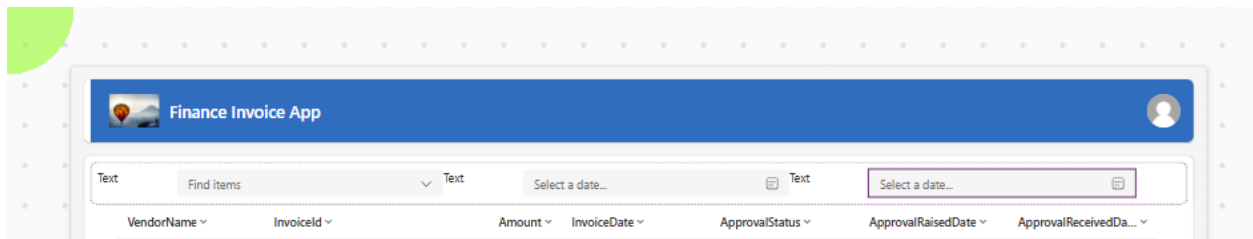


Switch Off Flexible height and set Height to 50.



2. Add 6 controls below to the container.

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

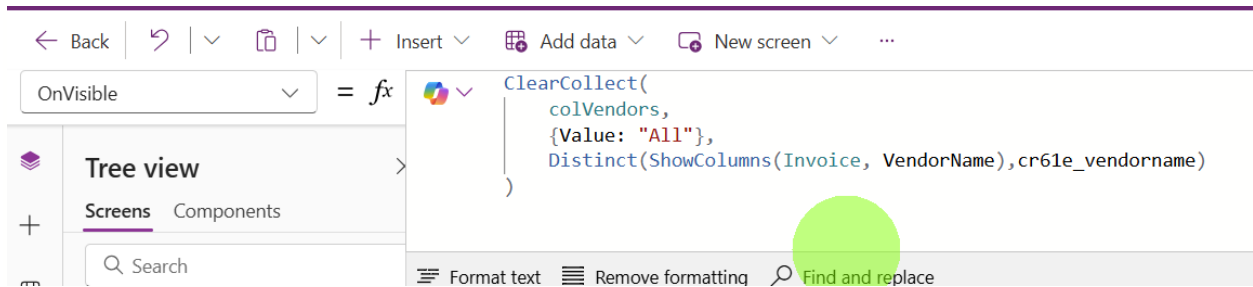


3. Change properties of those controls to below.

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

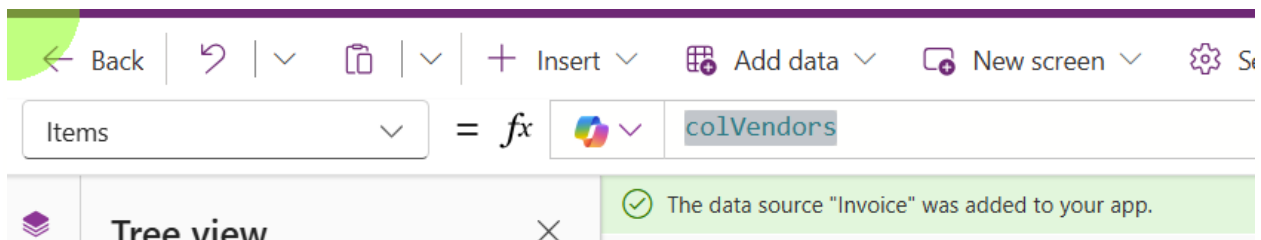
4. Insert this formula into OnVisible property of Screen_Invoices.

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



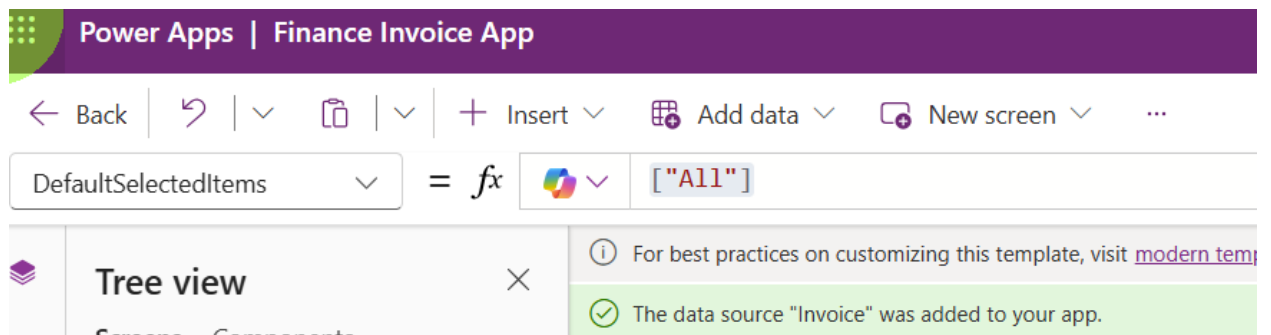
5. Change Items property of Combo_VendorName to below.

`colVendors`



11. Change DefaultSelectedItems property of Combo_VendorName to below.

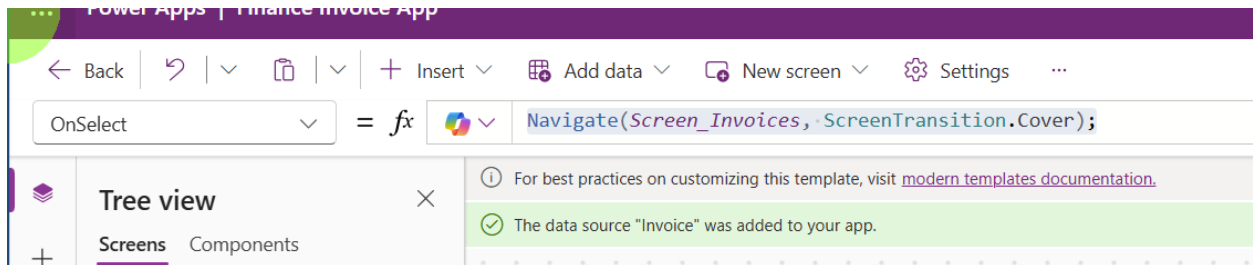
`["All"]`



Task 2: Applying Filter

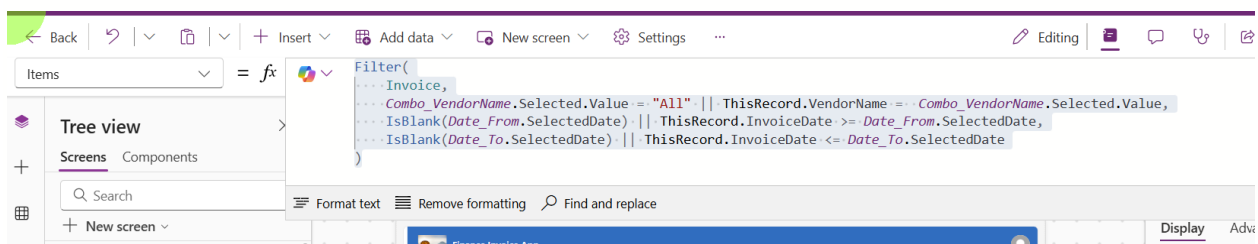
1. Click Screen_Home. And set OnSelect property of Invoices Icon to below.

`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
)
```



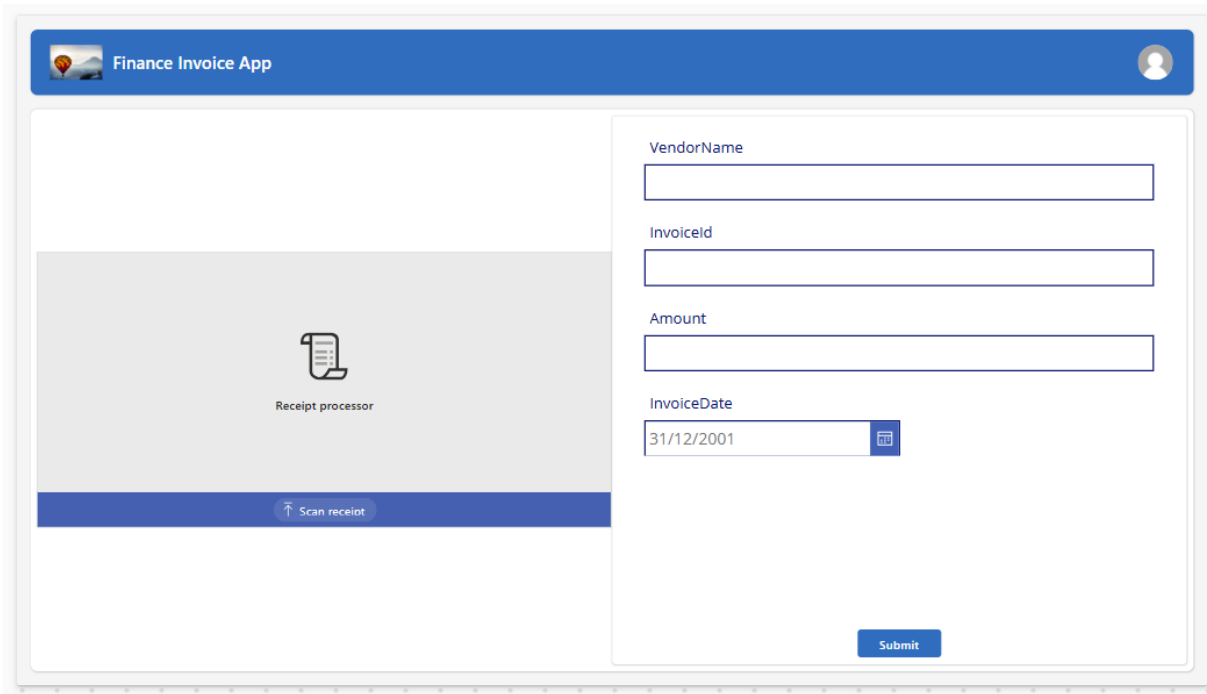
You should see the vendor list populated and be able to filter the table by vendor name, date range.

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 with an upward arrow and the text '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 input with a calendar icon). The 'InvoiceDate' field is pre-filled with '31/12/2001'. At the bottom right of the form is a blue 'Submit' button.

Objectives

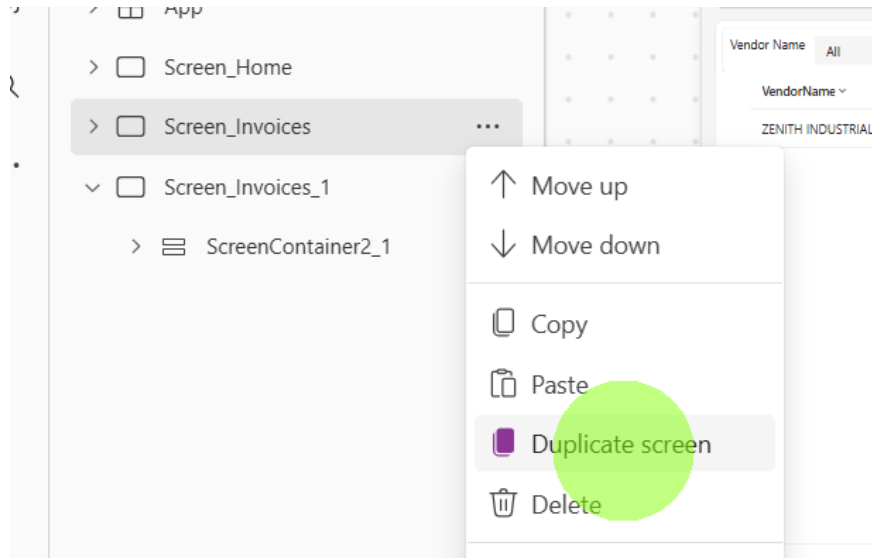
- User Receipt Processor to extract values from PDF
- Add in form that allows users to create a new request

Estimated time to complete this lab

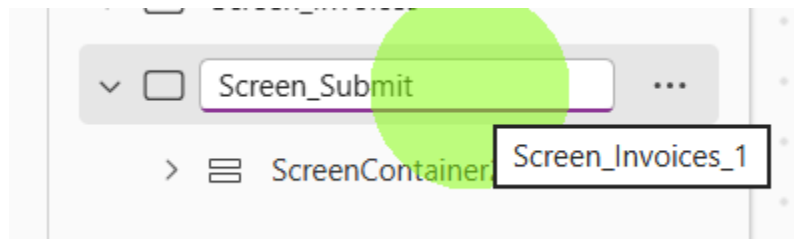
30 mins

Task 1: Duplicate the existing Screen

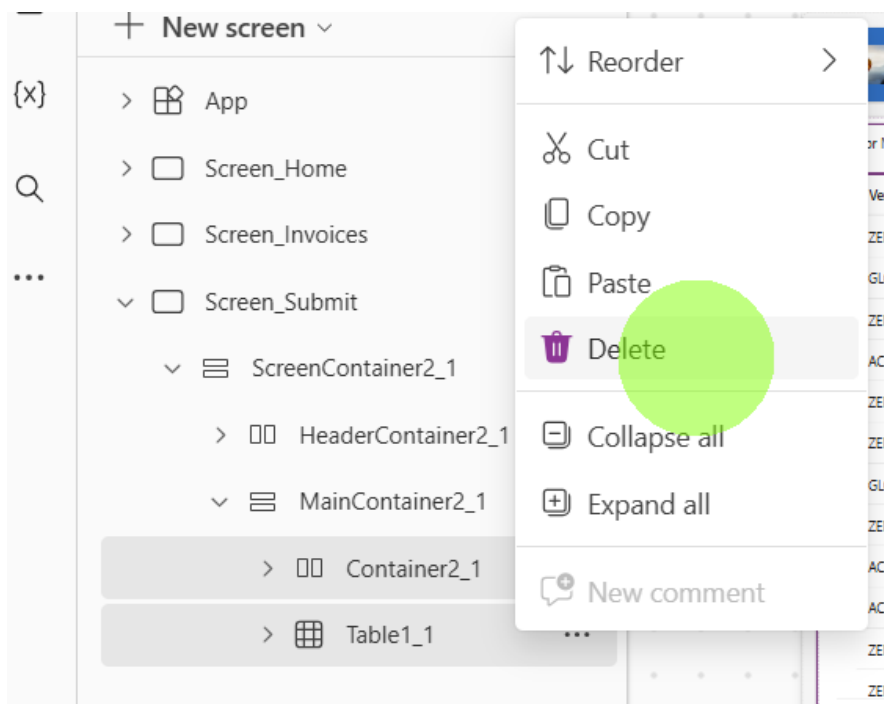
1. Duplicate Screen_Invoices by following selecting the screen, right clicking it, and then selecting the 'Duplicate Screen' option.



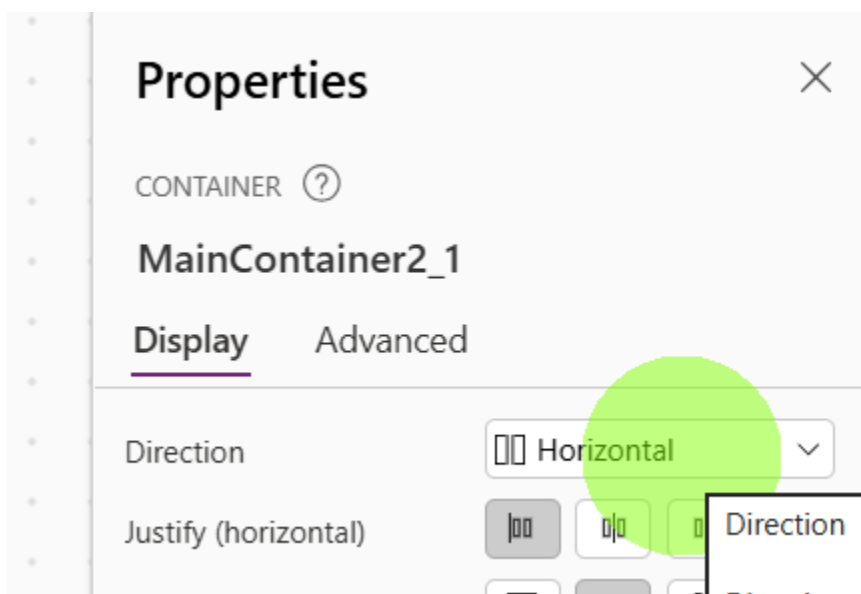
Rename the new screen to Screen_Submit.



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

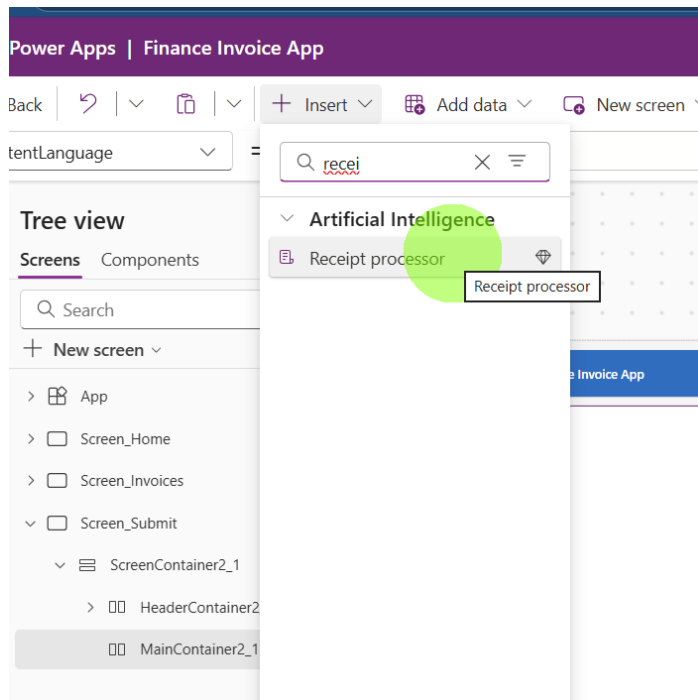


3. Change Main Container's Direction to Horizontal.



Task 2: Adding Controls to the Screen

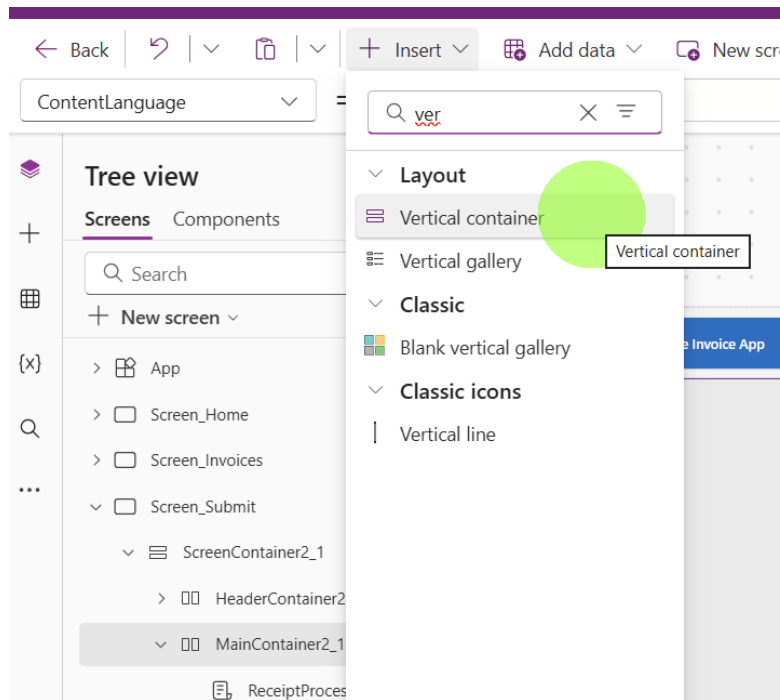
1. Select Main Container and add Receipt Processor control.



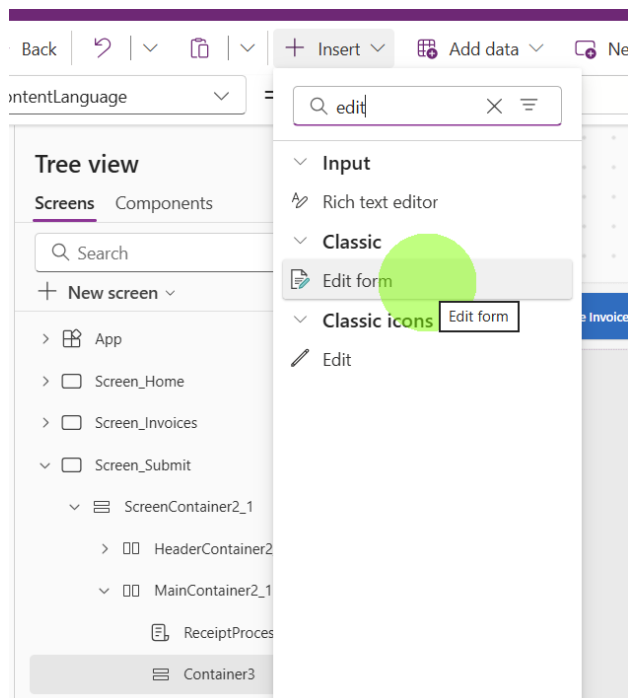
Change below properties of the Receipt Processor.

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

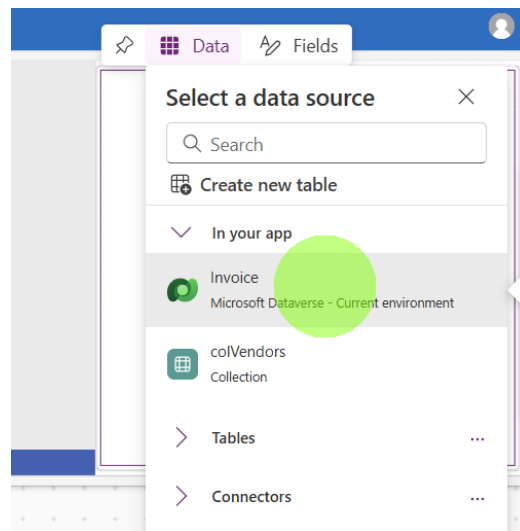
2. Add a new Vertical Container into the Main Container.



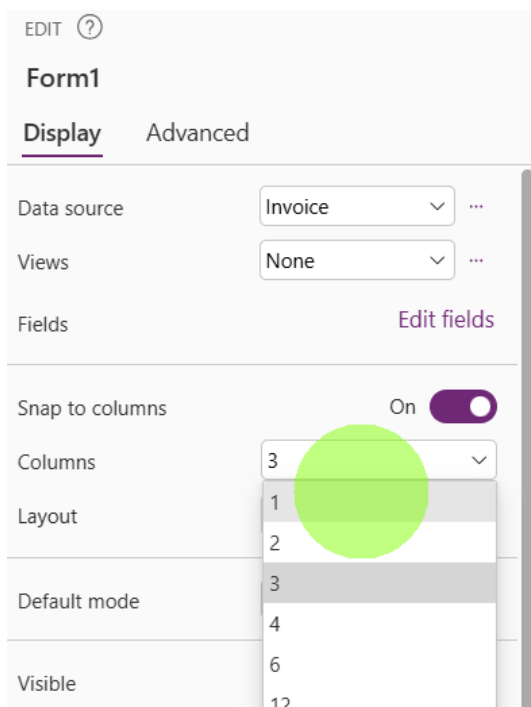
3. Add a new Form into the Vertical Container.



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



5. Change the number of Columns to 1

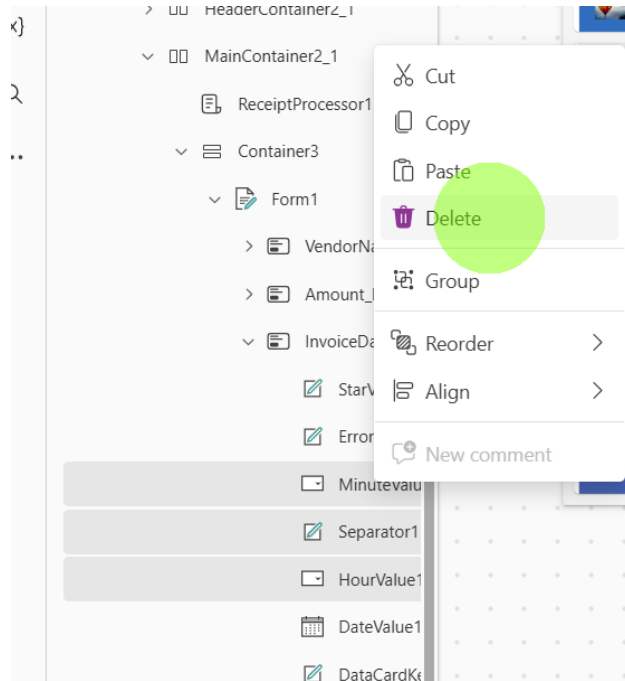


6. Select Fields to edit the fields displayed and only keep these 4 fields below.

The screenshot displays the Microsoft Power Platform interface. On the left, a form titled 'Form1' is visible with fields for 'VendorName', 'InvoiceId', 'Amount', and 'InvoiceDate'. The 'InvoiceDate' field contains the value '31/12/2001'. In the center, the 'Fields' panel is open, showing a list of fields: 'VendorName', 'InvoiceId', 'Amount', and 'InvoiceDate'. Each field has a dropdown arrow and a small icon. On the right, the 'Properties' panel 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'. The 'Fields' section shows '4 selected'. The 'Snap to columns' toggle is turned 'On'. The 'Columns' dropdown is set to '1'. The 'Layout' dropdown is set to 'Vertical'. The 'Default mode' dropdown is set to 'Edit'. The 'Visible' toggle is turned 'On'.

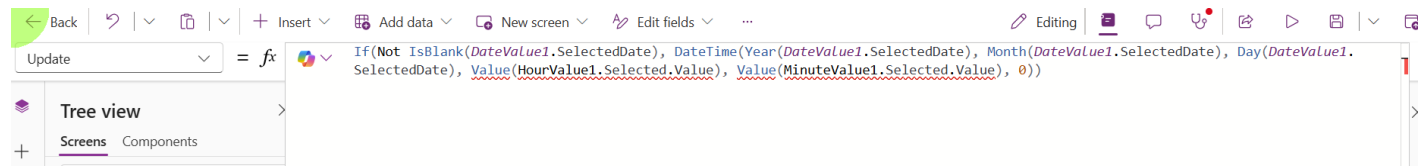
7. Unlock Invoice Date Field and delete Hours, Minutes and Separator controls.

The screenshot shows a close-up of the 'InvoiceDate' field in the form. The field is labeled 'Card : InvoiceDate' and contains the value '31/12/2001'. A context menu is open over the field, showing options: 'Unlock', 'Paste', and 'New comment'. The 'Unlock' option is highlighted with a green circle.



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

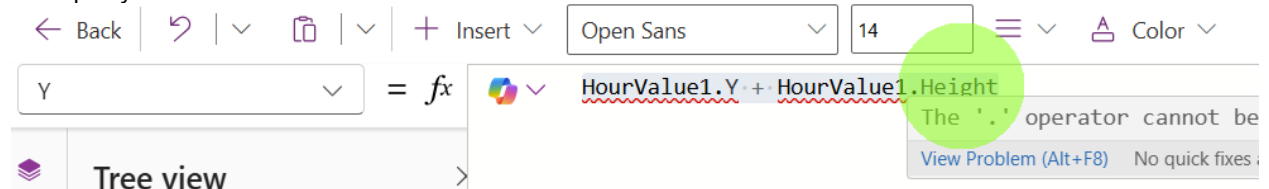
Update Property



Value(HourValue1.Selected.Value) → 0

Value(MinuteValue1.Selected.Value) → 0

Y Property



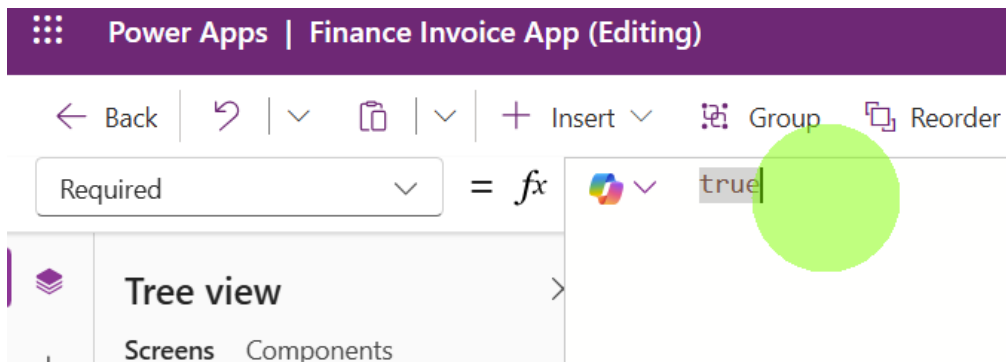
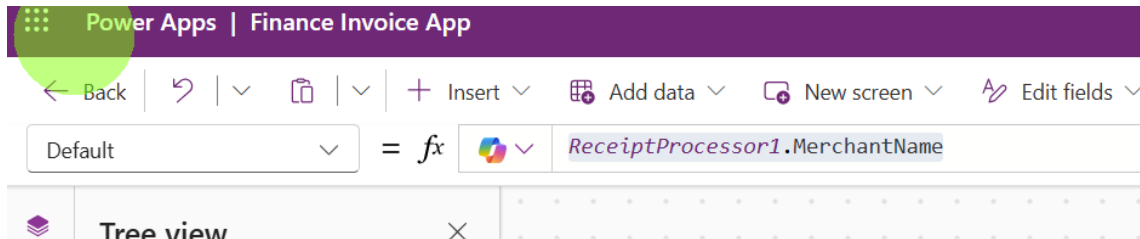
HourValue1.Y + HourValue1.Height → 0

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

Task 3: Extracting Values from Receipt Processor

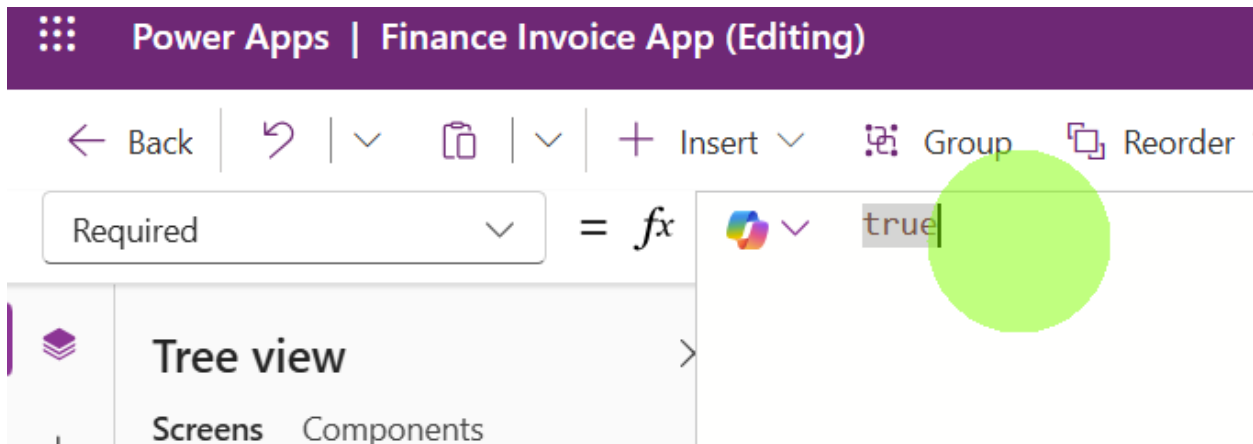
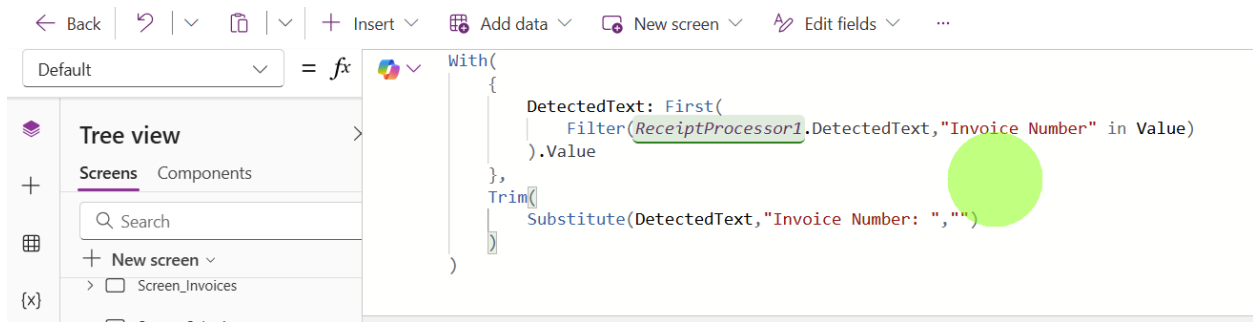
1. Unlock VendorName Card and set Default property to below and Required to true.

ReceiptProcessor1.MerchantName



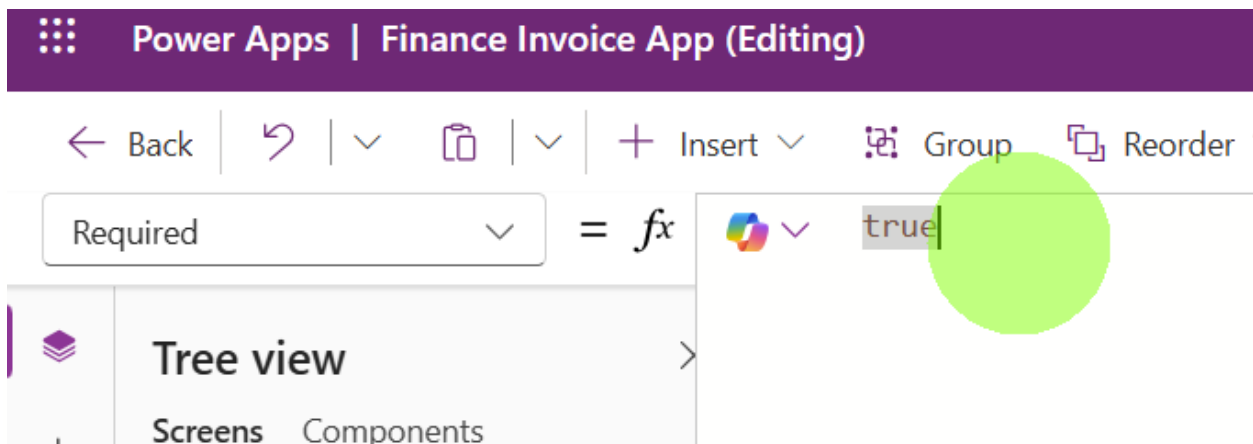
2. Unlock InvoiceId Card and set Default property to below and Required to true.

```
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

The screenshot shows the Power Apps editor interface for a 'Finance Invoice App'. The top bar includes navigation icons and a 'Tree view' pane on the left. The main workspace displays the 'Default' property of a card, which is set to the formula `ReceiptProcessor1.TransactionDate`. Below this, the 'Required' property is set to `true`. A large green circle highlights the formula bar area.

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

NewForm(Form1);

The screenshot shows the Power Apps editor interface for a 'Finance Invoice App'. The top bar includes navigation icons and a 'Tree view' pane on the left. The main workspace displays the 'OnChange' property of a card, which is set to the formula `NewForm(Form1);`. A large green circle highlights the formula bar area.

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

The screenshot shows the 'Finance Invoice App' interface. On the left, there's a preview of the invoice form. The form includes a header with the company name 'GLOBAL TECH SOLUTIONS PTE LTD', an invoice number 'INV-3001', and an invoice date '23/01/2025'. The main body of the invoice is a table with columns: Description, Qty, Unit Price (\$G0), and Line Total (\$G0). The table contains five rows of data. At the bottom of the table, there's a 'Subtotal' row and a 'Total (\$G0)' row. On the right side of the app, there are input fields for 'VendorName', 'Invoiceld', 'Amount', and 'InvoiceDate'. The 'VendorName' field is filled with 'GLOBAL TECH SOLUTIONS PTE LTD', 'Invoiceld' with 'INV-3001', 'Amount' with '15685.1', and 'InvoiceDate' with '23/01/2025'.

Description	Qty	Unit Price (\$G0)	Line Total (\$G0)
Apple's Macintosh (MacBook)	1	1568.00	1568.00
Microsoft's Surface Tablet	2	1284.00	2568.00
Apple's Macintosh (MacBook)	2	1284.00	2568.00
Microsoft's Surface Tablet	2	1284.00	2568.00
Apple's Macintosh (MacBook)	2	1284.00	2568.00
Subtotal			15685.10
Total (\$G0)			15685.10

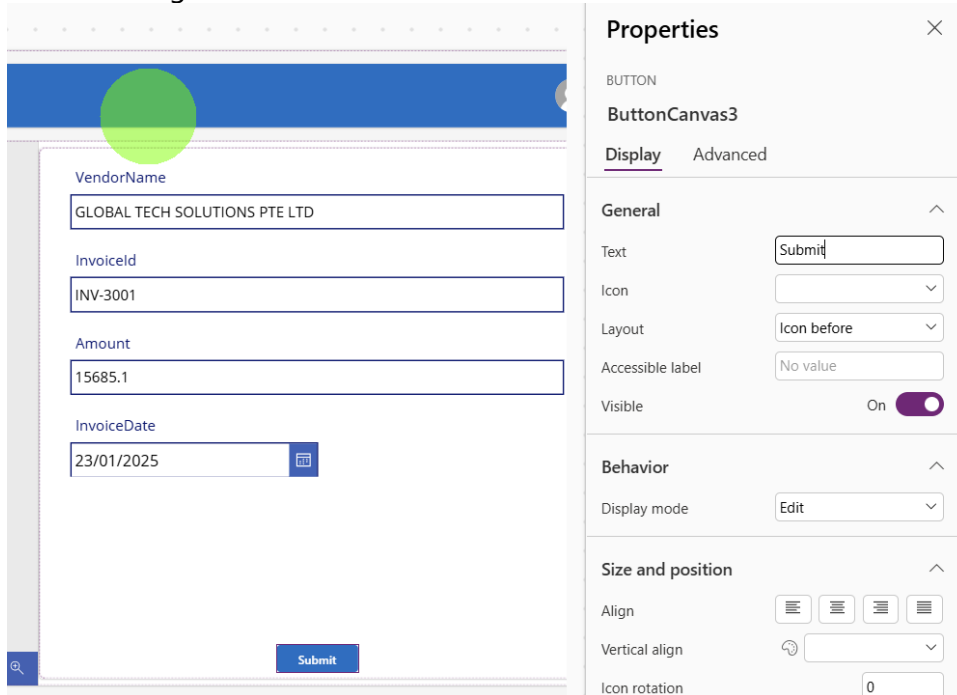
----- End of Task -----

Task 4: Submitting Extracted Values

1. Add a button below the form.

The screenshot shows the Power Apps designer interface. The 'Insert' menu is open, and the 'Button' option is highlighted with a green circle. The 'Tree view' on the left shows the app structure, including 'Screen_Invoices', 'Screen_Submit', and 'ScreenContainer2_1'. The 'MainContainer2_1' is selected. The 'Insert' menu also shows other options like 'Text', 'Form', 'Table', 'Icon', 'Grid container', and 'Card'. The 'Button' option is highlighted with a green circle, and a tooltip 'Button' is visible next to it. The background shows a preview of the 'Finance Invoice App' interface.

2. Change Text to Submit



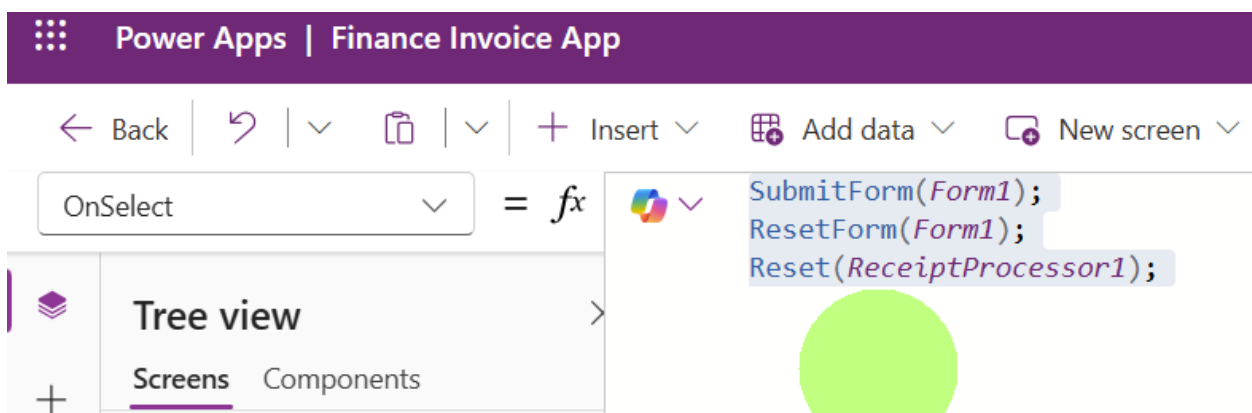
The screenshot shows a Power Apps form with the following fields:

- VendorName: GLOBAL TECH SOLUTIONS PTE LTD
- InvoiceId: INV-3001
- Amount: 15685.1
- InvoiceDate: 23/01/2025

At the bottom right is a blue button labeled "Submit". The Properties pane on the right is open, showing the "Display" tab. The "Text" property is set to "Submit".

3. Set OnSelect property of Submit button to this.

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



The screenshot shows the Power Apps interface with the "OnSelect" property of the Submit button set to the following formula:

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

The "Tree view" on the left shows the "Screens" tab selected. The "Properties" pane on the right shows the "OnSelect" property set to the formula.

4. Set DisplayMode property of Submit button to this.

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



... Power Apps | Finance Invoice App

← Back | ↶ | ∨ | 📄 | ∨ | + Insert ∨ | 📊 Add data ∨ | 🖨️ New screen ∨ | ⚙️ Settings | ...

DisplayMode ∨ = *fx* | 🌈 ∨ | `If(Form1.Valid, DisplayMode.Edit, DisplayMode.Disabled)`

Tree view >

Screens Components



Exercise 6: Creating Message Popups using Components

Utilising components to create reusable message popups.

The screenshot shows a form titled 'Employee Requests' within a Power Platform interface. The form is designed for managing and tracking requests. It includes the following fields and components:

- Request Title:** A text input field containing 'Client Meeting Room Booking'.
- Employee Name:** A text input field containing 'System Administrator'.
- Date of Request Starting:** A date picker field showing '05/05/2025'.
- Date of Request Ending:** A date picker field showing '05/05/2025'.
- Request Details:** A text input field containing 'Request to book meeting room for client visit'.
- * Request Category:** A dropdown menu with 'Leave' selected.
- Approver Assigned:** A text input field.
- Request Status:** A text input field containing 'Submitted'.
- Request Outcome:** A text input field.
- Deadline:** A text input field.
- Submit:** A blue button at the bottom right of the form.

The form is set against a light gray background with a dotted grid pattern. The top of the form has a header bar with the title 'Employee Requests' and a back arrow icon. The bottom of the form has a 'Submit' button.

Objectives

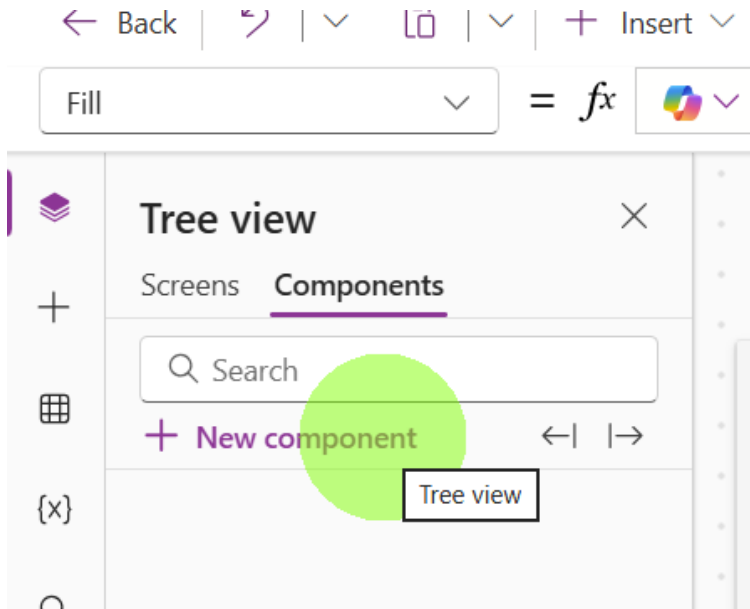
- Understand the use of components and custom properties

Estimated time to complete this lab

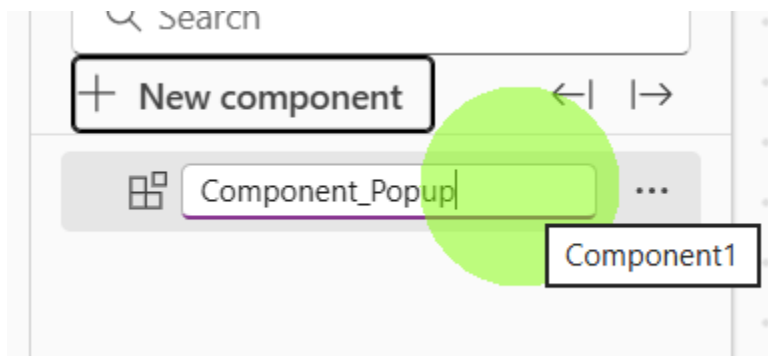
20 mins

Task 1: Adding a New 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 ☐

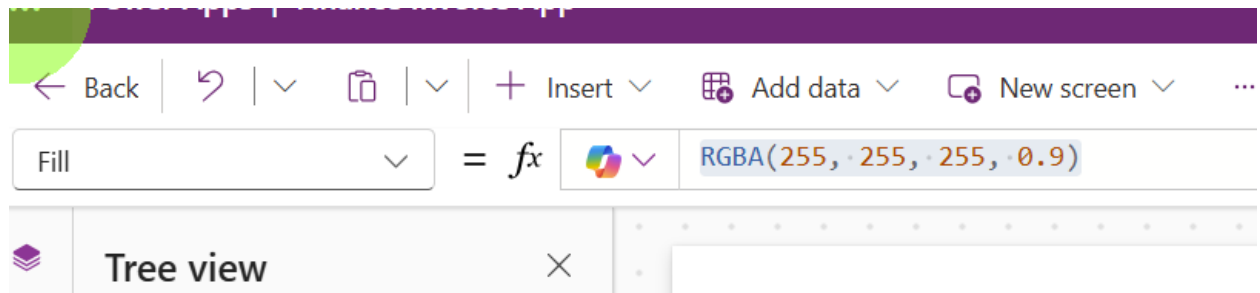
Visible On ☒

Size
Width Height

Fill

Set Fill Property to the component as below.

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



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

CreateCancel

Properties

COMPONENT ⓘ
Component_Popup

DisplayAdvanced

DescriptionNo value

Access app scopeOff

VisibleOn

Size800600
WidthHeight

Fill

Custom properties ⓘ
+ New custom property

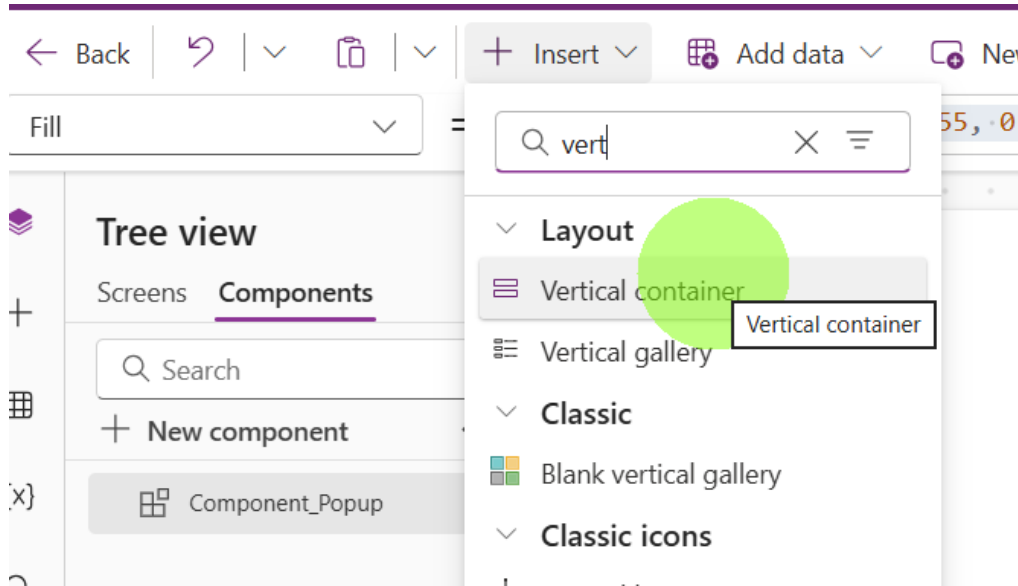
MessageTypeText...

MessageText...

----- 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

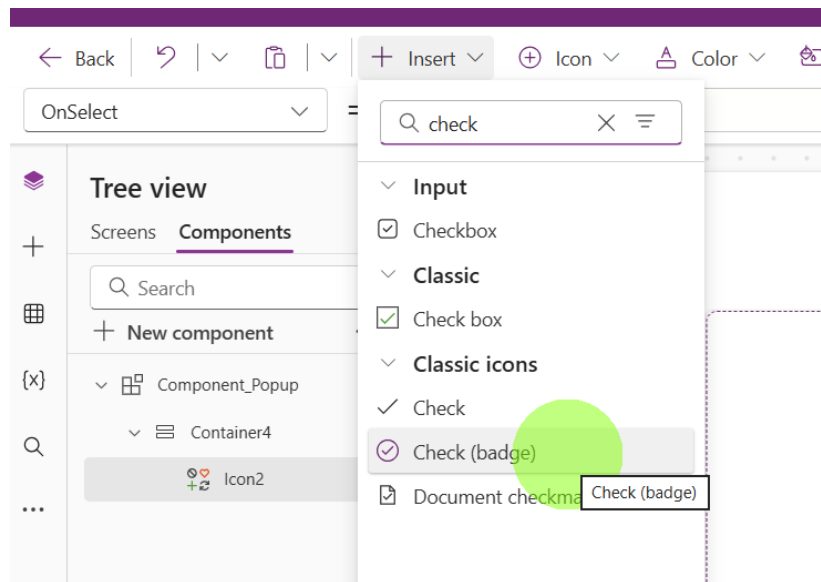
LayoutGap: 30

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

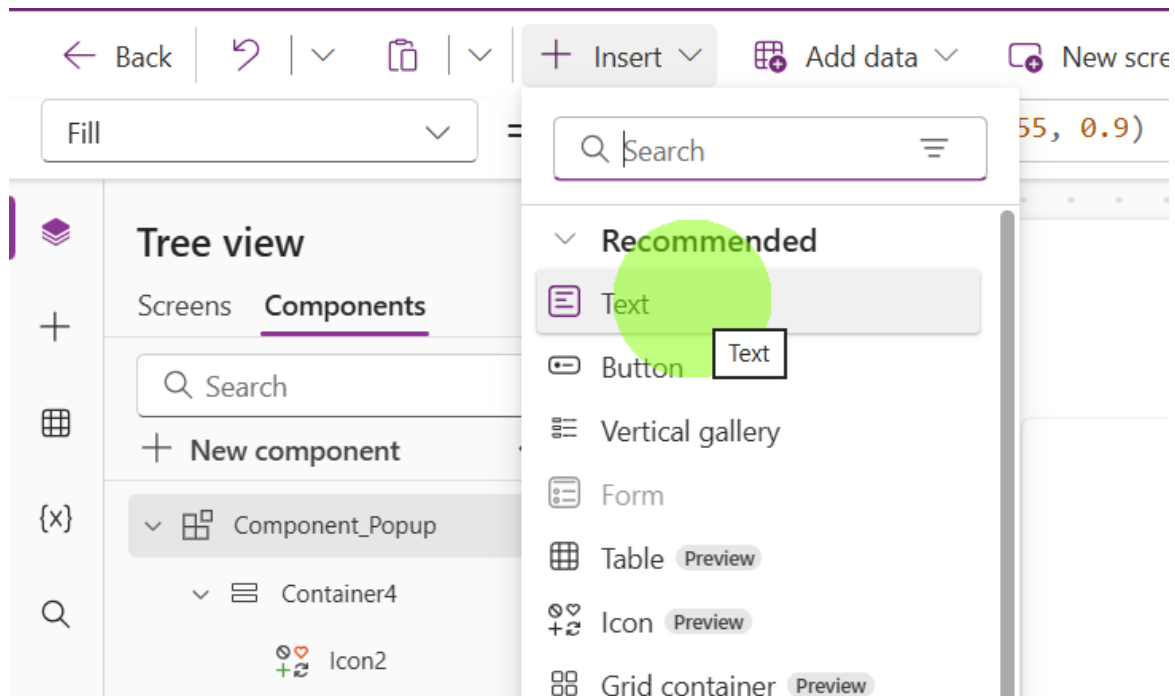
Icon

= 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.

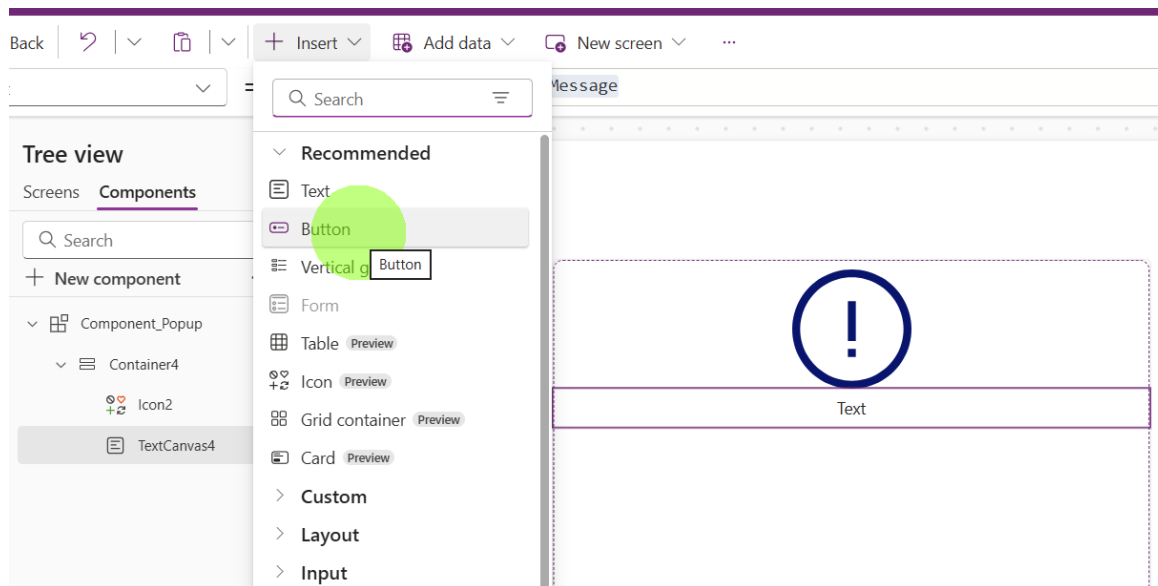
Width: `Parent.Width`

Align: `'TextCanvas.Align'.Center`

VerticalAlign: `VerticalAlign.Middle`

Icon: `Component_Popup.Message`

3. Insert a Button control into the container.

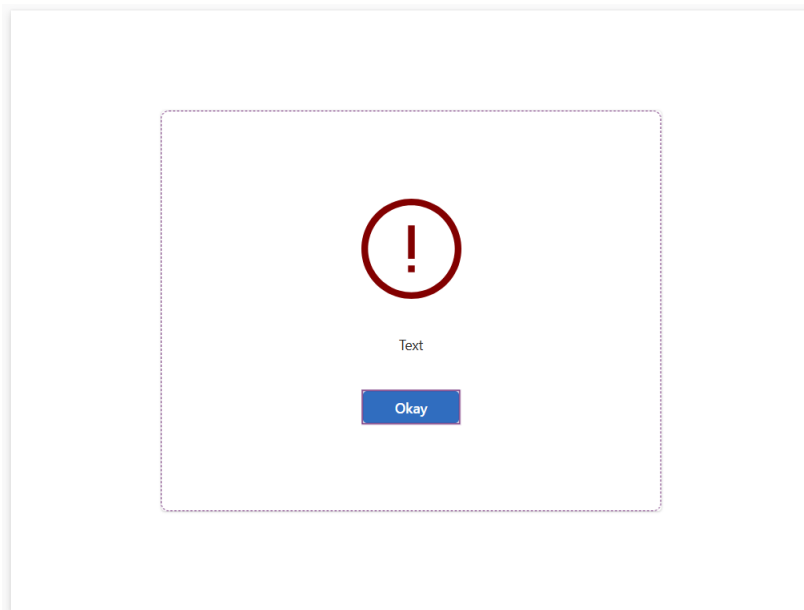


Change below properties of the Button.

Text: "Okay"

OnSelect: `Component_Popup.ClickEvent()`

Your Component should look something like below.



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

Task 3: Adding Popup Component to Submission Screen

2. 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. The top toolbar includes buttons for Back, Undo, Redo, Insert, Add data, Edit fields, Background color, and a menu icon. The left sidebar shows the Tree view with the following structure:

- Screens
 - Screen_Home
 - Screen_Invoices
 - Screen_Submit
 - ScreenContainer2_1
 - HeaderContainer2_1
 - MainContainer2_1
 - ReceiptProcessor1
 - Container3
 - Form1

The main canvas shows a form titled "GLOBAL TECH SOLUTIONS PTE LTD" with the following fields:

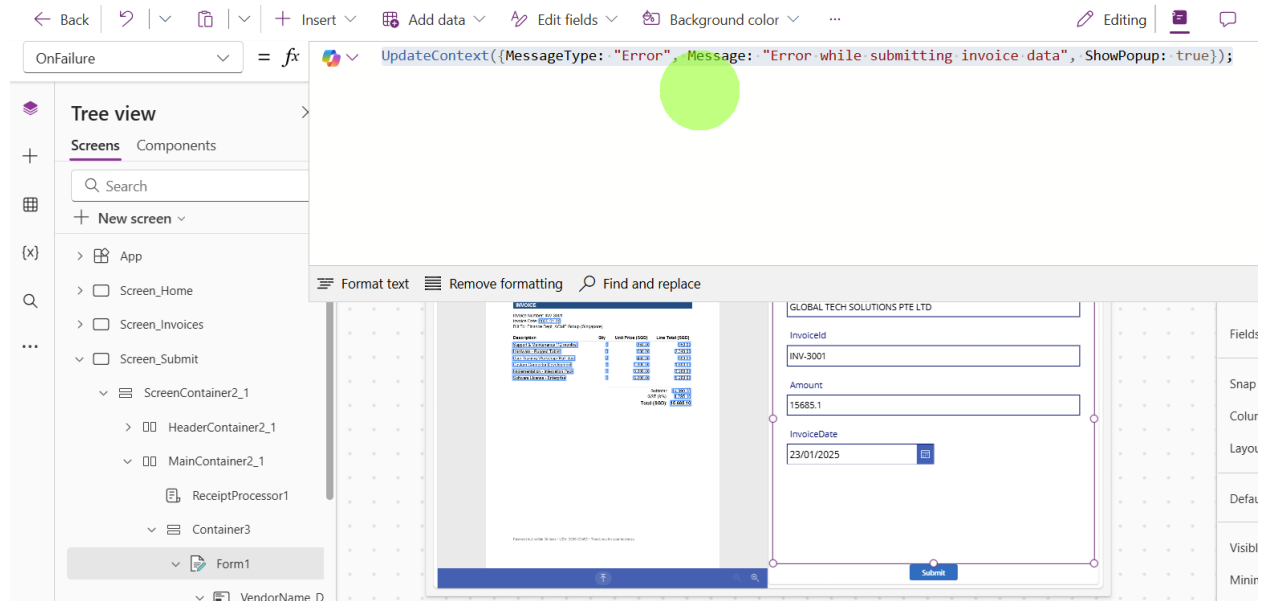
- InvoiceId: INV-3001
- Amount: 15685.1
- InvoiceDate: 23/01/2025
- Submit button

The OnSuccess property for the form control is set to the following code:

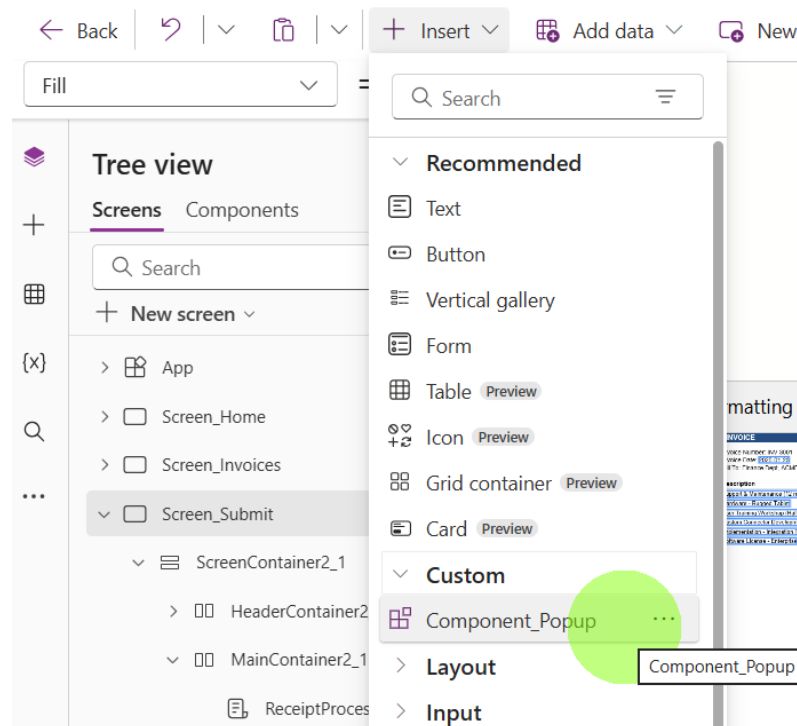
```
UpdateContext({MessageType: "Success", Message: "Invoice data submitted successfully", ShowPopup: true});
```

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`

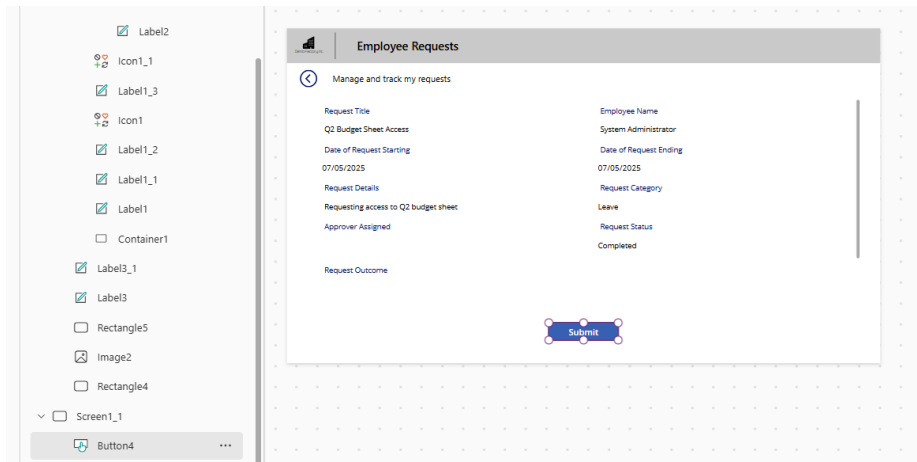
4. Change the text property of the button to the following value

```
If(ThisItem.'Request Status'.Value = "Completed", "View Request", "Edit Request")
```

The screenshot displays a Power Apps interface for managing requests. At the top, there is a navigation bar with a dropdown menu labeled 'All Requests', a blue button labeled '+ New Request', and a sort icon. Below the navigation bar, there are two request cards. The first card is for 'Client Meeting Room Booking' by John Lim, submitted 4 days ago on 05 May 2025, with a status of 'Pending' and an 'Edit Request' button. The second card is for 'Q2 Budget Sheet Access' by Daniel Wong, submitted 4 days ago on 07 May 2025, with a status of 'Approved' and a 'View Request' button.

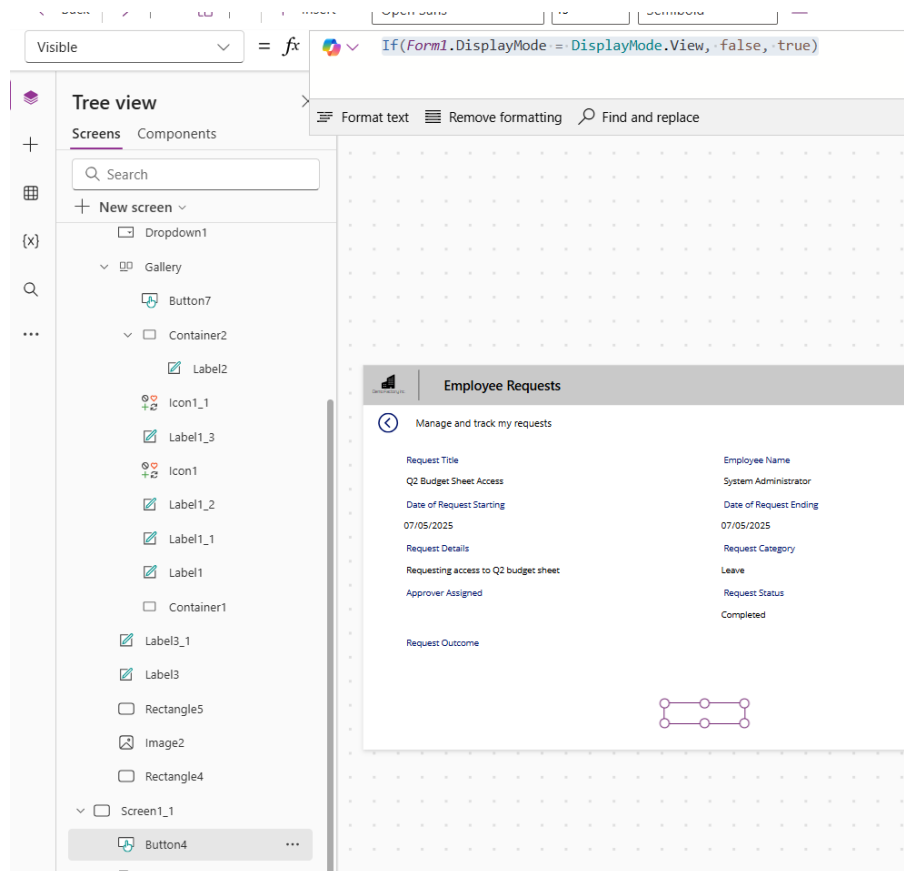
Request Title	Requester	Submitted	Status	Action
Client Meeting Room Booking	John Lim	Submitted 4 days(s) ago 05 May 2025	Pending	Edit Request
Q2 Budget Sheet Access	Daniel Wong	Submitted 4 days(s) ago 07 May 2025	Approved	View Request

5. Select the "submit" button on the Form page.



Change the visible property of the button to the following value

`If(Form1.DisplayMode = DisplayMode.View, false, true)`



End of Task