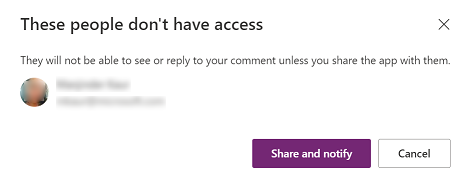
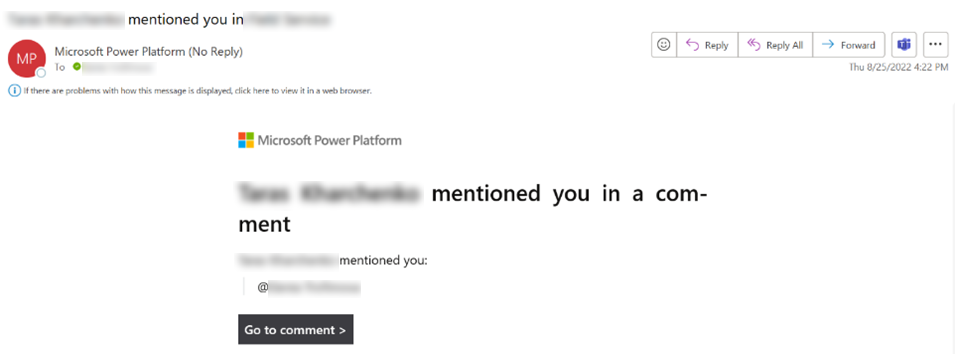
**Adding Comments and Using @Mentions to controls/components**

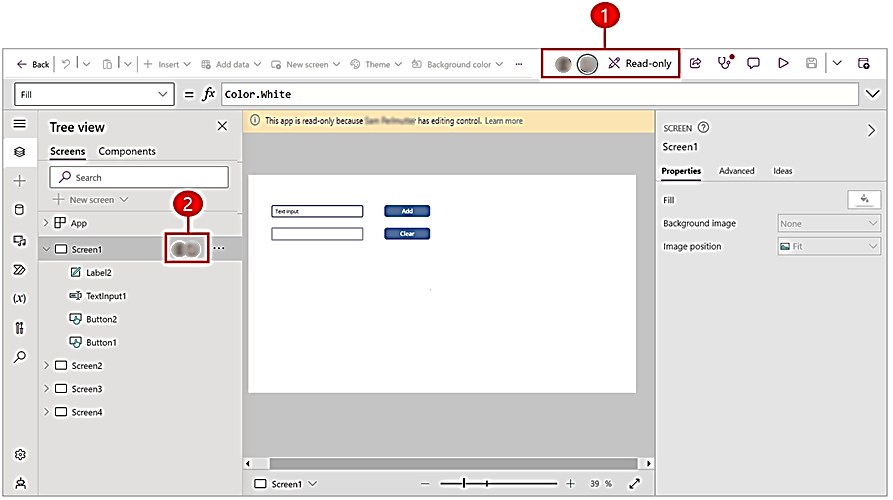
1. Rclick on a control/component🡪new comment🡪try using @mentions
2. If you tag a maker that doesn’t have access to your app, then you’ll be prompted to share the app with them.



1. When someone tags you in a comment, you'll receive an email that lets you know which app you're tagged in and the person who tagged you. It also includes the comment text and a direct link to that comment.



**Copresence**



Legend

1. The command bar displays the names and icons of other makers who are working on the app and may be making changes.
2. The left-hand navigation pane in Power Apps Studio displays the app's structure, indicating which part of the app is being worked on. You receive a notification to refresh the app when the maker that's editing the app makes changes and saves that app.

**How copresence works**

* The first time someone opens your app in Power Apps Studio while you're working on it, copresence indicators appear that shows other people are also working on the app.
* The maker that opens the app first has editing control. If a second user tries to open the app, a notification appears letting them know that someone else is editing the app so they're in read-only mode. If you're in read-only mode, you can save a copy of the app.
* Icons of copresent makers appear in both the command bar and the left navigation pane showing where other makers are working in the app.
* You may be working on the app, or you may be idle, but once someone else saves a change to the app, you receive a notification letting you know that another maker made changes. When you see this notification, consider refreshing the app to get the latest version.

**Adding and navigating screens**

**Add a new screen**

1. In Power Apps Studio, on the command bar, select New screen and then select a screen layout.
2. You can rename the screen afterwards

**Reorder screens**

1. When you have more than one screen, you can reorder them.
2. In the left pane, hover over a screen that you want to move up or down, and then select Move up or Move down.
3. Note: Use the StartScreen property set which screen will be displayed first.

**Add navigation**

1. When you create a canvas app with multiple screens, you can add navigation so your users can navigate between screens.
2. With the screen selected, select Insert and in the search box type Next arrow and then select it.
3. (optional) Move the arrow so that it appears in the lower-right corner of the screen.
4. With the arrow selected, set the OnSelect property to the Navigate function.

Navigate(Target, ScreenTransition.Fade)

1. Try adding next and back arrows or buttons for navigation.

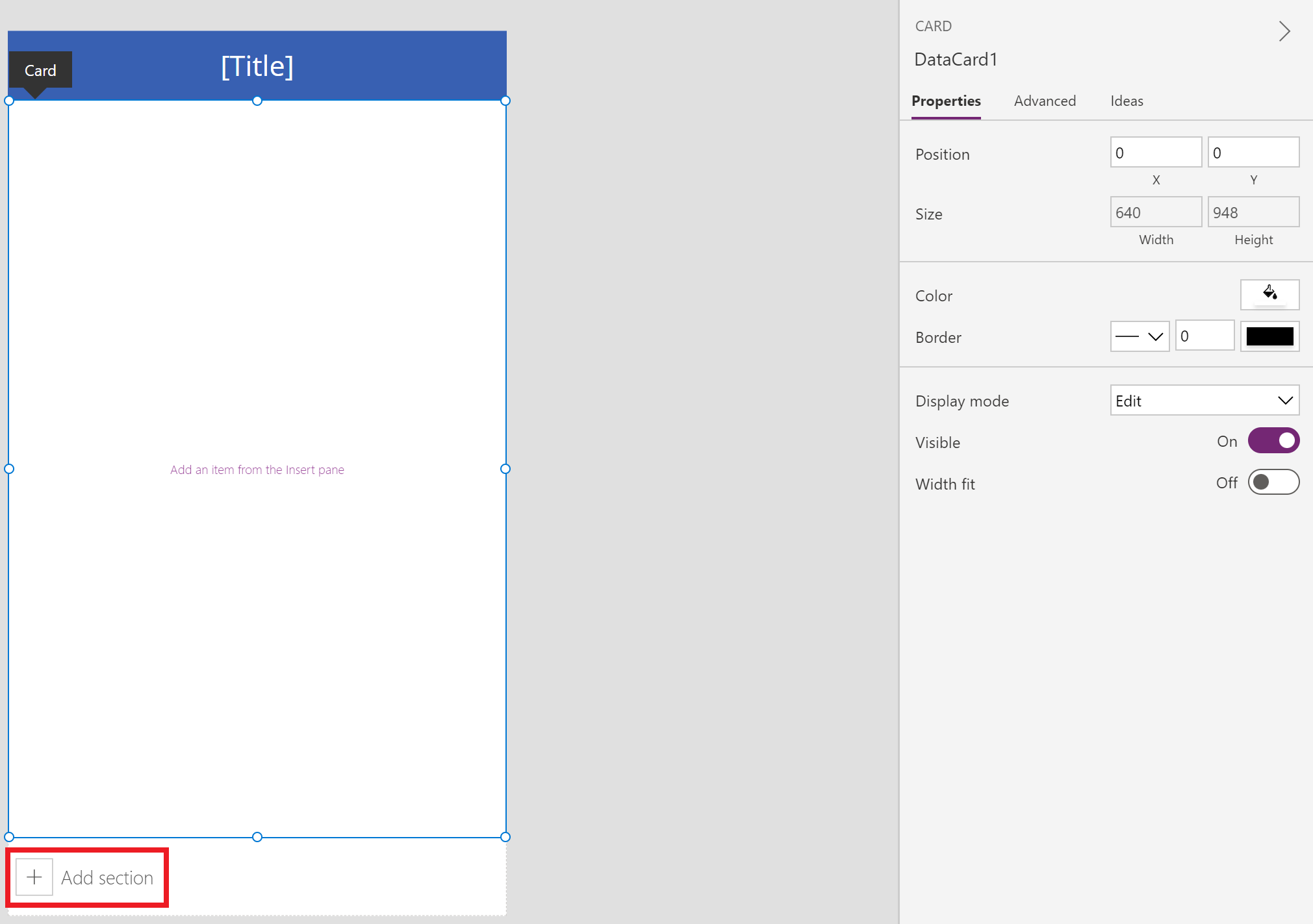
**Add Scrollable Screens**

**Create a scrolling screen**

1. Select New screen below the top menu🡪Select Scrollable.

**Add controls**

1. Scrollable screen includes a fluid grid with one data card by default. Data cards help separate building blocks on the screen. To make screen scrollable with multiple controls, add more data cards. And then, add controls in data cards as required.
2. To add data cards, you can select Add section at the bottom of the scrollable screen.



We'll start by adding controls on the data card available with the scrollable screen by default, and then add a new section that adds another data card. Once a new data card is available, we'll then add another control inside the new data card.

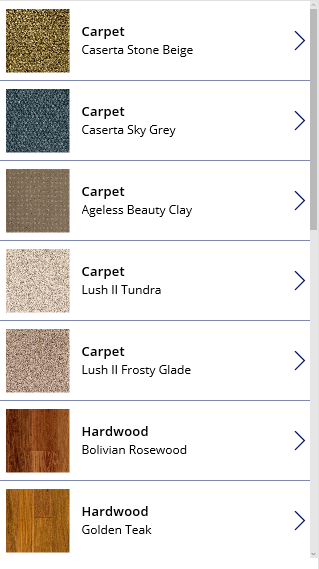
Together, both data cards and the controls within the data cards would extend the default length of the screen, requiring the use of the scrolling ability of the screen.

**Show a list of items in canvas apps**

1. Create a blank canvas app.
2. Add a connection to the FlooringEstimates table in the Excel file.

<https://github.com/johnreygoh/powerapps/blob/master/FlooringEstimates.xlsx>

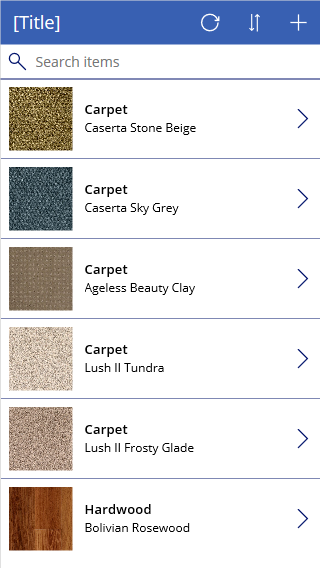
1. On the Insert tab, select Gallery, and then select Vertical.
2. On the Properties tab of the right-hand pane, open the Items list, and then select Flooring Estimates.



1. (optional) In the Layout list, select a different option.

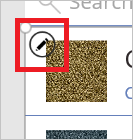
**Add a gallery in a screen**

1. On the Home tab, select New screen > List screen.
2. A screen that contains a Gallery control and other controls, such as a search bar, appears.
3. Set the gallery's Items property to FlooringEstimates.
4. The Gallery control shows the sample data.

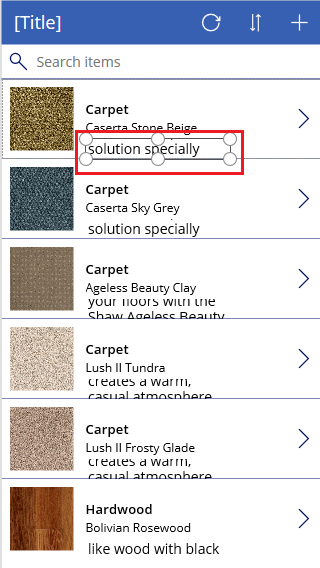


**Add a control to the Gallery control**

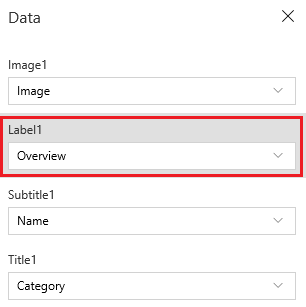
1. Select the template by clicking or tapping near the bottom of the Gallery control and then selecting the pencil icon in its upper-left corner.



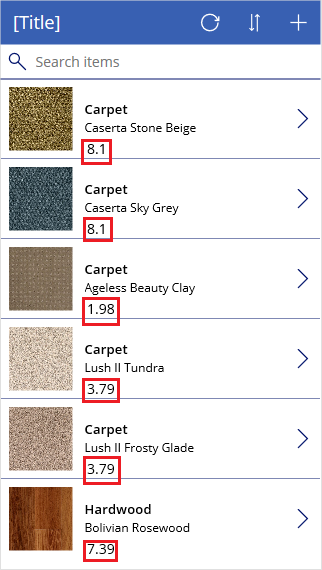
1. With the template still selected, add a Label control, and then move and resize it so that it doesn't overlap with other controls in the template.



1. Select the gallery, and then select Edit next to Fields on the Properties tab of the right-hand pane.
2. Select the label that you added earlier in this procedure, and then open the highlighted list in the Data pane.

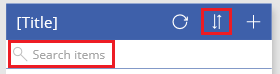


1. In that list, click or tap Price. The Gallery control shows the new values.



**Filter and sort a gallery**

1. The Items property of a Gallery control determines which items it shows. In this procedure, you configure that property so that it also determines which records appear based on filter criteria and in what order.



1. Set the Items property of the Gallery control to this formula:

Sort

(If

(IsBlank(TextSearchBox1.Text),

FlooringEstimates,

Filter(

FlooringEstimates,

TextSearchBox1.Text in Text(Name)

)

),

Name,

If(

SortDescending1,

SortOrder.Descending,

SortOrder.Ascending

)

)

1. Double-click the search box, and then type part or all of a product name in it.
2. Only those items that meet the filter criterion appear.
3. While pressing the Alt key, select the sort icon one or more times to switch the sort order.
4. The records toggle between ascending and descending alphabetical order based on the product name.

**Highlight the selected item**

1. Set the Gallery control's TemplateFill property to a formula that's similar to this example, but you can specify different colors if you want:

If(ThisItem.IsSelected, LightCyan, White)

**Change the default selection**

1. Set the Gallery control's Default property to the record that you want to select by default. For example, you can specify the fifth item in the FlooringEstimates data source:

Last(FirstN(FlooringEstimates, 5))

In this example, you specify the first item in the Hardwood category of the FlooringEstimates data source:

First(Filter(FlooringEstimates, Category = "Hardwood"))

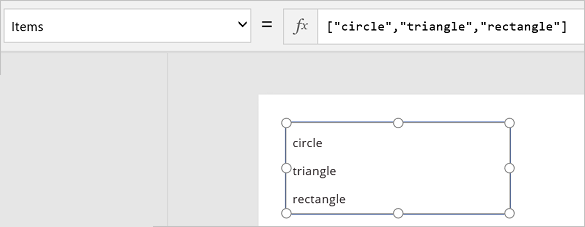
**Add a list box, a drop-down list, a combo box, or radio buttons to a canvas app**

**Create a simple list**

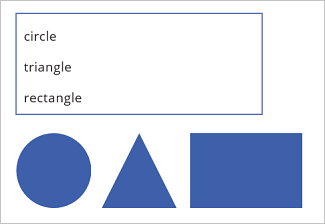
1. Add a List box control named MyListBox, and set its Items property to this expression:

**["circle","triangle","rectangle"]**

1. Your designer looks similar to the following:



1. On the Insert tab, select Icons, select the circle, and move it under MyListBox:
2. Add a triangle and a rectangle, and then arrange the shapes in a row under MyListBox:



1. Set the Visible property of the following shapes to the following functions:

Shape Set Visible function to

circle If("circle" in MyListBox.SelectedItems.Value, true)

triangle If("triangle" in MyListBox.SelectedItems.Value, true)

rectangle If("rectangle" in MyListBox.SelectedItems.Value, true)

**Add radio buttons**

1. On the Home tab, select New Screen, and then select Blank.
2. On the Insert tab, select Controls, and then select Radio.
3. Rename the Radio control to Choices, and set its Items property to this formula:

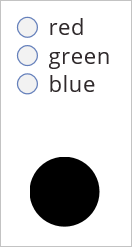
["red","green","blue"]

1. On the Insert tab, select Icons, and then select the circle.
2. Set the Fill property of the circle to the following function:

If(Choices.Selected.Value = "red", Red, Choices.Selected.Value = "green", Green, Choices.Selected.Value = "blue", Blue)

In this formula, the circle changes its color depending on which radio button you choose.

1. Move the circle under the Radio control, as in this example:



**Add an item to an existing list**

1. Add a Button control and name it "btnReset".
2. Set OnSelect property on btnReset to this formula:

ClearCollect(MyItems, {value: "circle"},{value: "triangle"},{value: "rectangle"})

1. Set the Text property on btnReset to "Reset".
2. Add a List box control named lbItems, and set its Items property to MyItems.
3. While holding down the Alt key, press the Reset button.
4. Note: The list box should populate with the items from the "MyItems" collection.
5. Arrange the list box and button so they're lined up vertically.

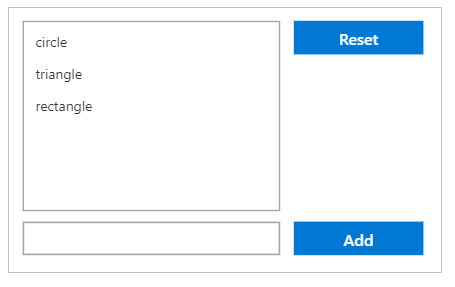


1. Add a Text Input control, and name it "txtAdd".
2. Set Text property of txtAdd to "".
3. Add a Button control, and name it "btnAdd".
4. Set the Text property of btnAdd to "Add".
5. Set OnSelect property of btnAdd to the following formula:

Collect(MyItems,{value: txtAdd.Text}); Reset(txtAdd)

Note:

* The collect function will add the text from the text input as an item in the collection.
* The reset function will reset the text input back to it's default state.
* Arrange txtAdd and btnAdd so they're lined up vertically underneath lbItems and btnReset.

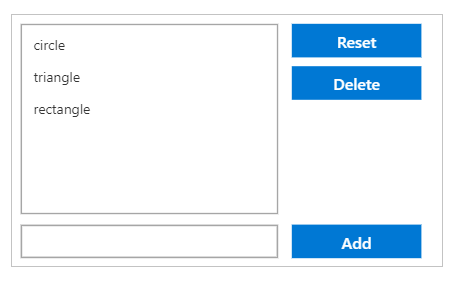


**(Optional) Remove an item from an existing list**

1. Add a Button control, and name it "btnDelete".
2. Set the Text property of btnDelete to"Delete".
3. Set OnSelect property of btnDelete to the following formula:

Remove(MyItems, lbItems.Selected)

1. Arrange btnDelete so it's lined up vertically underneath btnReset



**Activity 1**

Task:

1. Create a Tablet canvas app from blank called "User Canvas".
2. Scale the app so that it fits onto an iPad (4:3).
3. Insert a gallery, and connect that to the Users table - this is a Dataverse table.
4. Resize the gallery so that it fits the left-hand part of the screen.
5. Change the Layout to "Title, subtitle and body", and add the columns/fields "License type" and the "Primary email".
6. Change the font size in the "Primary email" so that the text is more readable.

Solution:

1. Powerapps right panel🡪create🡪blank app🡪canvas app🡪set app name “user canvas” and format to “Tablet”
2. Once editor loaded🡪settings🡪display🡪size = 4:3 ipad
3. Insert🡪vertical gallery🡪use the dataverse Users table

A screenshot of a computer

Description automatically generated

1. In the properties window of the View Gallery, set the layout
2. In the properties window of the View Gallery, click on fields to set the values to bind on Title (Full Name), Subtitle (License Type) and Body (Primary Email)

A screenshot of a computer

Description automatically generated

1. Reduce the font size of the sections of the by then modifying from the properties window

Task:

1. Add a Display Form into our existing screen, which shows the details of the item that we have clicked on in the gallery.

a. Change the number of columns to 1.

b. Add the "Primary Email" and "License Type" fields.

1. Add an icon at the top which, when clicked, launches into a new Edit screen (you will need a Edit Form).

a. Add the "Primary Email" and "License Type" fields.

b. Change the number of columns to 1, and the Layout to Horizontal.

* You will need to use the Navigate function.
* But where should you use the Navigate function? Have a look at the Icon's dropdown list of properties for a clue.
* And should you rename the forms?

1. Add working Submit and Cancel icons to the top of the screen.

* You will need to use the SubmitForm, ResetForm, and Back functions.
* But where should you use the Back function?
* Have a look at the Form's dropdown list of properties for a clue.

1. Add some rectangles at the top of our screen. Add titles to our screens, and a working Refresh icon in the Gallery.

* You will need to use the Refresh function.

Solution:

1. Insert🡪display form, and set it to the right half of the screen
2. Display form properties window🡪basic🡪
   1. Data source: Users
   2. Column: 1
   3. Fields: Full Name, License Type, Primary Email, Contact Number, etc…
3. Display form properties window🡪advanced🡪item: **Gallery1.Selected**
4. Lower the height of both the gallery and displayform so we can insert an edit icon on upper right

A screenshot of a computer

Description automatically generated

1. Add an edit screen using the tree view and insert an edit form to the new screen
2. Editform properties window🡪advanced🡪item: **Gallery1.Selected**
3. Editform properties window🡪basic🡪

fields (set fields that can be edited)

columns: 1

layout: horizontal

1. Rename screens (DetailScreen and EditScreen)
2. Back to screen 1🡪click on the edit icon🡪OnSelect:

Navigate(EditScreen)

1. Rename edit form (“Form1”) to “editform”
2. On the editscreen🡪add a submit icon🡪click the icon🡪OnSelect:

SubmitForm(editform);Back()

1. Or, On the editscreen🡪add a submit icon🡪click the icon🡪OnSelect:

SubmitForm(editform)

Click the editform🡪OnSuccess:

Back()

1. On the editscreen🡪add a cancel icon🡪click on the cancel icon🡪OnSelect:

ResetForm(editform);Back()

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

1. On the edit screen, add a rectangle shape on the top part to contain the submit and cancel icons, change the rectangle color, rclick rectangle 🡪reorder🡪send to back
2. Add a text label “Edit User Details” on the rectangle by inserting text label
3. On the detail screen, add a rectangle shapes on the top part to act as title banners, change the rectangle color, rclick rectangle 🡪reorder🡪send to back
4. Add text labels “All Users” and “Selected User” on the rectangle by inserting text label
5. Add a reload icon🡪click the reload icon to reload the datasource🡪OnSelect:

Refresh(Users)

A screenshot of a computer

Description automatically generated A screenshot of a computer

Description automatically generated

**Sorting and filtering data in a canvas app gallery**

**Sorting Gallery using sort icon/button**

1. Click on app🡪OnStart:

Set(varColSort,"fullname");

Set(varSortDir, SortOrder.Descending);

To use the variables now, click the ellipsis […] beside “app” 🡪 run OnStart

1. Add a sort icon/button🡪OnSelect:

Set(varColSort,"fullname");

Set(

varSortDir,

If(varSortDir=SortOrder.Descending,

SortOrder.Ascending,

SortOrder.Descending)

);

1. Display gallery🡪Item:

SortByColumns(Users,varColSort,varSortDir)

**Adding Search/Filter Controls**

1. Add a text input to be used for searching and filtering
2. In the “app”🡪OnStart:

Set(varColSort,"fullname");

Set(varSortDir, SortOrder.Descending);

Set(varSearch, "");

To use the variables now, click the ellipsis […] beside “app” 🡪 run OnStart

1. In the gallery🡪items:

SortByColumns(

Filter(Users,TextInput1.Text in 'Full Name'),

varColSort,

varSortDir

)

**Adding a custom pop-up messagebox**

1. On the editscreen, add a rectangle + cancel icon + text label, highlight the three then click on the “group” option on the menu, set “visibility” to false
2. Click on the group🡪visibility:

pop1

1. Group1 cancel icon🡪OnSelect:

UpdateContext({pop1:false});

Back();

1. In the editform🡪OnSuccess:

UpdateContext({pop1:true});

**Removing Selected Item in Gallery**

1. You can add a button/icon to perform this action🡪OnSelect:

Remove( Contacts, Gallery1.Selected )

**Adding a slider filter control (Optional)**

1. Add a Slider control (Insert tab > Controls), rename it to StockFilter, and move it under the gallery.
2. Configure the slider so that users can't set it to a value outside the range of units in stock:
   1. On the Content tab, select Min, and then enter the following expression:

Min(Inventory, UnitsInStock)

* 1. On the Content tab, select Max, and then enter the following expression:

Max(Inventory, UnitsInStock)

1. Select any item in the gallery except the first one. Set the Items property of the gallery to the following expression:

Filter(Inventory, UnitsInStock<=StockFilter.Value)

1. In Preview, adjust the slider to a value that's between the highest and the lowest quantity in the gallery. As you adjust the slider, the gallery shows only those products that are less than the value you choose:
2. Let us now implement both name search and filter. Go back to the designer.
3. On the Insert tab, select Text, select Input Text, and rename the new control to NameFilter. Move the text control below the slider.
4. Set the Items property of the gallery to the following expression:

Filter(Inventory, UnitsInStock<=StockFilter.Value && NameFilter.Text in ProductName)

1. Example: In Preview, set the slider to 30, and type the letter g in the text-input control. The gallery shows the only product with less than 30 units in stock and has a name with the letter "g":

**Show data in a line, pie, or bar chart in canvas apps**

1. Create your own sample data using the example above and save it in Excel.

For example, your data should look similar to the following:

Product Revenue2012 Revenue2013 Revenue2014

Europa 21000 26000 28000

Ganymede 15000 17000 21000

Callisto 14000 19000 23000

**Import the sample data**

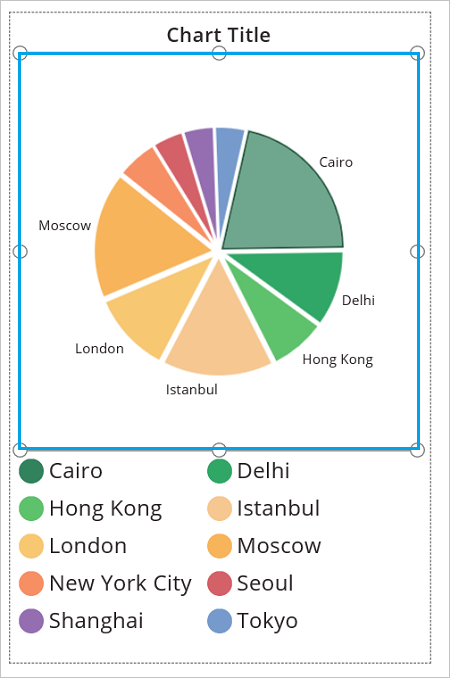
1. In these steps, we import the sample data into a collection, named ProductRevenue.
2. On the command bar select, Insert > Media > Import.
3. Set the control's OnSelect property to the following function:

Collect(ProductRevenue, Import1.Data)

1. On the app actions menu, select Preview the app and then select the Import Data button.
2. In the Open dialog box, select your Excel file, select Open, and then press Esc.
3. On the app authoring menu (menu on the left) select, Variables > Collections.
4. The ProductRevenue collection should be listed with the chart data you imported.
5. Press Esc to return to the default workspace.

**Add a pie chart**

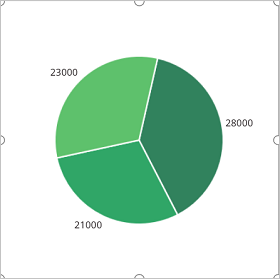
1. On the command bar selelct, Insert > Charts > Pie Chart.
2. Move the pie chart under the Import data button.
3. In the pie-chart control, select the middle of the pie chart:



1. Set the Items property of the pie chart to this expression:

ProductRevenue.Revenue2014

1. The pie chart shows the revenue data from 2014.

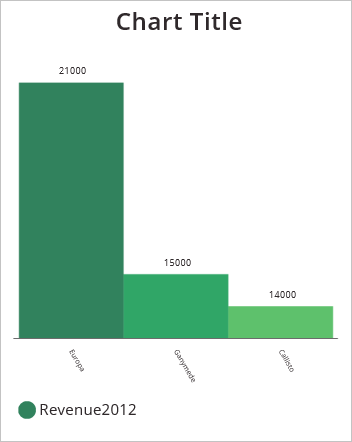


**Add a bar chart to display your data**

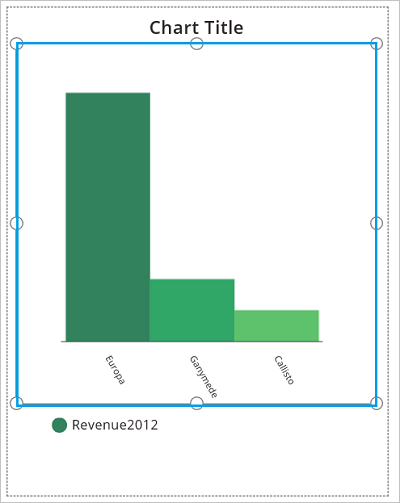
1. On the command bar, select New screen > Blank.
2. On the command bar, select Insert > Tree view > Column Chart.
3. Select the middle of the column chart. Set the Items property of the column chart to ProductRevenue:

Items property to ProductRevenue

1. The column chart shows the revenue data from 2012:



1. In the column chart, select the center square:



1. On the Chart tab, select Number of Series, and then enter 3 in the formula bar:
2. The column chart shows revenue data for each product over three years:

