Using Advanced PowerApps Design Features

Lab Time: 60 minutes

Lab Folder: C:\Student\Modules\02_DesigningPowerApps\Lab

Lab Overview: In this lab, you will build a device ordering application using PowerApps Studio for the Web which will run in most browsers including Microsoft Edge, Internet Explorer 9+, Google Chrome, Firefox or Safari.

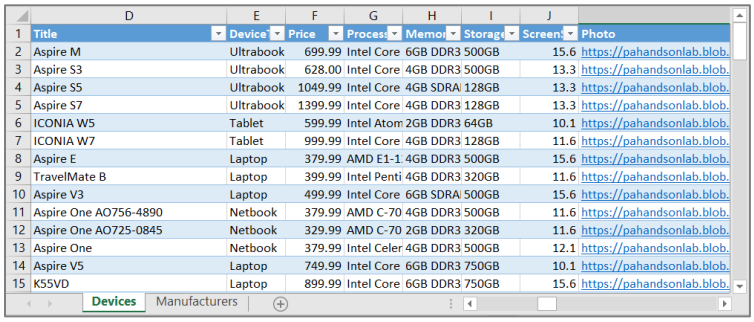
Exercise 1: Create a New App using the Start From Blank Template

In this exercise, you will use the PowerApps Studio to create an app for comparing and selecting devices such as laptop computers. You will begin by creating a new app using the **Start from blank** app template and working with a pre-provided Excel workbook file named **Device-Order-Data.xlsx** which contains two tables with the data for devices and manufacturers.

1. Examine the Excel workbook named **Device-Order-Data.xlsx**.
   a) Using Windows Explorer, verify that you have an Excel workbook named **Device-Order-Data.xlsx** at the following path.

   ```
   C:\Student\Modules\02_DesigningPowerApps\Lab\Device-Order-Data.xlsx
   ```

   b) If you have Microsoft Excel installed, open **Device-Order-Data.xlsx** and examine the two tables inside.



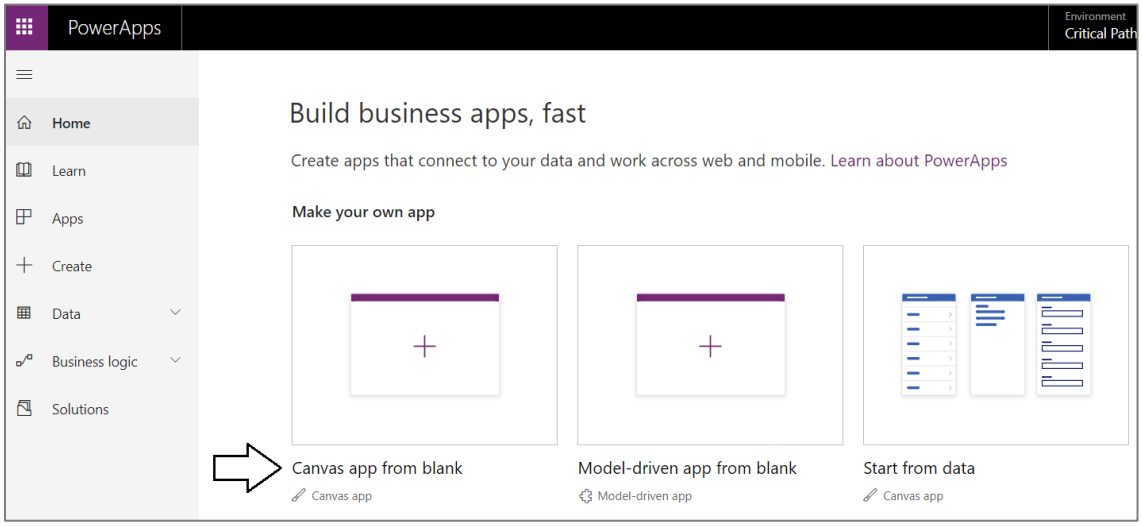
	Title	Device	Price	Process	Memor	Storag	Screen	Photo
2	Aspire M	Ultrabook	699.99	Intel Core	6GB DDR3	500GB	15.6	https://pahandsonlab.blob.
3	Aspire S3	Ultrabook	628.00	Intel Core	4GB DDR3	500GB	13.3	https://pahandsonlab.blob.
4	Aspire S5	Ultrabook	1049.99	Intel Core	4GB SDRA	128GB	13.3	https://pahandsonlab.blob.
5	Aspire S7	Ultrabook	1399.99	Intel Core	4GB DDR3	128GB	13.3	https://pahandsonlab.blob.
6	ICONIA W5	Tablet	599.99	Intel Atom	2GB DDR3	64GB	10.1	https://pahandsonlab.blob.
7	ICONIA W7	Tablet	999.99	Intel Core	4GB DDR3	128GB	11.6	https://pahandsonlab.blob.
8	Aspire E	Laptop	379.99	AMD E1-1	4GB DDR3	500GB	15.6	https://pahandsonlab.blob.
9	TravelMate B	Laptop	399.99	Intel Penti	4GB DDR3	320GB	11.6	https://pahandsonlab.blob.
10	Aspire V3	Laptop	499.99	Intel Core	6GB SDRA	500GB	15.6	https://pahandsonlab.blob.
11	Aspire One AO756-4890	Netbook	379.99	AMD C-70	4GB DDR3	500GB	11.6	https://pahandsonlab.blob.
12	Aspire One AO725-0845	Netbook	329.99	AMD C-70	2GB DDR3	320GB	11.6	https://pahandsonlab.blob.
13	Aspire One	Netbook	379.99	Intel Celer	4GB DDR3	500GB	12.1	https://pahandsonlab.blob.
14	Aspire V5	Laptop	749.99	Intel Core	6GB DDR3	750GB	10.1	https://pahandsonlab.blob.
15	K55VD	Laptop	899.99	Intel Core	6GB DDR3	750GB	15.6	https://pahandsonlab.blob.

Devices | Manufacturers

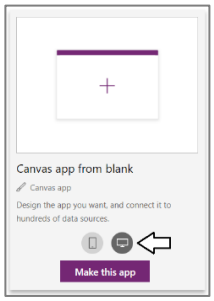
   c) After you have examined the contents of **Device-Order-Data.xlsx**, close Microsoft Excel and do not save any changes.

2. Sign into the PowerApps portal.
   a) Navigate to https://web.powerapps.com and sign-in with your Office 365 user account.

3. Create a new app with a tablet layout using the **Start from blank** app template.
   a) Hover the mouse over the **Start from blank** template.



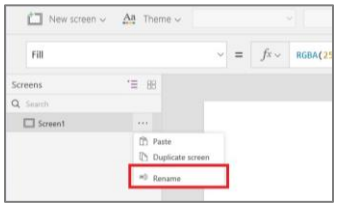
b) Make sure the Tablet/Desktop icon is selected and then click **Make this app**.



c) Wait while PowerApps Studio creates the new app.

When using the **Canvas app from blank** template, PowerApps Studio creates an app with a screen named **Screen1** with no controls.

4. Rename **Screen1** to **MainScreen**.
   a) Select the screen by clicking the **Screen1** tile in the left navigation bar.
   b) Click the ellipse (**…**) menu next to **Screen1** and select **Rename**.



   c) Change the name to **MainScreen**.

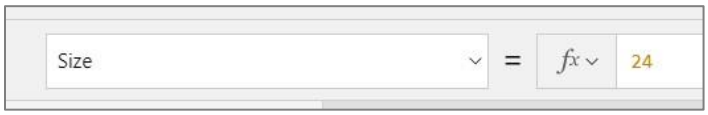
Note that you can also rename the screen by clicking on the screen name in the right pane and selecting the edit (pencil) icon.

5. Add a header across the top of your app containing the app name and logged in user's name.
   a) Select the **Insert** tab in the ribbon, and select **Label** to add a text label.
   b) With your new label selected:
   c) Rename the label from Label1 to **HeaderLabel.**
      i) To do this, click … next to the Label1 control in the tree view and select **Rename**.
   d) Select **Text** from the property dropdown list and enter **"Device Ordering App"** in the formula bar.
      i) You can also type directly in the label control.
   e) Resize the label such that the width is the width of the screen and the height is a reasonable size for a header.
   f) You can set a specific value for the height, e.g. 80 pixels, by using the Properties pane on the right of the screen.

   g) Select the **Home** tab and set the **Align** option in the ribbon to **Align center**.
   h) Set the label fill color to blue and the font color to white. In the **Home** menu, use the following command buttons to set the colors. Or you can also use the Properties pane on the right

i) Similarly, set the **font size** to 24 from the Home ribbon.

i) You can also achieve this by modifying the **Size** property of this label control.



6. In the header, display the logged in user's name.

a) Insert a new label control on the right side of the header label.

b) Rename the label to **UserGreeting**.

c) Set the label **Color** property to white.

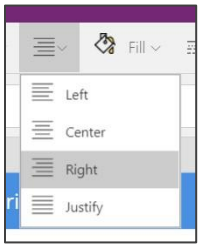
d) Change the label **Text** property to:

```
"Hello, " & User().FullName
```

e) The formula you add should match the one shown in the following screenshot.

Note: All functions in PowerApps are case sensitive. As you start typing "User" you will see a dropdown of available choices. It's a good idea to pick from the autocomplete options. You'll also notice help text at the top showing the required parameters, in this case the **User()** function requires no input parameters.

f) You can right-justify the text in the label by clicking the **Home** tab on the ribbon and using the **Text Alignment** dropdown menu (*which is not the same as the **Align** button*) as shown in the following screenshot.

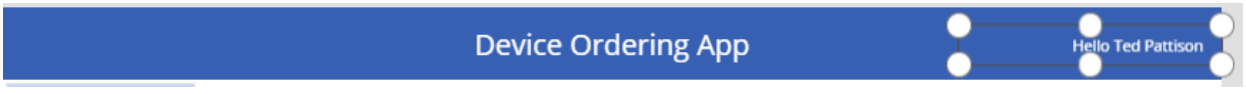


g) If necessary, widen the label so the text doesn't wrap.

h) To add some padding between the text and the edge of the screen, change the **PaddingRight** property from 5 to 20.

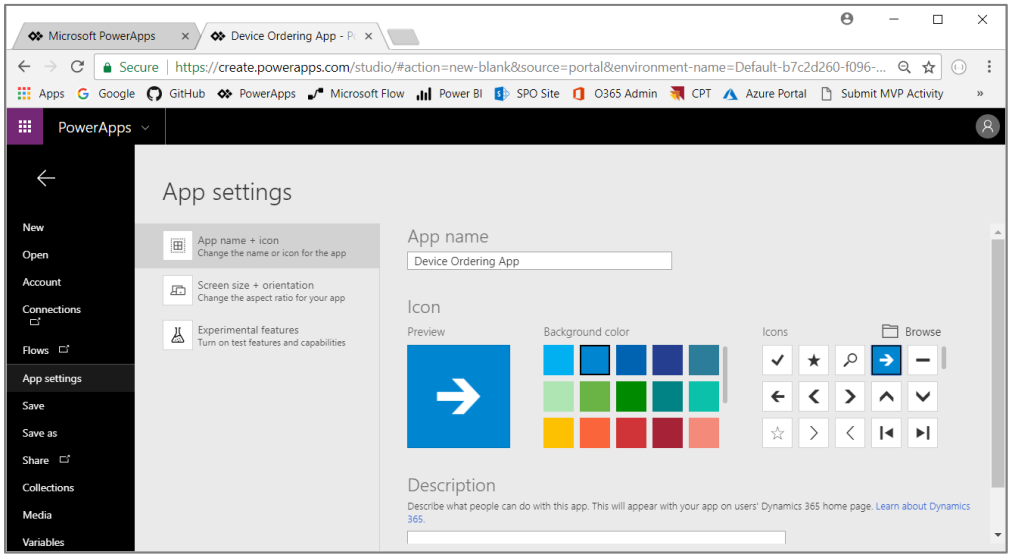
i) You can do this quickly using the **Properties** pane on the right side of the PowerApps Studio window.

i) Your label with the user's display name should look something like the following screenshot.



The **User()** function in PowerApps allows you to retrieve the Email, Full Name and Picture for the currently logged in user. App users will always be logged in with their business or school account (Azure Active Directory (AAD) credentials), so this information will always be available for any apps you create using PowerApps.

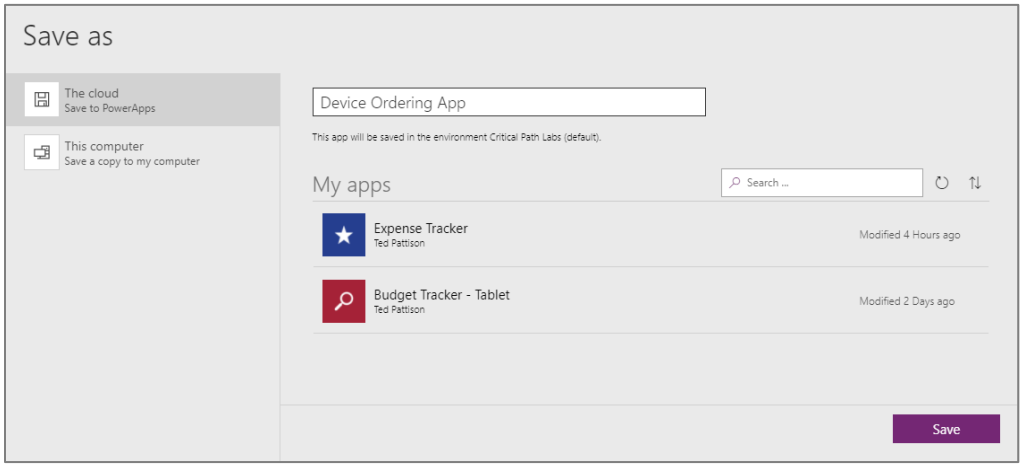
7. Save the app to the cloud for the first time.
   a) Click on the **File** option in the ribbon menu and go to the **App settings** page.
   b) Give your app a name, such as **Device Ordering App**.
   c) Customize the app icon – choose a background color and icon.



   d) Select **Screen size + orientation** to view the available screen orientation and aspect ratio settings.

For this app, we'll leave it at the default setting of Landscape with 16:9 aspect ratio.

   e) Once you've named your app, select the **Save** option on the left pane.
   f) Make sure the option to save the app to **The cloud** is selected, so it's available instantly to run on any device.
   g) Click **Save** in the bottom right.



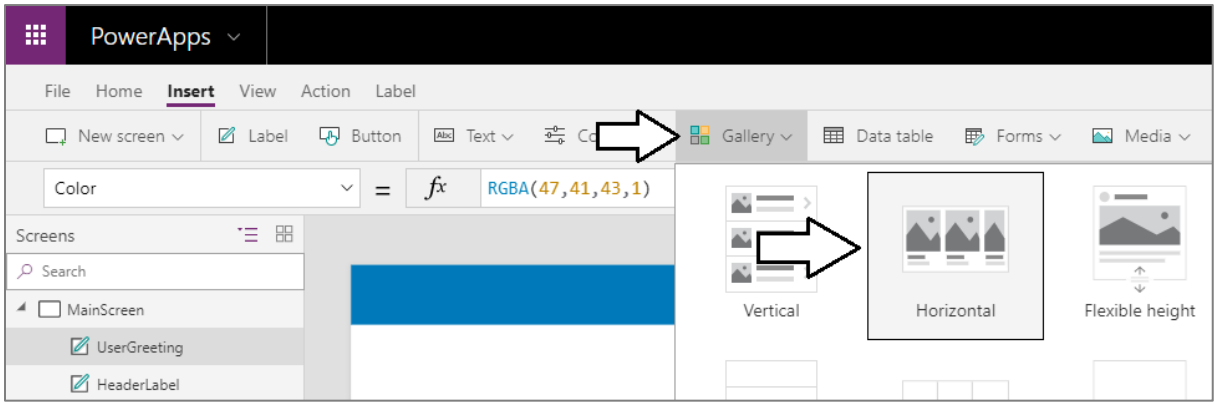
Tip: In PowerApps when you save a version of your app, the changes are only visible to the app maker. You must explicitly publish the app for all app users to get an update. For more details on saving, publishing and sharing apps, see:

https://powerapps.microsoft.com/en-us/tutorials/save-publish-app/
https://powerapps.microsoft.com/en-us/tutorials/share-app/
https://powerapps.microsoft.com/en-us/blog/saveandpublish/

Exercise 2: Add Galleries and Configure Data Binding

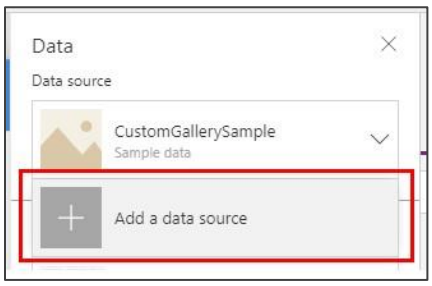
Over the next few steps, you'll add a gallery of all the available devices making it easy for users to browse the list and get a quick overview of the devices available.

1. Create a gallery that will show all available devices.
   a) Select the **Insert** tab in the ribbon.
   b) Select **Gallery** to see a list of gallery templates.
   c) Select the **Horizontal** gallery.

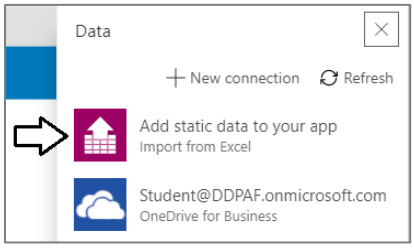


This will add a gallery called **Gallery1** onto the screen. Notice the control tree view on the left displays this gallery with three controls within it – two labels and an image. When you add a new gallery, the Data pane will automatically pop up on the right.

2. Select the data source for the gallery control.
   a) In the **Data** pane on the right, click the **Data source** dropdown and select **+ Add a data source**.



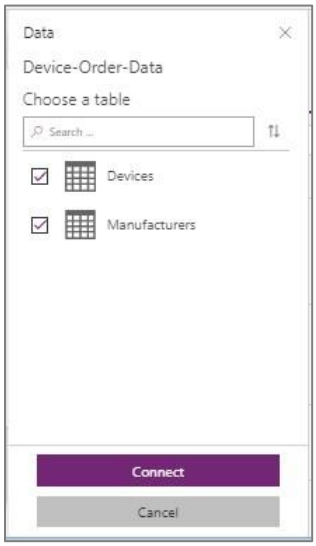
   b) Choose the **Add static data to your app (Import from Excel)** option.



   c) In the File Open dialogue, browse to the location to select **Device-Order-Data.xlsx**.

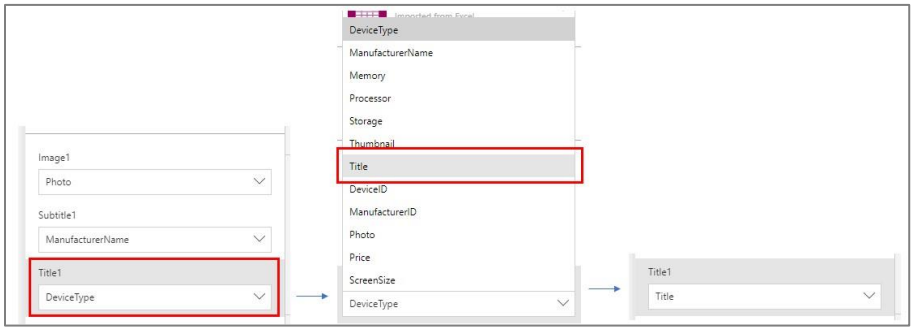
```
C:\Student\Modules\02_DesigningPowerApps\Lab\Device-Order-Data.xlsx
```

d) Select both tables, **Devices** and **Manufacturers**, and click the **Connect** button.



In this lab, you will work with two tables imported from an Excel workbook that are embedded within the app itself as a static resource. In a real-world solution, the same data would likely be stored in a SharePoint list, a SQL Server table or a CDS for Apps entity.

3. The first table named **Devices** will automatically be selected as the source for this gallery.
4. Update the field mappings
   a) The system by default will bind each control within the gallery to some data field in the table.
   b) Click on the first text label in the gallery, just below the image. This control is named **Title1**. Use the dropdown menu in the **Data** pane to change the field binding from **DeviceType** to **Title** which is the name of the device.



   c) Notice that the gallery UI updates as soon as you do this to display the device name below the image
   d) Also notice that the formula bar shows that **Text** property for this **Title1** control is set to **`ThisItem.Title`** which provides another way to change the binding by directly typing into the formula bar.

   e) Notice that **Image1**, which is the image control, is correctly bound to the **Photo** field.

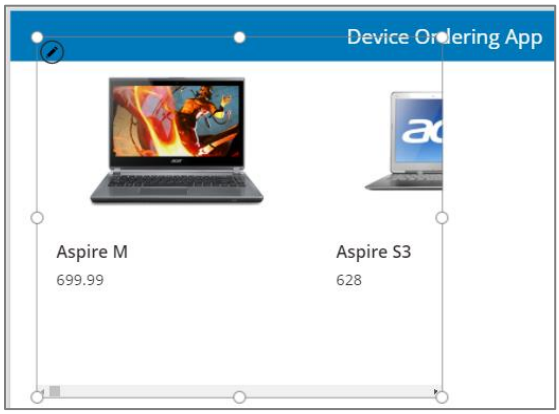
If for some reason it does not show the image, change it to bind to the Photo field.

   f) Similarly, change the binding for the **Subtitle1** label from **ManufacturerName** to **Price**.

g) Close the **Data** pane on the right.

To get back to the **Data** pane, select the gallery control using the tree view on the left and in the **Properties** pane on the right click either **Data** or **Layout**.

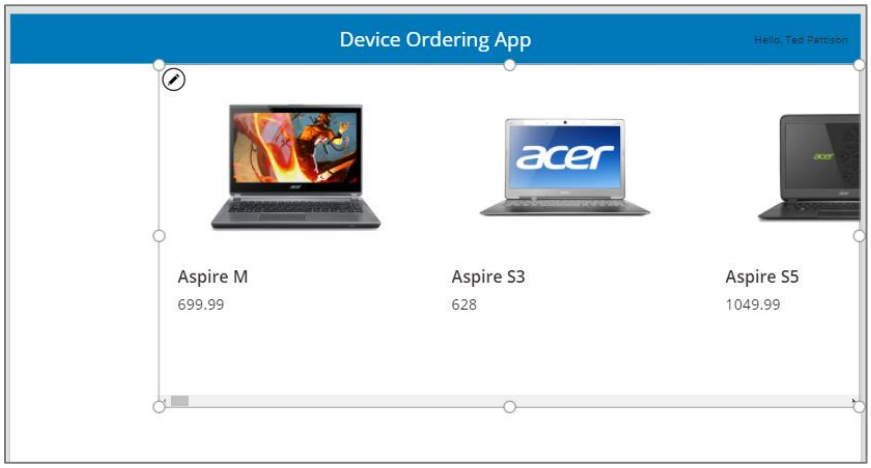
h) Your gallery should match the following screenshot.



5. Position and resize the gallery.

a) **Position the gallery** to the middle and right of the screen just below the header label, such that there is space on the left to add a vertical gallery and space on the bottom for action buttons.

b) Use the drag handles to resize the gallery so that it covers approximately 80% of the width of the screen. See image below.



6. **Rename** the gallery to **DeviceGallery**. There are two ways to rename a control.

a) In the **tree view on the left**, click the three dots (…) next to the control and select **Rename**, or

b) In the **right pane**, click the **edit icon** (pencil) next to the name of the control.

Galleries are an essential aspect of UI construction because they provide a powerful way to visualize tabular data in PowerApps. It's important to become familiar with customizing a gallery. Key components of a gallery are: the gallery control, the template cell, and controls within the template cell.

To select the **entire gallery** – click on the gallery control in the tree view on the left pane, or click on the second or third cell of the gallery. Clicking any cell that is not the first cell of the gallery will select the entire gallery. Now you can configure properties that apply to the entire gallery, such as the **Items** property which is the data source, the gallery fill color, borders, etc.

To customize how each item is displayed in the gallery, you will customize the **template cell**. Select the template by clicking in the first cell of the gallery or clicking on the pencil icon in the top left corner when the entire gallery is selected.

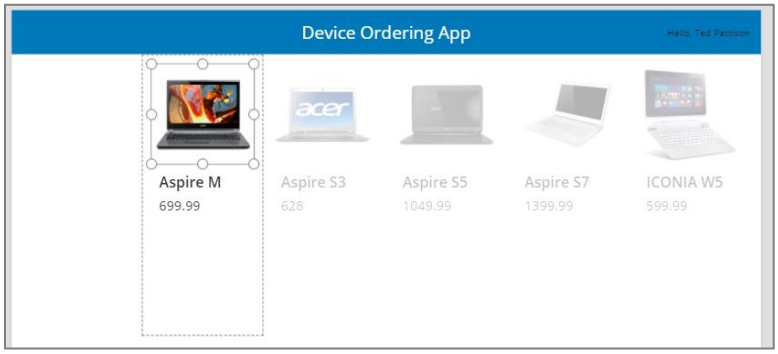
7. Arrange the device gallery.

   a) Select the **DeviceGallery** control and click the **Edit (pencil) icon** in the top left of the gallery to edit the template cell.

   b) Using the right drag control, resize the first box to be narrower. Notice that all the items get narrower and more devices are visible on the screen.

   c) Narrow the image as well by **clicking on the image control and resizing it using the drag handles**.

Make sure the width and height of the image control are positioned within the template.



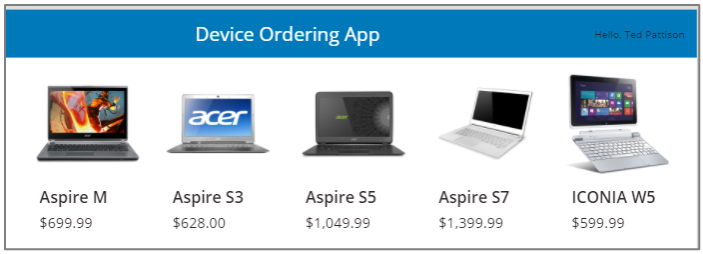
8. Format the **Price** field to display values in the format of **$#,###.00**.

   a) Select the second label, named **Subtitle1** and modify its **Text** property using the following formula.

```
Text(ThisItem.Price,"$#,###.00")
```

   b) Here is what your formula bar should look like.

   c) The currency formatting should now appear in the app.



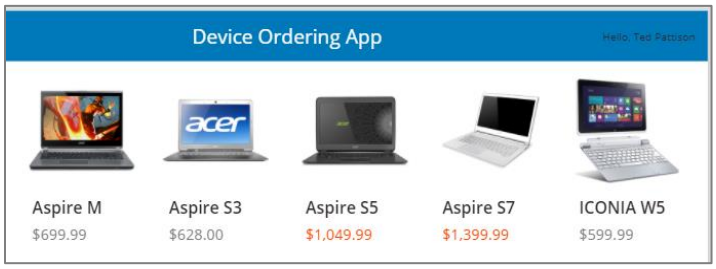
Note: After you enter the above value in the formula bar, notice that it will automatically resolve to including your locale, e.g. [$-en-US]. If you see an error here, it might be because your locale is not yet supported, in which case as a workaround, you can manually change it to [$-en-US].

9. Format the **Price** field to show devices above $1,000 in a different color
   a) Let's say we want to make it easy to spot devices that cost more than $1,000, by displaying the price in red.
   b) To do this, select the label in the template cell that displays the price
   c) Set the **Color** property using the following expression:

```
If(ThisItem.Price>1000,OrangeRed,Gray)
```

   d) The formula you add should match the following screenshot.

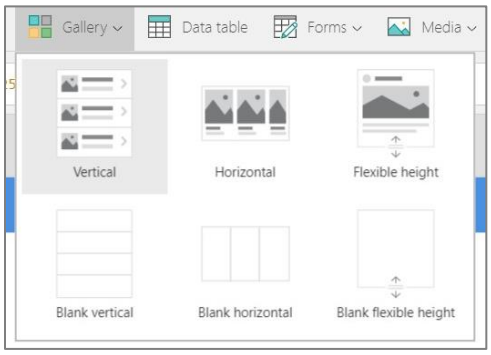
   e) The app should now show larger values in a red font.



Note: As you're typing this formula notice that the autosuggest shows a choice of matching colors. PowerApps comes with a set of standard colors that you can reference in any property that accepts a color value. You can also set specific RGB values. For a full list of Color functions and colors, see https://powerapps.microsoft.com/en-us/tutorials/function-colors/

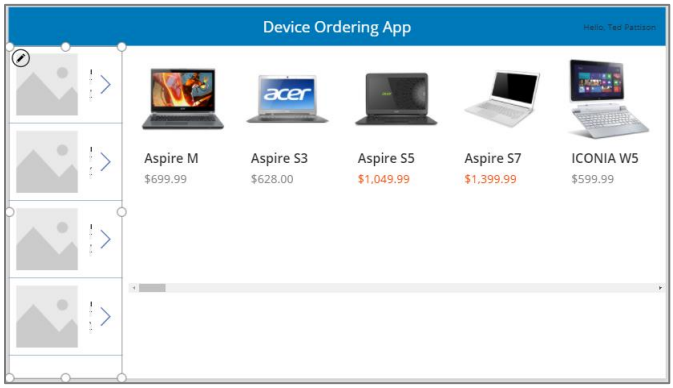
Over the next few steps, you'll add a second gallery that will list the various device manufactures. This gallery will later be used as a filter for the device gallery created earlier in this lab.

10. Create a second gallery that will show the list of manufactures of all the devices. This should be a single column, vertical gallery down the left side of the screen, with each cell displaying the manufacturer's logo image.
   a) Insert a vertical gallery on the left.
   b) Do this by clicking the **Insert** tab on the ribbon and opening the **Gallery**, then select **Vertical.**



   c) Close the **Data** pane**,** we will come back to it.
   d) Rename the gallery to **ManufacturerGallery**.

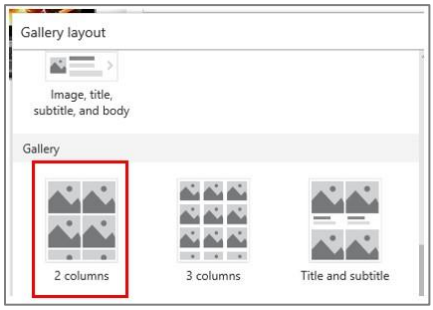
11. Reposition this new gallery.
    a) Move it so that it's left aligned with the left edge of the screen and top aligned with the top of the device gallery.
    b) Use the right middle grab handle to make the gallery narrower so it's touching the left edge of the device gallery.
    c) Use the bottom grab handle to align the bottom of the gallery with the bottom of the screen.



12. Change the gallery layout.
    a) With the manufacturer gallery selected, click on the **Data** setting in the Properties tab on the right to open the **Data** pane.

    b) In this **Data** pane, select the **Layout** dropdown.

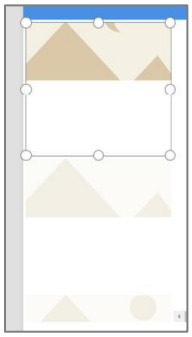
    c) Scroll down and select the layout called **"2 columns"** with just images and no text.



d) Close the **Data** pane.

e) Click the **Properties** tab in the right pane and change **Wrap count** from 2 to 1 to convert the gallery to a single column layout.

f) Select the image control within the gallery and reduce its height by dragging the middle bottom drag control upwards. The image size will reduce whereas the template size will still be expanded.



g) Next, select the template cell by clicking the white space just below the image in the first cell. You can also select the template cell by selecting the entire gallery and clicking the pencil icon in the top left.

h) Reduce the height of the template cell to match the image. You essentially want the image to occupy the entire cell.

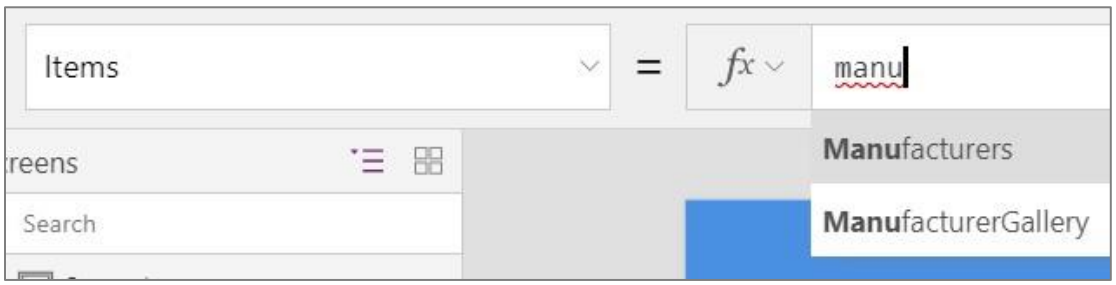
13. Connect the gallery to the **Manufacturers** table

a) For the devices gallery, you connected it to the data source using the Data tab in the right pane.

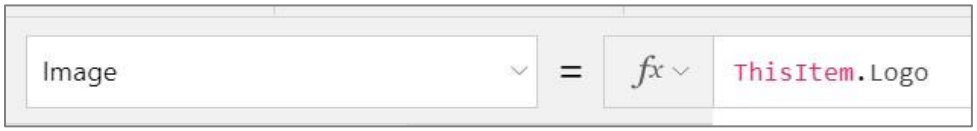
i) You can also connect to data via the formula bar

b) Select the **ManufacturerGallery** and make sure the whole gallery is selected and not just the first cell.

c) Select **Items** property and type **Manufacturers** in the formula bar.



14. Display manufacturer logos in this gallery.
    a) Notice that the gallery gets populated with images of buildings.
        i) This is because PowerApps picked a default binding which mapped to the HQ column in the table.
    b) Select the image control in the first cell of the gallery and change the value of the **Image** property from `ThisItem.HQ` to **`ThisItem.Logo`**.



    c) All the gallery items will now display logo images.

15. Highlight the selected item in the gallery:
    a) Use the **TemplateFill** property of the manufacturer gallery to specify a highlight color for the selected item.
    b) With the whole gallery selected (not just the first cell), set the **TemplateFill** property to:

```
If( ThisItem.IsSelected,ColorFade(HeaderLabel.Fill,75%) )
```

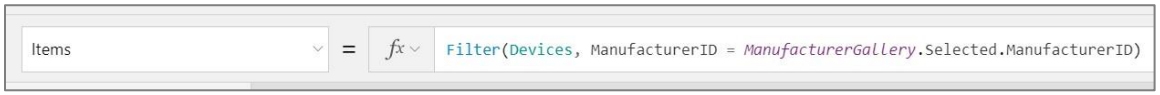
    c) This is conditionally setting a Fill color if the cell is selected.
    d) You could have set a specific color or RGB value, but we recommend using the **ColorFade** function so it matches the header label with a 75% fade. If you change the fill color of header label, this template fill color will automatically change.

    e) Click different items in the gallery, notice the selected item is highlighted in a light blue color.

Over the next few steps you will configure the manufacturer gallery as a filter for the device gallery. This will assist the user to browse the list of available devices.

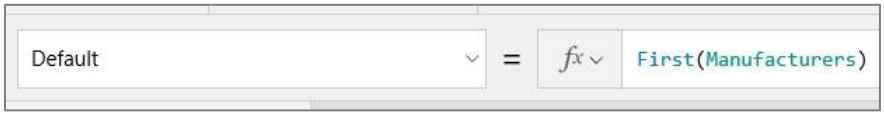
16. Use the **Filter** function to filter the device gallery based on the **ManufacturerID** of the selected manufacturer
    a) Select the **DeviceGallery**. With the **Items** property selected, enter the following expression in the formula bar:

```
Filter(Devices, ManufacturerID = ManufacturerGallery.Selected.ManufacturerID)
```



    b) This will filter the device gallery to only display items that match the selected manufacturer based on **ManufacturerID**.
    c) Test you work by selecting different items in the manufacturer gallery on the left. When you do this, you should be able to verify that device gallery is updated accordingly.

17. Set the default when the app loads to select the first manufacturer. We don't want a blank screen when the app first loads
    a) **Select the entire gallery** (by clicking **ManufacturerGallery** in the tree view on the left)
    b) Set the **Default** property of the gallery in the formula bar to **First(Manufacturers)**.

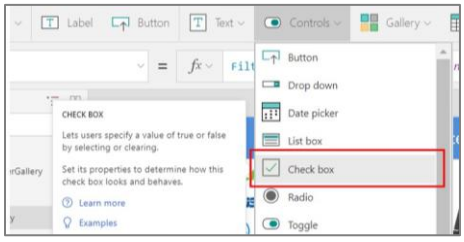


This will configure the manufacturer gallery to be initialized using the first item in the table when the app starts.

Exercise 3: Add Checkboxes for Users to Select Devices

Over the next few steps, you'll add checkboxes that allow users to select the devices they want to compare.

1. Add a checkbox that users can select to add a device to the comparison list
    a) Select the device gallery. Click the Pencil edit icon in the top left of the gallery to select the template cell.
    b) Make sure that only the first item in the gallery is selected (not the entire gallery).
    c) Add a check box by navigating to the **Insert** tab in the ribbon and selecting the **Controls > Checkbox** menu command.



d) Move the inserted checkbox below the label control for price

e) Change the checkbox text to **"Compare"**. You can do this by setting the **Text** property or clicking within the text of the control and typing directly into the control.

f) Confirm that there is a checkbox in each of the gallery items, similar to the image in the screenshot above.

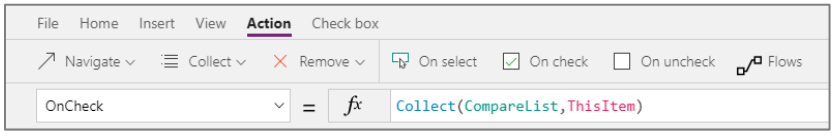
If there is only one checkbox on the screen, it's likely that you placed it outside the context of the gallery. In that case, delete it, click again on the first template cell within the gallery and re-insert the control.

2. When checked it should add the selected device to a Collection

a) When a user selects a device to compare, we will add it to a collection called **CompareList**. You can think of this as an in-memory collection of devices that have been selected for comparison.

b) Select the Checkbox control that you just inserted in the gallery, and click on the **Action** tab in the ribbon, click **OnCheck** and set the value in the formula bar to:

```
Collect(CompareList,ThisItem)
```

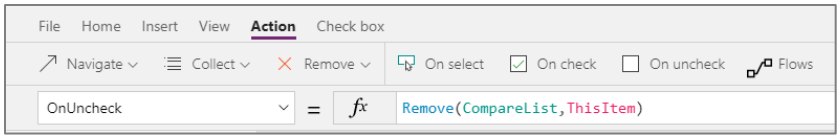


3. When unchecked it should remove the device from the collection

a) Set **OnUncheck** to the following expressions.

```
Remove(CompareList,ThisItem)
```

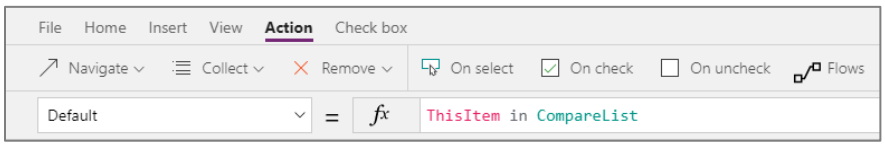
b) Your formula bar should match the following screenshot.



c) Also, set the **Default** property of the checkbox to:

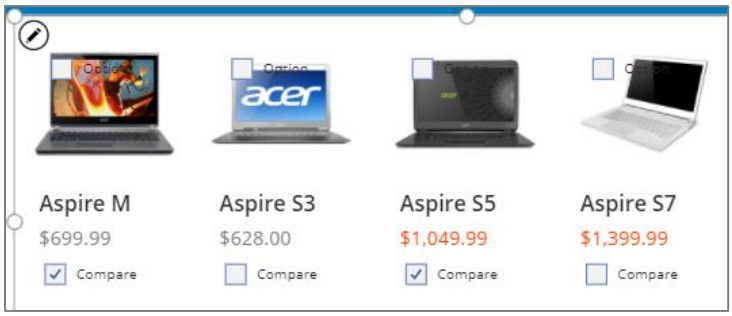
```
ThisItem in CompareList
```

d) Your formula bar should match the following screenshot.

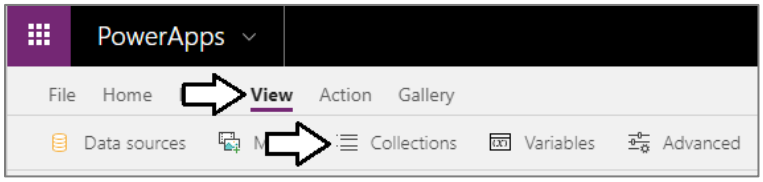


The **Default** setting of the checkbox is a true or false value that determines whether the checkbox should be checked or not when it is first displayed. Setting the **Default** property to this formula will ensure that all checkbox stay in sync with the **CompareList** collection.

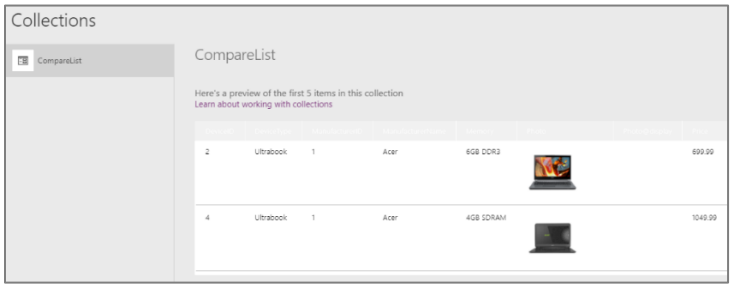
4. Test that items have been added to the **CompareList** collection
   a) Click a few checkboxes to select some devices.



   b) Click on **View** > **Collections**

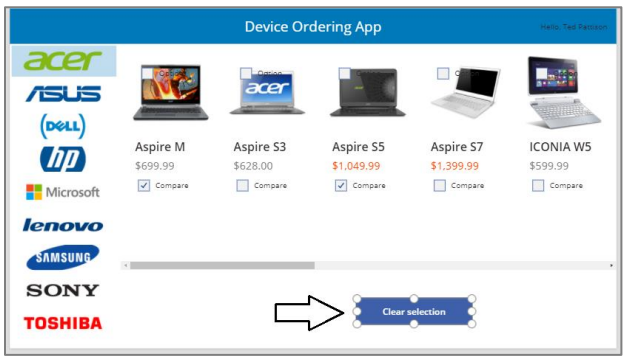


   c) You should be able to see the contents of the collection you have created for devices named **CompareList**.



Over the next few steps, you will add two buttons to the screen – one to clear the selection and another to navigate to the second screen to submit the approval request.

5. Add the **Clear** selection button: this should clear all selected check boxes
   a) On the left tree view, click **MainScreen**, click on the **Insert** tab on the ribbon and select **Button** to add a button to the screen.
   b) Position the button just left of the bottom right corner (leave room for the compare button to the right).
   c) Set the button's **Text** property to: "Clear Selection"

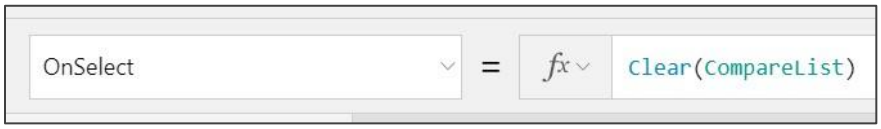


d) In the left navigation, change the name of the button to **ClearSelection**.

e) Set the **OnSelect** property for this button to:

```
Clear(CompareList)
```

f) Your formula bar should match the following screenshot.



Clicking the **Clear selections** button will now remove all the items from the **CompareList** collection

6. Add the "Compare X item(s)" button, where X is the number of devices the user has selected via check boxes.

a) This button won't do anything yet, but will eventually take you to the comparison screen.

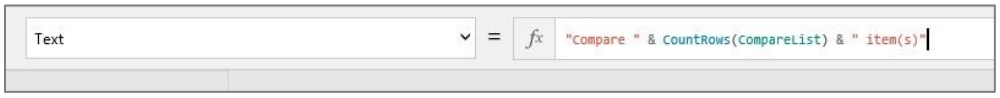
b) On the left tree view, click on **MainScreen**, click **Insert** -> **Button** to add a button to the screen.

c) Position the button in the bottom right corner.

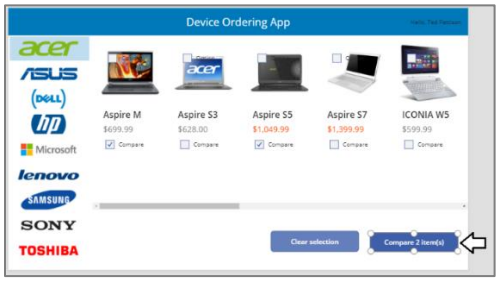
d) In the left navigation, change the name of the button to **CompareItems**.

e) Set the button's **Text** property to:

```
"Compare " & CountRows(CompareList) & " item(s)"
```



f) Resize the button so the text fits – use the grab handle to make it wider.



g) To disable the button when no items are selected, select the button and set its **DisplayMode** to the following formula.

```
If(CountRows(CompareList) > 0, DisplayMode.Edit, DisplayMode.Disabled)
```

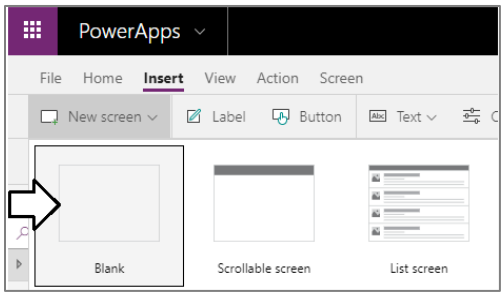
h) This will enable the button only if the number of items in **CompareList** is greater than zero.

i) You can test this by unselecting all devices – notice the button is grayed.

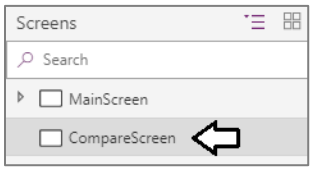
Exercise 4: Add a New Screen for Device Comparison

In this exercise, you'll create a second screen used for comparing the selected devices.

1. Add a second screen to your app
   a) From the ribbon click **Home** > **New Screen,** and choose **Blank**.



   b) Rename the screen to **CompareScreen**.



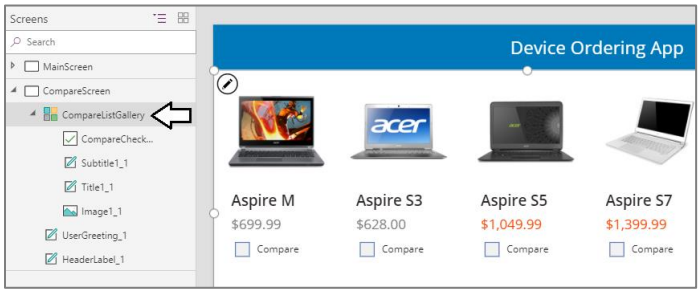
Note that you can copy and paste screen elements from the **MainScreen** to the **CompareScreen**.

2. Give the new screen the same header as the first screen by copying and pasting from one screen to the other.
   a) Click on **MainScreen** in the left navigation bar and keep the Ctrl key pressed down while multiselecting the blue header label and the "Hello, <user>" label. Once you've multi-selected the controls, type (Ctrl-C) to copy them to the clipboard.
   b) Go the **CompareScreen** by clicking in the left navigation bar, and type (Ctrl-V) to paste the controls. You may need to move them up to align with the top of the screen.

3. Extend **CompareScreen** with a device gallery for viewing the devices selected for comparison
   a) Copy the **DeviceGallery** control from **MainScreen** and paste a copy into the newly created **CompareScreen**.
   b) Move the gallery to the left edge of the screen.
   c) Align the top of the gallery to be just under the header banner.
   d) Use the right drag handle to reduce the width of the gallery and create space for a data entry form on the right of the screen.

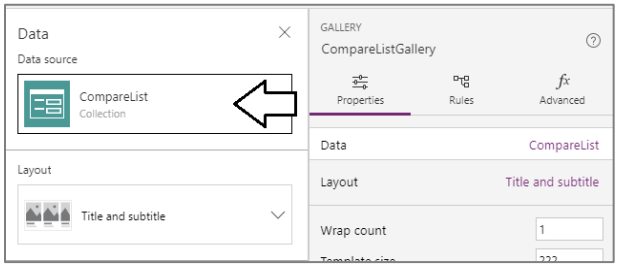
You will complete the layout of this screen later in this lab by adding a Form control on the right-hand side.

e) Rename the gallery to **CompareListGallery**.

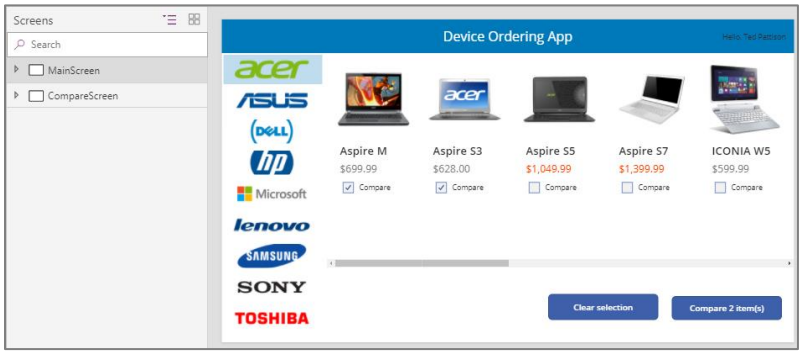


4. Configure the **Data** property setting for the **CompareListGallery** control.
   a) Currently, the **Data** property setting for the **CompareListGallery** control in the **Properties** pane has a value of **Devices**.
   b) Click on the link to edit the **Data** property setting.

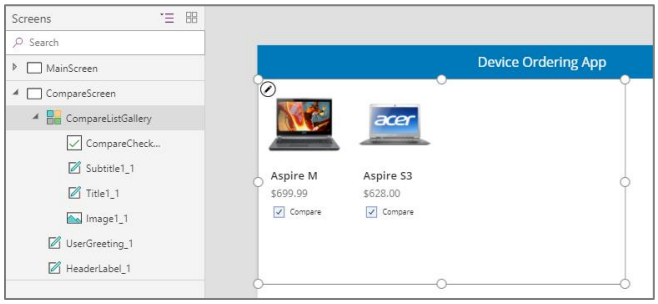
   c) Change the data source to the **CompareList** collection.



5. Make sure the **CompareList** collection is not empty.
   a) Navigate to the **MainScreen**.
   b) Make sure at least 2 or 3 devices are selected as shown in the following screenshot.

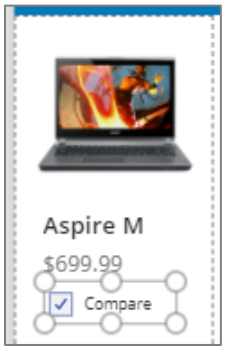


c) Return to **CompareScreen** and verify the you can see the devices that you selected on **MainScreen**.



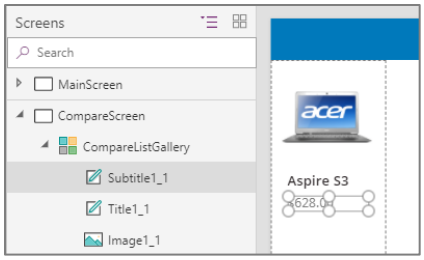
6. Remove the **Compare** checkbox from **CompareScreen**.

a) Select the **Compare checkbox** on the left most template cell and delete it.



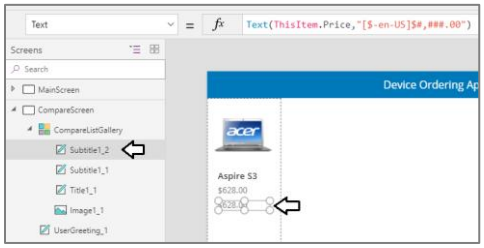
7. Add additional fields to the gallery to view more details about the device.

a) Select the **Price** label and copy it to the clipboard using **Ctrl-C**.



b) Paste (Ctrl-V) the contents of the clipboard to create a copy of the label with the **Price** value.

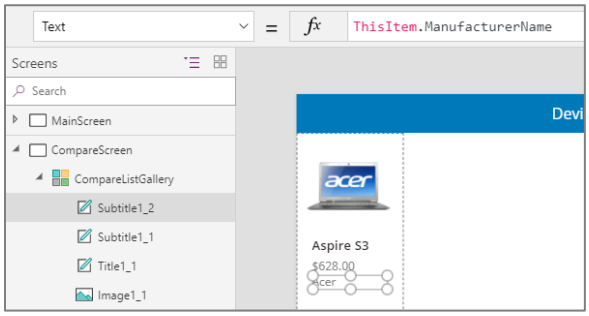
c) Move the copied label down just underneath the original **Price** label.



d) Modify the **Text** property of the new label with the following formula to display the manufacturer name.

```
ThisItem.ManufacturerName
```

e) The new label should now display the name of the device manufacturer.



f) Modify the **Color** property of the new label to **Blue**.

g) Modify the **Font Size** property of the new label to **14** as shown in the following screenshot.

Now that you have created a new label to display the manufacturer name, you will copy and paste the new label several more times to create labels to display additional information about the selected devices.

h) Copy the manufacturer name label and paste it to create a new label.

i) Move the new label down underneath all the other labels and set its **Text** property using the following formula.

```
ThisItem.DeviceType
```

j) Copy the manufacturer name label and paste it to create a new label.

k) Move the new label down underneath all the other labels and set its **Text** property using the following formula.

```
ThisItem.Processor
```

l) Copy the manufacturer name label and paste it to create a new label.

m) Move the new label down underneath all the other labels and set its **Text** property using the following formula.

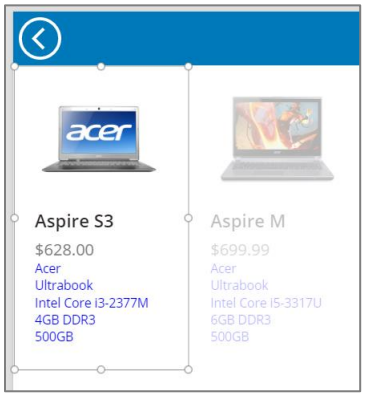
```
ThisItem.Memory
```

n) Copy the manufacturer name label and paste it to create a new label.

o) Move the new label down underneath all the other labels and set its **Text** property using the following formula.

```
ThisItem.Storage
```

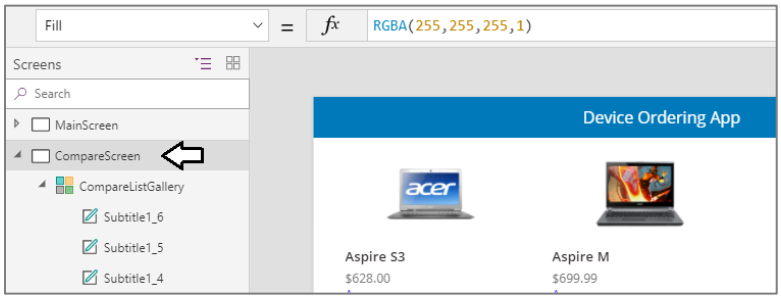
p) At his point, your item template for the gallery should match the following screenshot.



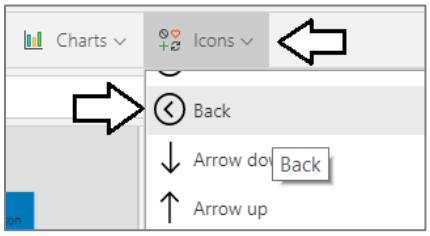
q) Make the template item wider so you can see all the data in each field including the processor type.

8. Add a Back button to the header of **CompareScreen** to navigate back to **MainScreen**.

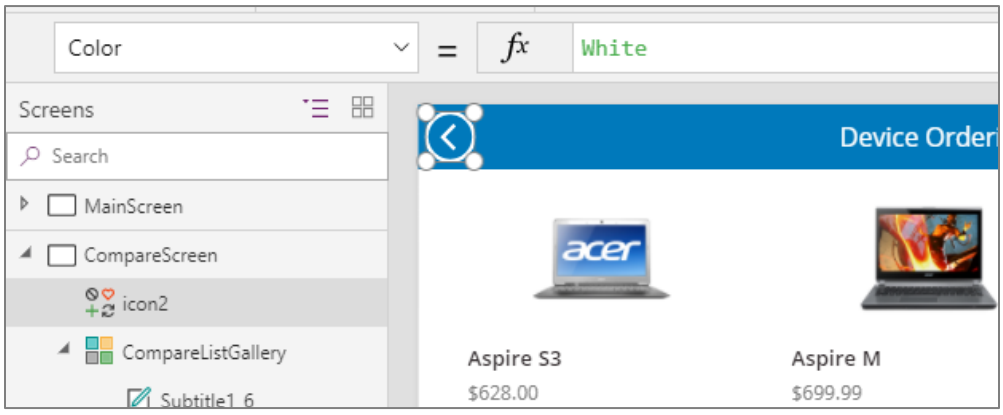
a) Click on **CompareScreen** in left navigation.



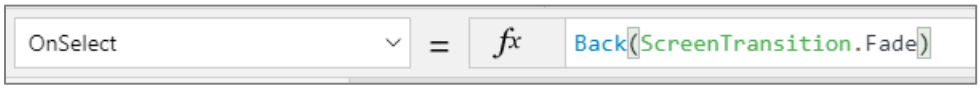
b) Go to **Insert**, then **Icons** and select the **Back** icon. Position it in the upper left corner of the screen.



c) With the arrow icon selected, change the **Color** property to **White**. You can change this in the formula bar or through the Properties pane on the right.



d) Set the **OnSelect** action for the icon to **Back(ScreenTransition.Fade)**. This will trigger navigation back to the main screen.



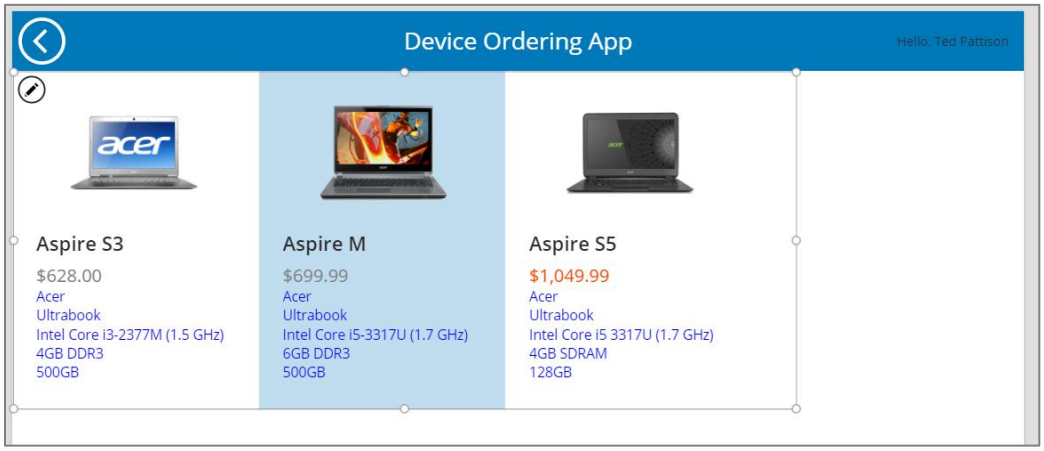
9. Configure the **TemplateFill** property of the **CompareListGallery** to highlight the selected device.
   a) Like the behavior in the manufacturer gallery on the first screen, use the **TemplateFill** property to highlight the selected item:
   b) Select the gallery (**CompareListGallery**)
   c) Set the **TemplateFill** property to:

```
If(ThisItem.IsSelected,ColorFade(HeaderLabel.Fill,75%))
```

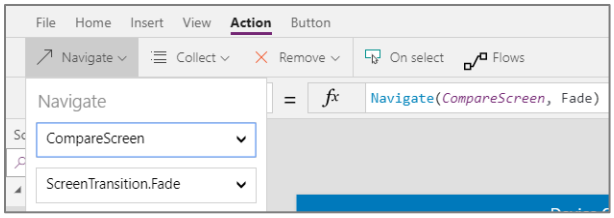
This formula will conditionally set a Fill color if the cell is selected.

   d) You could have set a specific color or RGB value, but we recommend using the ColorFade function so it matches the header label with a 75% fade. If you change the fill color of header label, this template fill color will automatically change.

   e) Click different devices in the gallery, notice the selected item is highlighted in a light blue color.

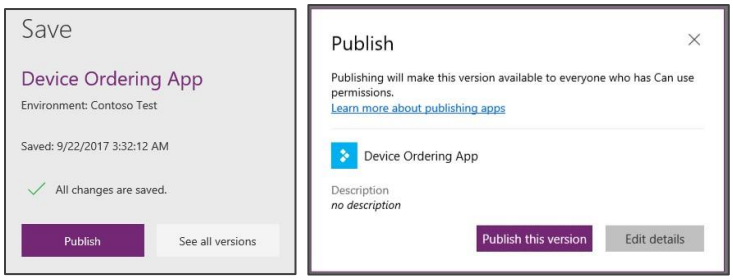


10. Wire up the **Compare** button on the first screen to navigate to the second screen
   a) Navigate back to **MainScreen**.
   b) With the **Compare** button selected, click on the **Action** tab and select **Navigate**.
   c) Select **CompareScreen** from the dropdown
   d) Select a transition type, you can select **ScreenTransition.Fade**.



   e) Click the **Compare** button and verify that it takes you to the second screen.

11. First, make sure to save the app by clicking **File > Save**
12. Use the **Play** button in PowerApps Studio to test out your app
    a) Test the filter by manufacturer functionality.
    b) Select a few devices to compare.
    c) Hit the compare button to go to the second screen.
    d) Verify that any device you selected is highlighted with a border.
    e) Hit the back icon and confirm you get back to the main screen.
13. Test your app on a mobile device:
    a) Publish the app. Select **File** -> **Save** and click the **Publish** button. Click **Publish this version** on the confirmation prompt.



    b) You should see a confirmation.

If you haven't already installed the PowerApps mobile application, go to the app store on your device, search for "PowerApps" and install the PowerApps mobile application. This is often referred to as the **PowerApps Player** or the PowerApps container app within which all your published apps will run.

14. Run the PowerApps player app and sign in with your Office 365 trial account.
    a) Look for the **Device Ordering App** in the app list and run it.
    b) Since this is a tablet app it will run in landscape mode. Although the controls will be a bit small on a mobile phone, you should be able to interact with it and test out the app functionality.