

Getting Started with Power Apps

Lab Time: 60 minutes

Lab Folder: C:\Student\Modules\08_PowerPlatform\Lab

Lab Overview: This lab covers how to get up and running with the Power Platform by creating a new Microsoft 365 tenant with trial subscriptions to Office 365, Power Apps, Flow, Power BI and SharePoint Online. The act of creating and configuring this new Microsoft 365 tenant will yield an isolated testing and development environment for building and testing the apps and components you can build with Power Apps and Flow. One valuable aspect of creating your own new Microsoft 365 tenant is that you will have Global tenant administrative permissions allowing you to create multiple Microsoft 365 user accounts for testing your apps and flows in isolation from any other existing Microsoft 365 tenant.

Exercise 1: Create a Canvas App using the Start From Data Template

In this exercise you will create a new app using **Start from Data** template.

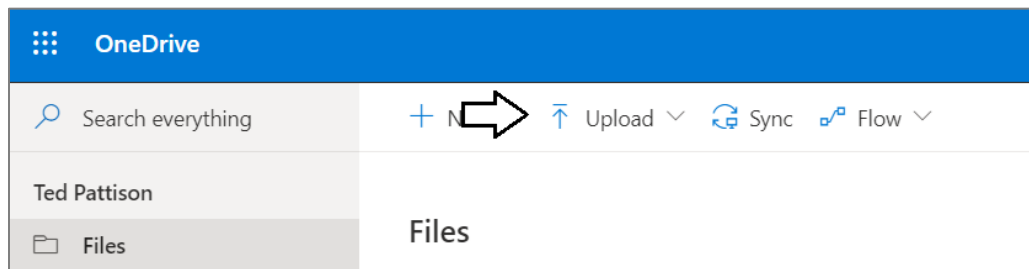
1. Upload the Excel workbook named **Expenses.xlsx** to OneDrive for Business.
 - a) Using Windows Explorer, verify that there is an Excel workbook file named **Expenses.xlsx** located at the following path.

C:\Student\Modules\08_PowerPlatform\Lab\Expenses.xlsx

- b) Drop down the Microsoft 365 app launcher menu and select **OneDrive** to navigate to your **Files** collection.



- c) Click the **Upload** button and then select **Expenses.xlsx** to upload this file to OneDrive for Business.

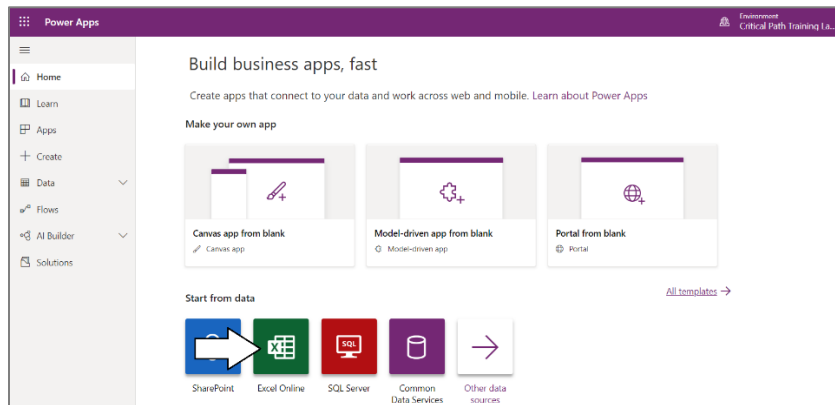


- d) Verify that **Expenses.xlsx** has been uploaded to your **Files** folder.

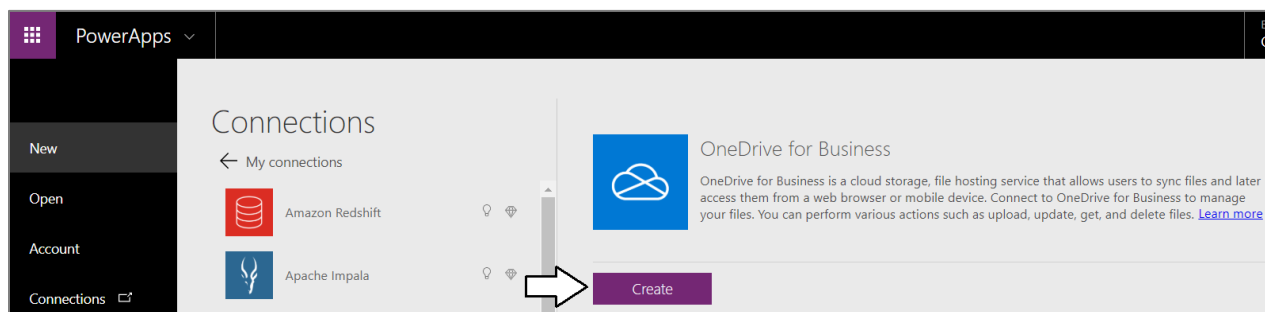
Files				
	Name ▾	Modified ▾	Modified By ▾	File Size ▾
	Expenses.xlsx	A few seconds ago	Ted Pattison	12.1 KB

Next, you will create a new Canvas app in Power Apps Studio that will read and write to the **Expenses** table in this Excel workbook.

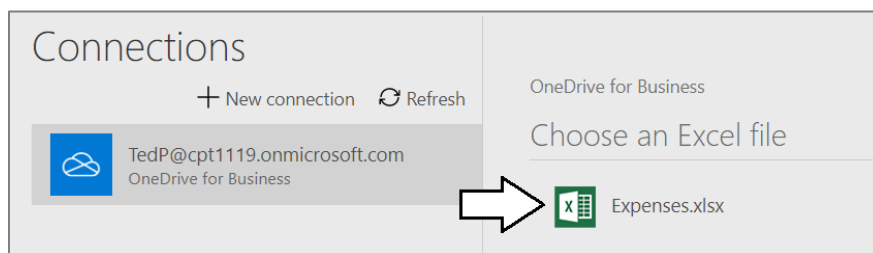
2. Create the new app using the data in the **Expenses.xlsx** workbook.
 - a) Navigate back to the **Home** page of the Power Apps portal at <https://make.powerapps.com>.
 - b) Click on the **Excel Online** tile in the **Start from data** section to begin the process of creating the new app.



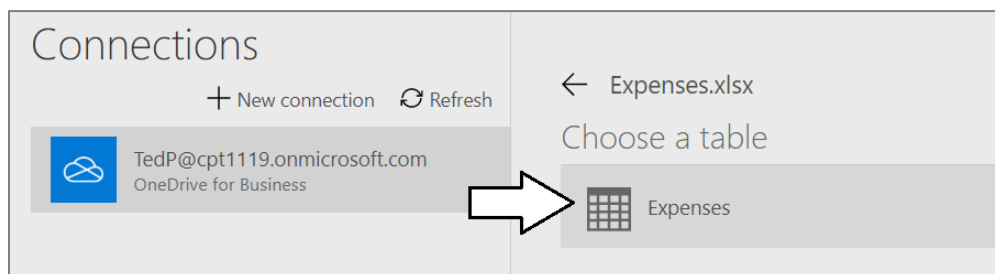
- c) Click the **Create** button to create a new connection using the **OneDrive for Business** connector.



- d) When prompted to **Choose an Excel file** on the **Connections** page, click the Excel workbook file named **Expenses.xlsx**.

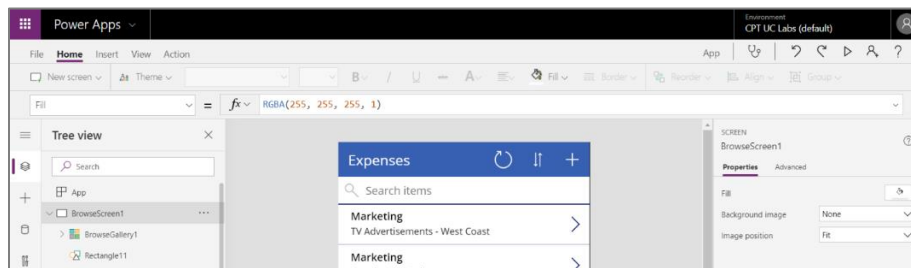


- e) When prompted to **Choose a table** on the **Connections** page, select the **Expenses** table and then click **Connect**.



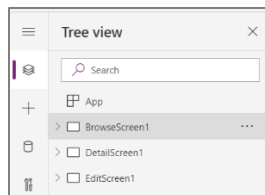
- f) Wait while Power Apps Studio generates the starting point for your app.

- g) If you are presented with the **Welcome to Power Apps Studio** dialog, click **Skip**.
- h) Once Power Apps Studio has created the new app, it should appear as the one in the following screenshot.

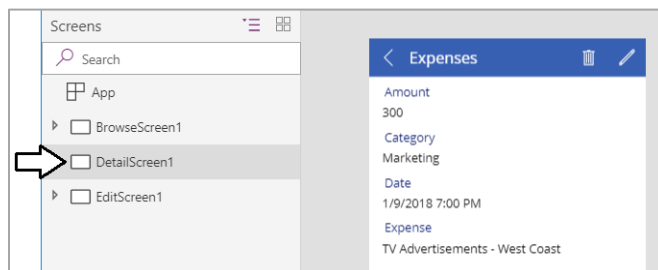


The new app has been created with three different screens. The browse screen shows many expenses at one time. The detail screen and the edit screen are both designed to display only one expense at a time.

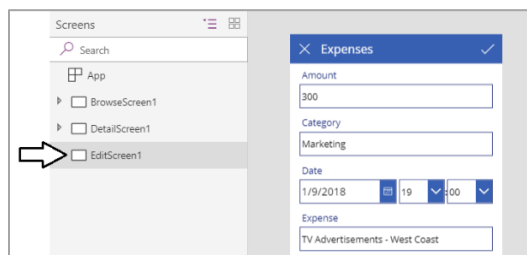
- i) Collapse the **Tree View** nodes for the screens named **BrowserScreen1**, **DetailScreen1** and **EditScreen1**.



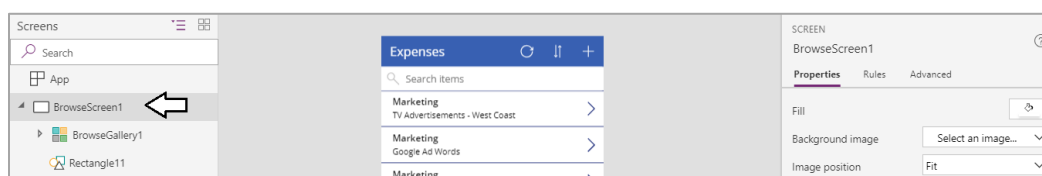
- j) Click on **DetailScreen1** in the left navigation to inspect the detail form.



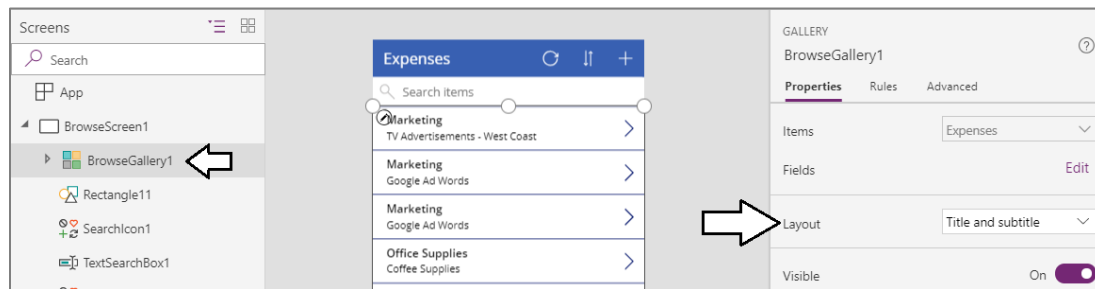
- k) Click on **EditScreen1** in the left navigation to inspect the edit form.



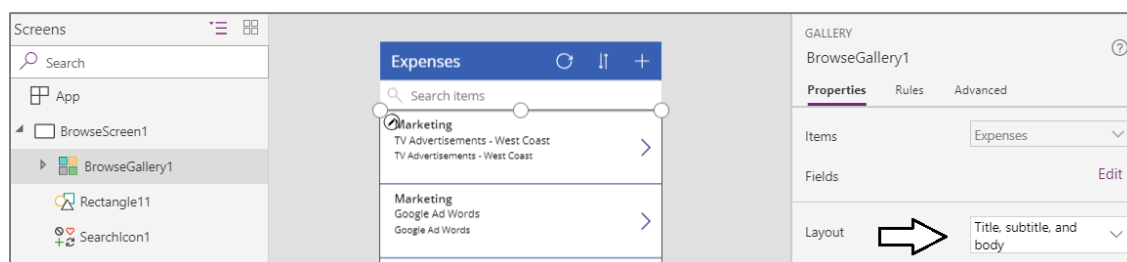
- l) Click on **BrowserScreen1** and expand its node in the left navigation.



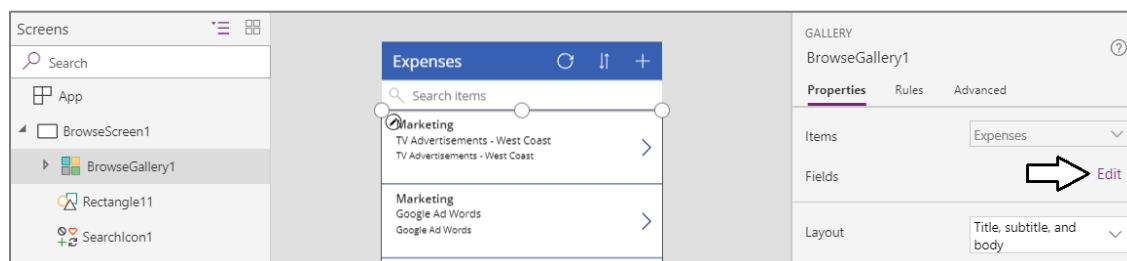
- m) Click the **BrowseGallery1** control and then locate the Layout section in the Properties pane on the right.
- n) You should see that the **BrowseGallery1** control currently has a **Layout** setting of **Title and subtitle**.



- o) Update the **Layout** setting for **BrowseGallery1** to a value of **Title, subtitle and body**.

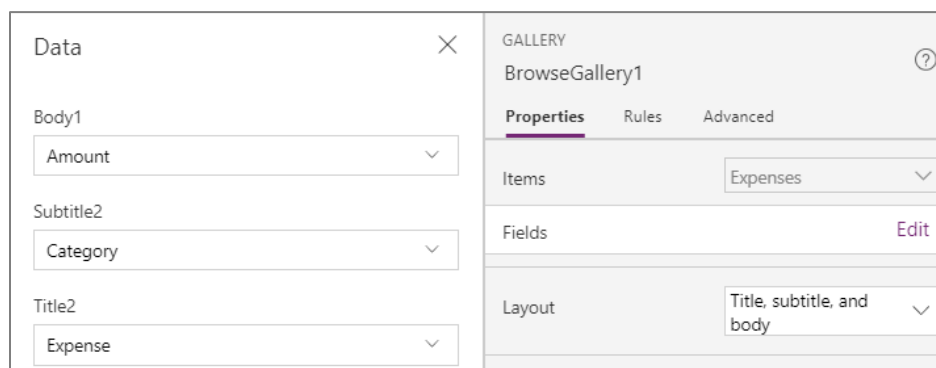


- p) Click on the **Edit** link for the **Fields** property to Display the Data pane.

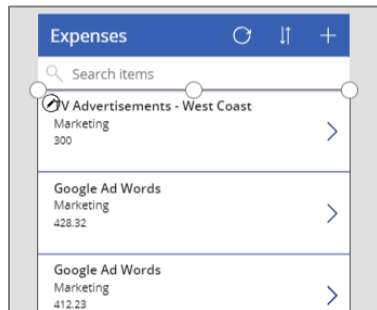


You should now see the Data pane allow you to map fields to items in the gallery template for **BrowseGallery1**.

- q) Set **Body1** to the **Amount** field.
- r) Set **Subtitle1** to the **Category** field.
- s) Set **Title1** to the **Expense** field.



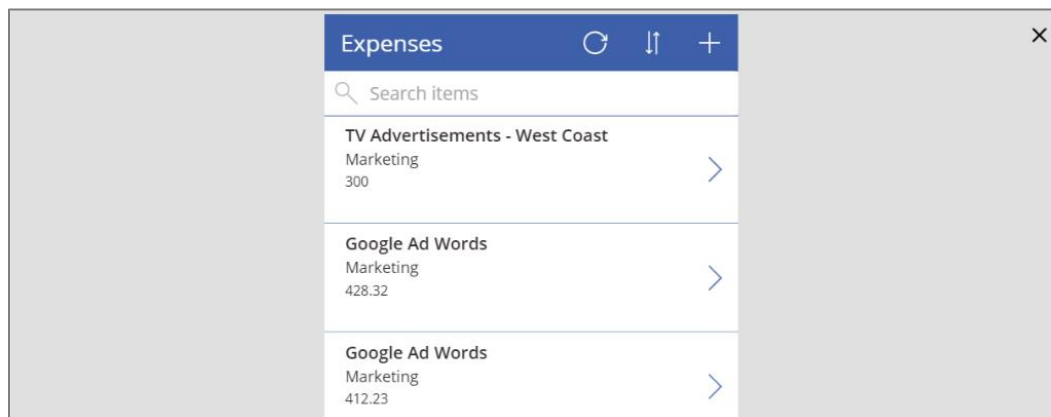
- t) Close the Data pane by clicking the x in the top right corner.
- u) The browse screen should now display its fields ordered by **Expense**, **Category** and **Amount**.



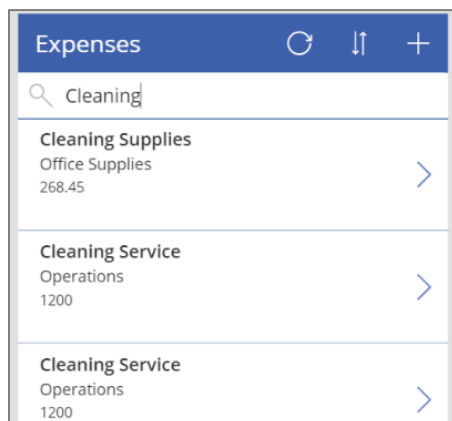
- 3. Test the app by starting it up and testing the search functionality.
 - a) Click the Start button with the arrow icon to launch the app for testing.



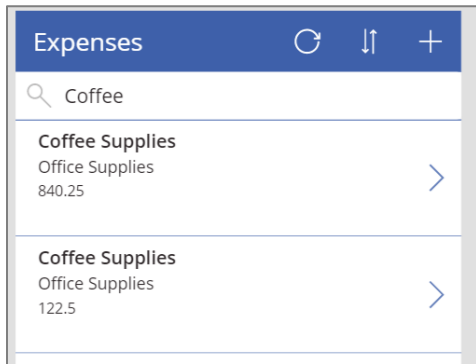
- b) The app should start and appear as shown in the following screenshot.



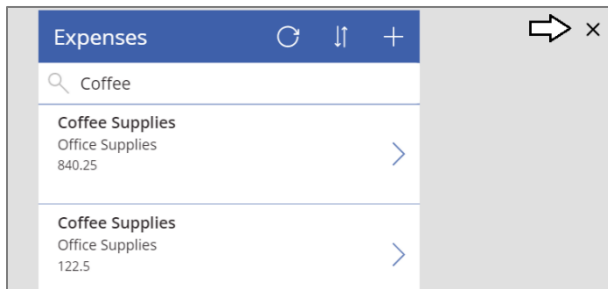
- c) Test search functionality by typing the word "Cleaning" in the search box.



- d) Try a different search by typing the word "Coffee" into the search box.

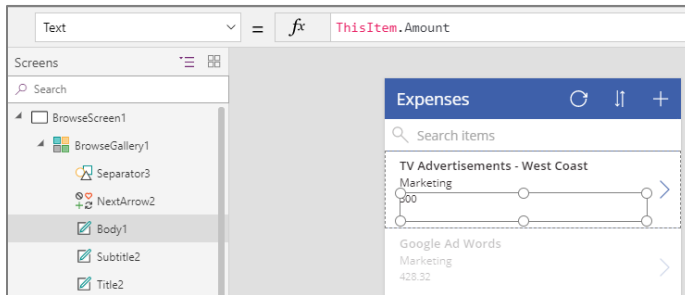


- e) Once you have tested the search functionality, stop the app by clicking the button with the **X** icon at the top right.



4. Configure the formatting of the expense **Amount** field.

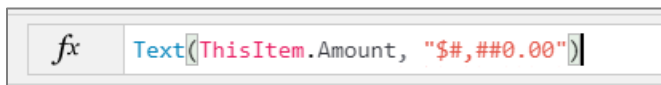
- a) Select the textbox named **Body1** which displays the **Amount** field for each expense. You should be able to see that the **Text** value of this textbox currently configured with a formula which is **ThisItem.Amount**.



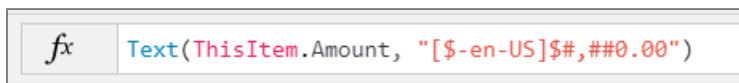
- b) Update the **Text** property of the **Body1** textbox with the following formula.

```
Text(ThisItem.Amount, "$#,##0.00")
```

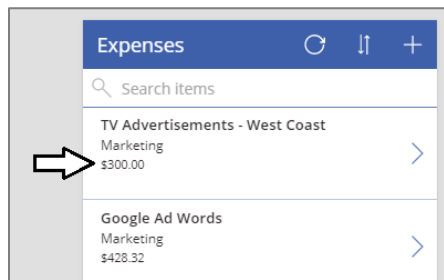
- c) When you update the formula, it will initially match the following screenshot.



- d) Note that after you update the formula, Power Apps Studio will automatically update the formula to include **[\$-en-US]**.

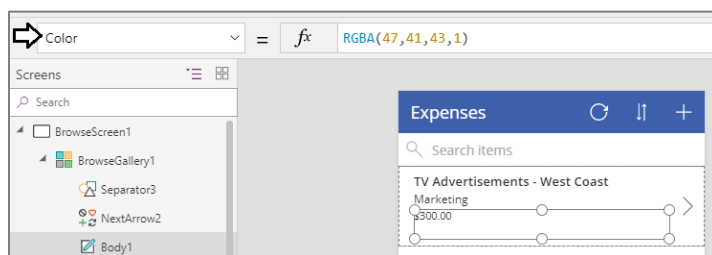


- e) The **Amount** field should now display its value with currency formatting.



5. Configure the **Color** property of **Body1** to display **Amount** values in red when they are \$500 or greater.

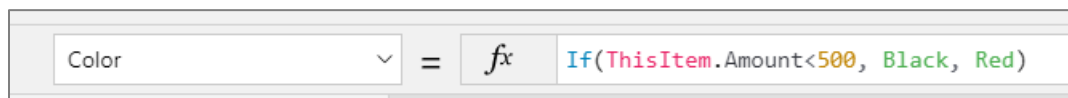
- a) With the **Body1** control selected, use the property drop down to display the **Color** property.



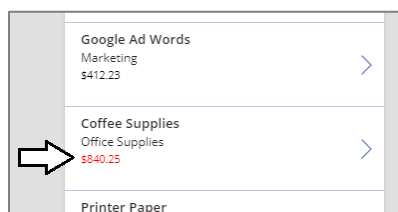
- b) Update the **Color** property for **Body1** with the following formula.

If(ThisItem.Amount<500, Black, Red)

- c) The formula bar should match the following screenshot.

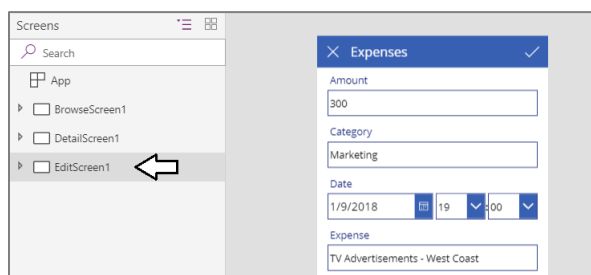


- d) You should now see that **Amount** values of \$500 or greater are displayed with a red font.

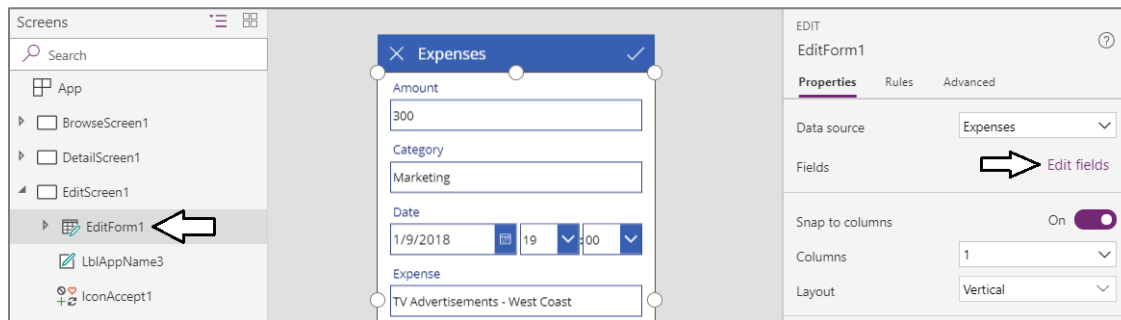


6. Modify the edit screen to streamline data entry for new expenses.

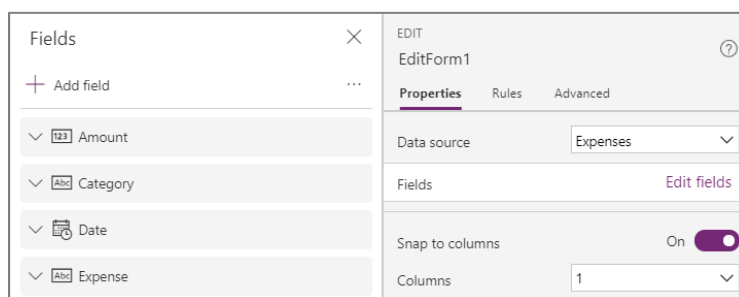
- a) Using the left navigation, move to the edit screen.



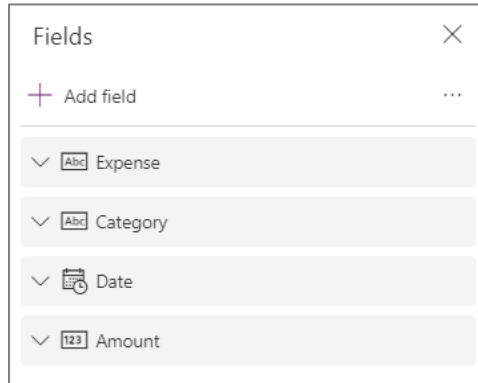
- b) Display the **Data** pane so you can see the **Fields** collection of the edit form. At this point, the fields are sorted alphabetically.



- c) Using the mouse, rearrange the fields by moving **Expense** to the top followed by **Category**, **Date** and then **Amount**.

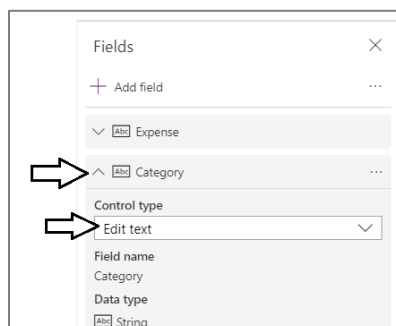


- d) The edit screen should now display its fields using the new sort order.

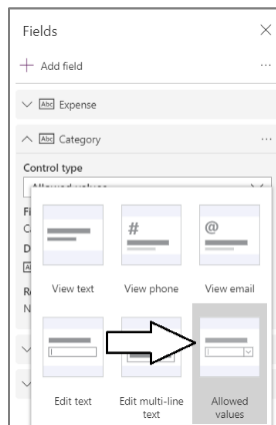


7. Update the data card for the **Category** field to provide a dropdown list with allowed values.

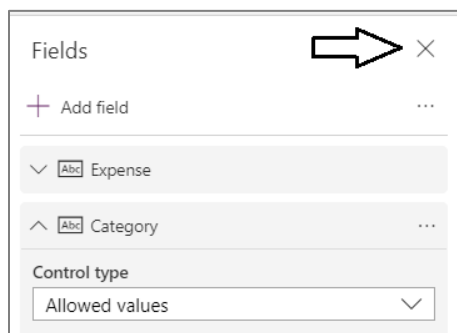
- a) Drop down the menu with the **abc** icon to the right of the **Category** field.



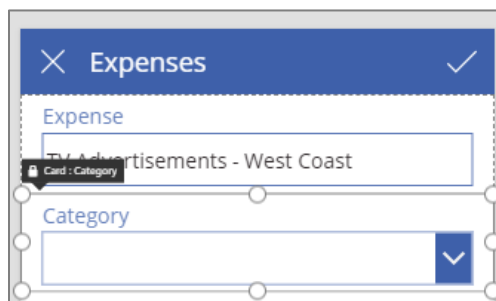
- b) Select a control type of **Allowed Values**.



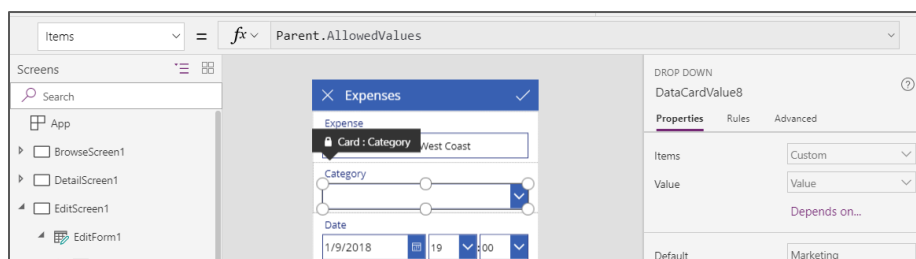
- c) Close the **Fields** pane.



- d) The control which displays the **Category** field should change to a dropdown menu.

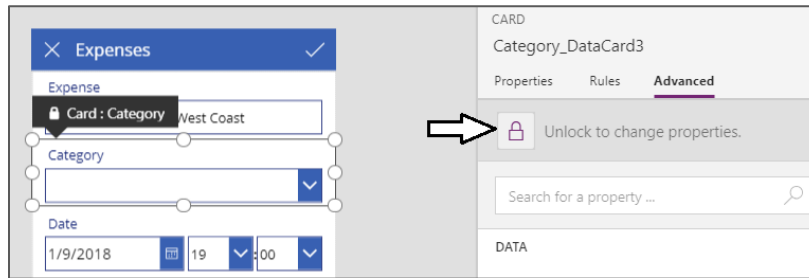


- e) Select the dropdown menu and examine the **Items** property in the formula bar.



You will notice that the formula bar is read-only for the **Items** property because the data card is locked by default.

- f) In the Advanced pane, click the Unlock to change properties button.

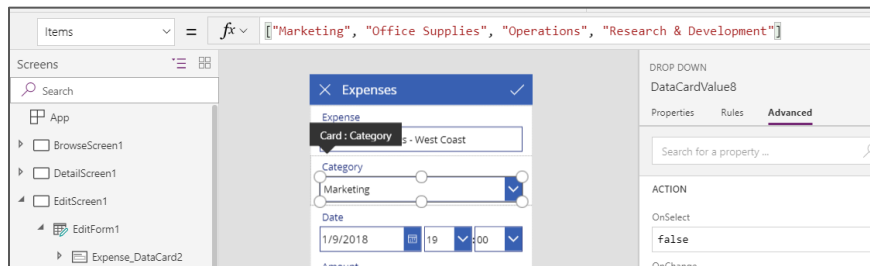


Note that the user interface experience might seem a bit strange when you click the **Unlock to change properties** button. At first it seems like nothing is happening. However, after a few seconds you should see that the **Items** property becomes editable.

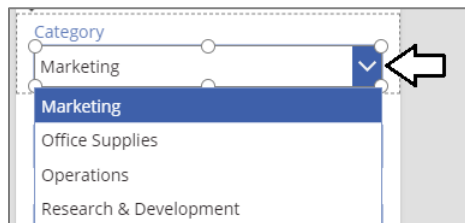
- g) Update the **Items** property of the dropdown list with the following formula.

["Marketing", "Office Supplies", "Operations", "Research & Development"]

- h) The formula bar should match the following screenshot.

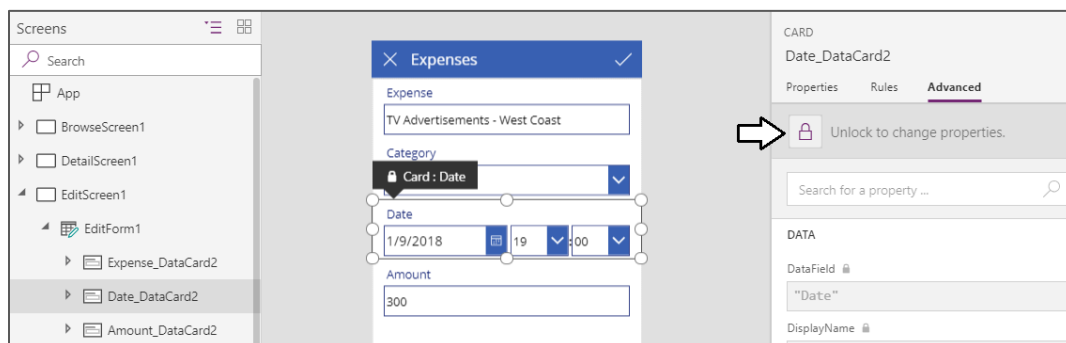


- i) If you start the app, you should be able to test the dropdown list and verify that it provides four allowed values.

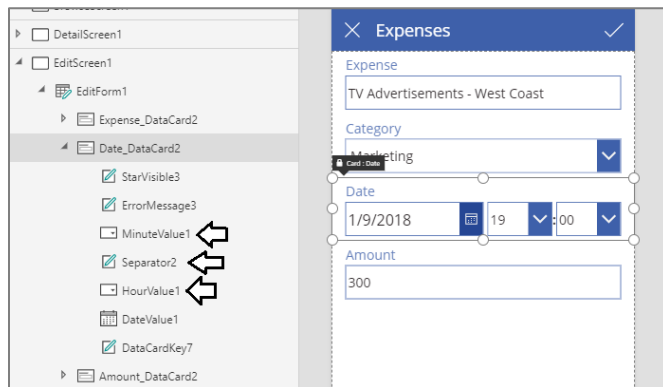


8. Update the data card for the **Date** field to make it a date-only.

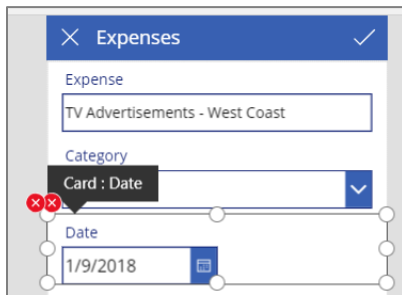
- a) Select the data card for the **Date** field.
b) In the **Advanced** pane, click the **Unlock to change properties** button for the data card for the **Date** field.



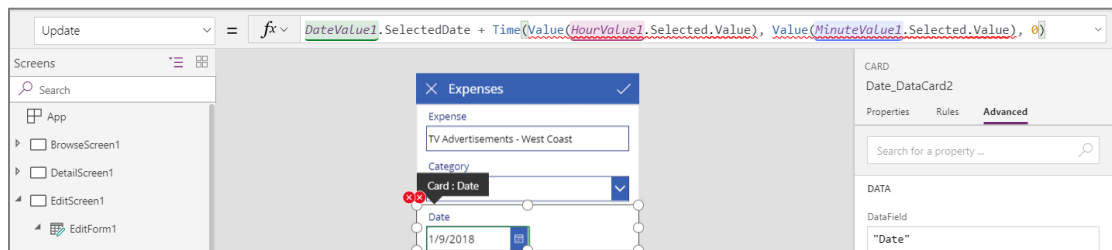
- c) Using the left navigation, select and delete the controls named **MinuteValue1**, **Seperator2** and **HourValue1**.



- d) After deleting **MinuteValue1** and **HourValue1**, you will notice formula errors due to referencing deleted controls.
e) Click on the red error icon with to the left.



- f) At this point, you should see the formula for the **Update** property in the formula bar.



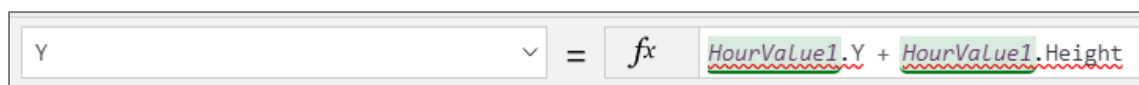
- g) Replace the existing **Update** formula with the following formula to remove references to **HourValue1** and **MinuteValue1**.

DateValue1.SelectedDate

- h) The formula for the **Update** property of the data card should now appear as the formula shown in the following screenshot.



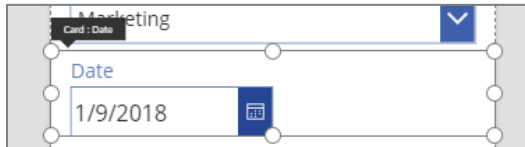
- i) Click on the one remaining red error icon to display the other formula error.
j) You should see the **Y** property of **ErrorMessage3** contains references to the deleted control named **HourValue1**.



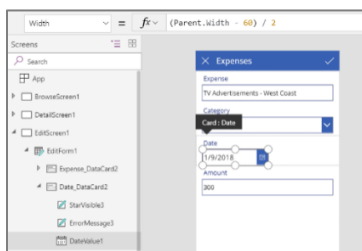
- k) Replace the existing formula for the **Y** property with a value of **0** as shown in the following screenshot.



- l) At this point, you should no longer see any error indicators.



- m) Select the **DateValue1** control and examine its **Width** property.



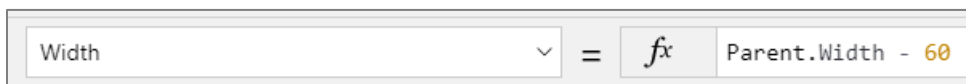
- n) You should see that the formula of the **Width** property has the following value.

(Parent.Width - 60) / 2

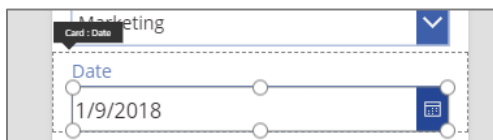
- o) Update the formula of the **Width** property to the following formula.

Parent.Width - 60

- p) Your formula bar should match the following screenshot.

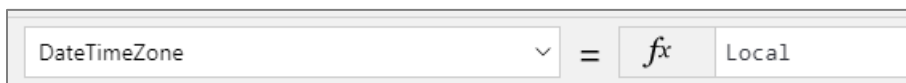


- q) The **DateValue1** control should expand to the same width of the other input controls on the edit screen.

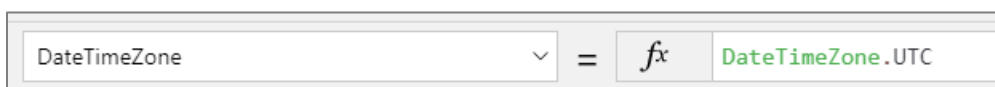


9. Update the **DateTimeZone** property of **DateValue1**.

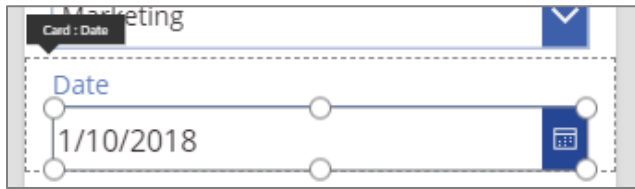
- a) Inspect the **DateTimeZone** property of the **DateValue1** control. Its current value should be **Local**.



- b) Update the **DateTimeZone** property to a value of **DateTimeZone.UTC** as shown in the following screenshot.



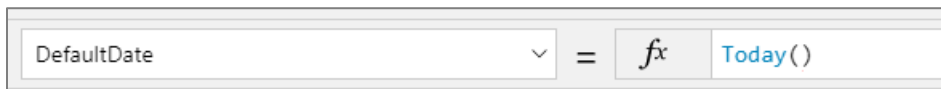
- c) All the dates displayed on the edit screen should now move ahead by one day and display their proper value.



The problem with date values in the Local format is that they are offset by the difference between Greenwich Mean Time and your local time zone. For example, if you are in Eastern Daylight Time (EDT), the date of **January 10, 2018** is displayed with a 5-hour offset which is **January 9, 2018 at 7:00 PM**. By setting the **DataTimeZone** property to **UTC**, you are effectively removing the offset and the dates are displayed more accurately.

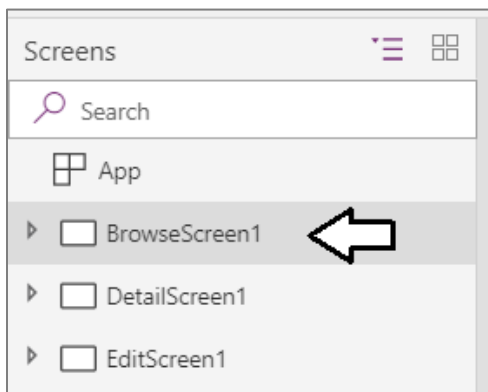
10. Configure the current day as the default value for **DateValue1**.

- Make sure the **DateValue1** control is selected.
- Inspect the **DefaultDate** property value for **DateValue1**.
- Update the **DefaultDate** property using the **Today()** function as shown in the following screenshot.



11. Test out the app by starting it and adding a new expense.

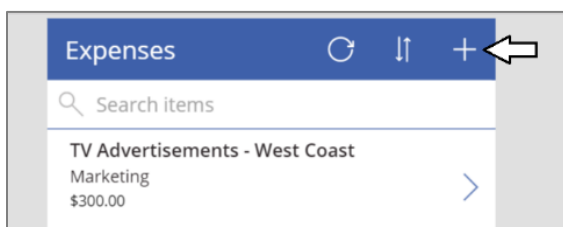
- a) Before starting the app, navigate to the screen named **BrowseScreen1**.



- b) Click the Start button with the arrow icon to launch the app for testing.



- c) When the browse screen appears, click to button with the **+** icon to add a new expense.



- d) You should now see the edit form into which you can enter a new expense.

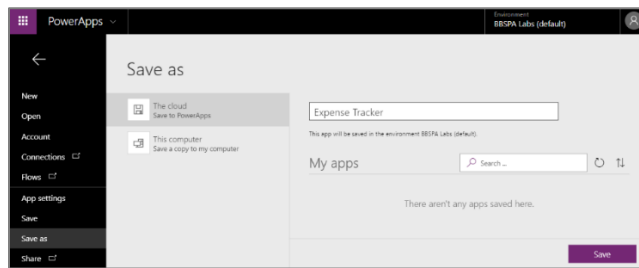
- e) Fill in the edit form for the new expense using the data shown in the following screenshot and then click the button with the checkmark icon in the upper right to save your work.

- f) Once you have saved the new expense, you should be able to see it in the browse screen.

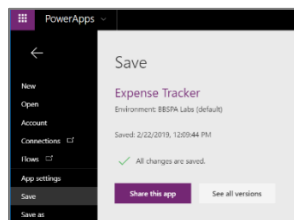
12. Save the app to the cloud.

- a) Drop down the **File** menu and click the **App settings** link.
b) Name the app **Expense Tracker** and assign a color, icon and description as shown in the following screenshot.

- c) Click the **Save** link in the left navigation and then click the **Save** button in the lower, right-hand side of the screen.

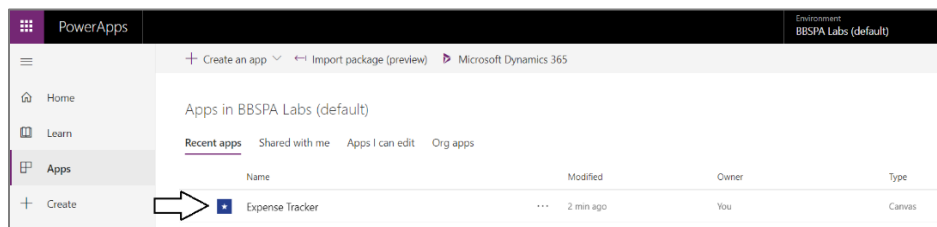


- d) You should be able to confirm that your app has been saved.

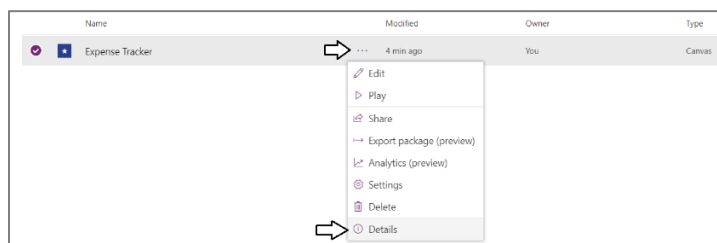


13. Examine the details of the new app.

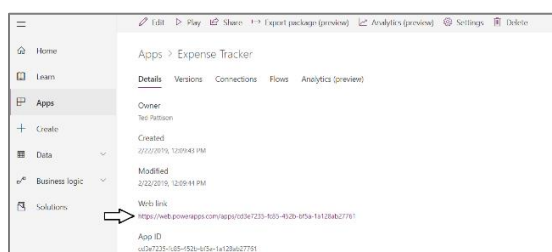
- a) Return the Power Apps portal at <https://web.powerapps.com> and click the **Apps** link.
b) Locate and the new app named **Expense Tracker**.



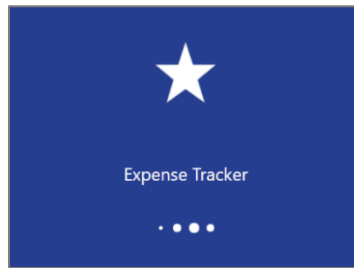
- c) Click on the ellipse (...) dropdown menu to the right of **Expense Tracker** app and select **Details**.



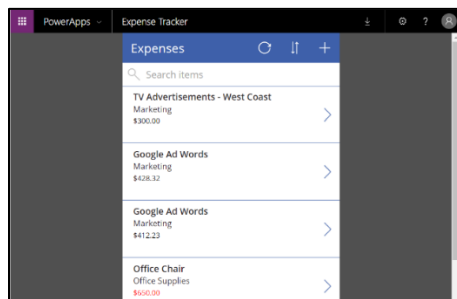
- d) On the app **Details** page, locate the **Web link** and click on it to launch the app.



- e) The app should start up when you click that Web link.



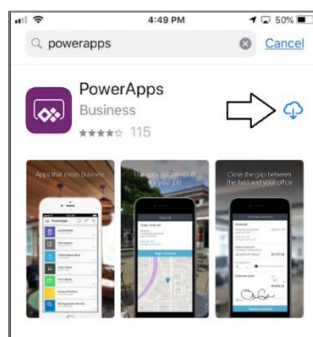
- f) The app should now start up in the usual run mode for end users.



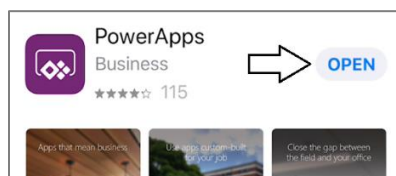
Exercise 2: Test the Expense Tracker Application From a Mobile Device

In this exercise you will create begin by installing the native Power Apps app on your mobile device. If you have already installed the Power Apps app on your mobile device, you can skip step 1 and move to step 2 of this exercise. If you do not have a mobile device that supports the Power Apps native app, you can skip the entire exercise and move on to Exercise 5.

1. Install the Power Apps native app on your mobile phone. This lab assumes you have a mobile phone which supports the Power Apps native app which include iPhones, Android phones and Windows 10 phones.
 - a) Navigate to the App Store for your mobile device and search for Power Apps.
 - b) Locate and installed the Power Apps app on your mobile device.



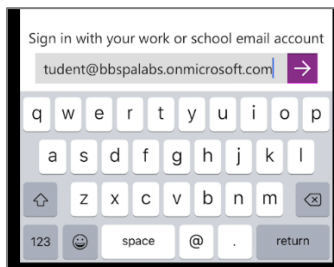
- c) Once the Power Apps app has been installed, open it.



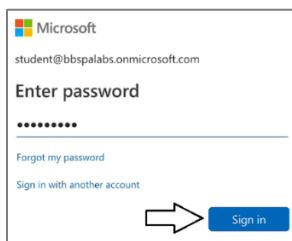
- d) You should now see the app welcome screen..



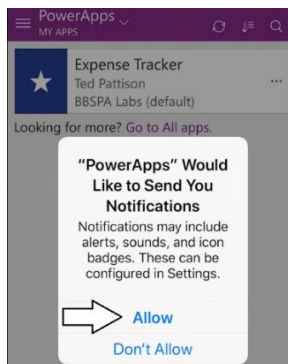
2. Sign into the Power Apps native app using the primary user account for the tenant you created at the start of this lab.
- a) Launch the Power Apps native app if it is not already running.
 - b) Enter the user name (i.e. email address) of the primary user account for your new Microsoft 365 tenant.
 - c) Click the purple button with the arrow to begin the sign in process.



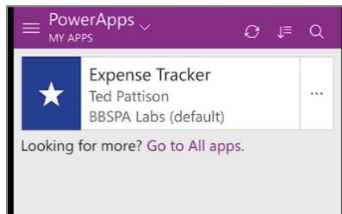
- d) When prompted, enter your password and click **Sign in**.



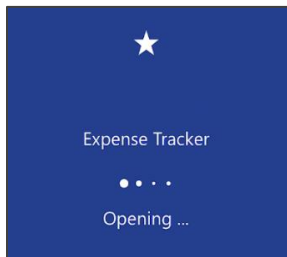
- e) If it's the first time you have run this app, you will be prompted to allow the app to send you notifications. Click **Allow**.



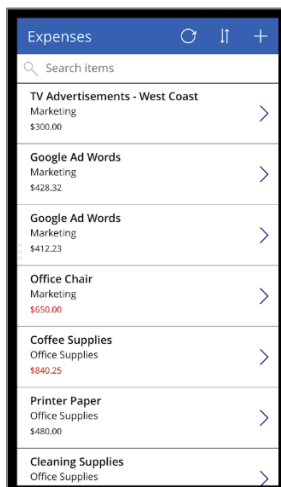
- f) Once you have logged in, you should be able to see the Expense Tracker canvas app in the list of apps.



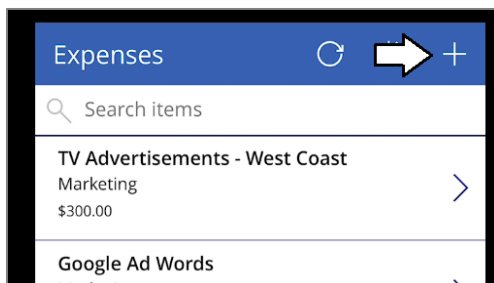
3. Open the Expense Tracker canvas app on your mobile device.
- Touch the Expense Tracker app in the apps list to launch it.
 - You should see that app starting up.



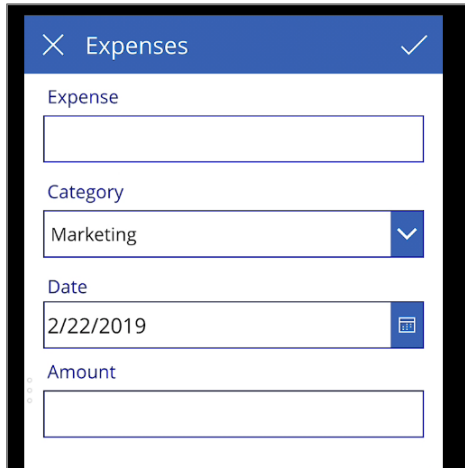
- c) When the app opens, you should be able to view existing expenses.



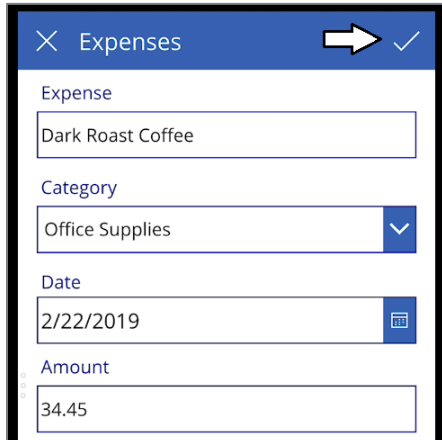
4. Use the mobile app to enter a new expense.
- Touch the button with the + sign to add a new expense.



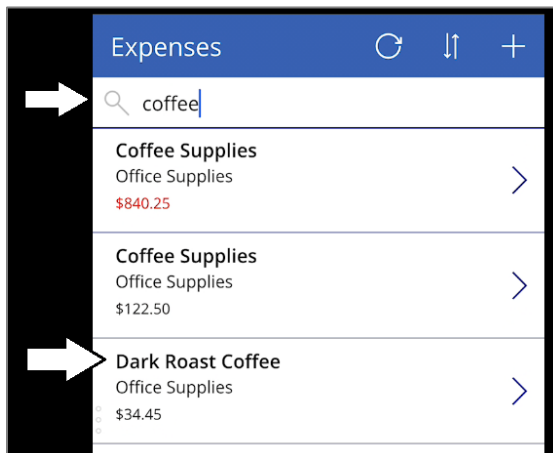
- b) You should now see the screen for adding a new expense.



- c) Add new expense data as shown in the following screenshot and touch the button with the checkmark to save your changes.



- d) Once the expense has been saved, search for coffee and verify you can see the new expense.



The purpose of this exercise is to ensure you can test canvas apps on your mobile device. Once you have successfully added a new expense, you can close the app on your mobile device and move ahead to the next exercise.

Exercise 3: Create a Canvas App using the Start from Blank Template

In this exercise you will create a new canvas app using **Canvas app from blank** template. This will give you experience creating the data binding for a canvas app without having to resort to using the **Start from data** template. In this lab you will also learn how to use the **Filter** function and the **Sort** function to manipulate data in a table. By the end, you will also work through the problems associated with a data source such as a table in an Excel workbook that does not support delegation.

1. Inspect the Excel workbook named **Customers.xlsx** located in the Student folder.
 - a) Using Windows Explorer, verify that there is an Excel workbook file named **Expenses.xlsx** located at the following path.

C:\Student\Modules\08_PowerPlatform\Lab\Customers.xlsx

- b) If you have Excel installed, open up this workbook and inspect what's inside.

CustomerId	FirstName	LastName	Company	EmailAddress	WorkPhone	HomePhone	Address
1	Nina	Diaz	Dunder Mifflin	Nina.Diaz@DunderMifflin.com	1(707)222-2222	1(707)666-4444	24 Route 41
2	Melinda	Carter	Vandelay Industries	Melinda.Carter@VandelayIndustries.com	1(707)111-6666	1(707)777-4444	405 Green Street
3	Pam	Miller	Brown Streak Railroad	Pam.Miller@BrownStreakRailroad.com	1(707)666-3333	1(707)222-0000	60 Mulberry Court
4	Merle	Blackwell	Krusty Burger	Merle.Blackwell@KrustyBurger.com	1(916)222-0000	1(916)333-2222	710 Ann Street
5	Ariel	Hale	ComTron	Ariel.Hale@ComTron.com	1(916)888-8888	1(916)444-3333	70 Walnut Street
6	Randy	Carter	ComTron	Randy.Carter@ComTron.com	1(916)666-1111	1(916)111-4444	9577 Mulberry Street
7	Lillie	Hinton	Blaid Durg Nuclear Power Plant	Lillie.Hinton@BlaidDurgNuclearPowerPlant.com	1(707)111-1111	1(707)333-0000	43 Lexington Court
8	Ladonna	Moody	Porter Automobiles	Ladonna.Moody@PorterAutomobiles.com	1(707)888-2222	1(707)888-7777	17 Warren Avenue
9	Buddy	McKay	InGen	Buddy.McKay@InGen.com	1(541)222-2222	1(541)555-5555	1364 Broad Street West
10	Warren	Sykes	Cyberdyne Systems	Warren.Sykes@CyberdyneSystems.com	1(916)444-5555	1(916)444-1111	94 Street Road
11	Jan	Rutledge	Umbrella Corporation	Jan.Rutledge@UmbrellaCorporation.com	1(503)333-2222	1(503)777-5555	6648 Hillside Avenue
12	Dallas	Lester	Strickland Propane	Dallas.Lester@StricklandPropane.com	1(541)111-7777	1(541)333-4444	507 Rose Street

- c) Scroll down until you reach the rows with the **CustomerId** starting with 500.

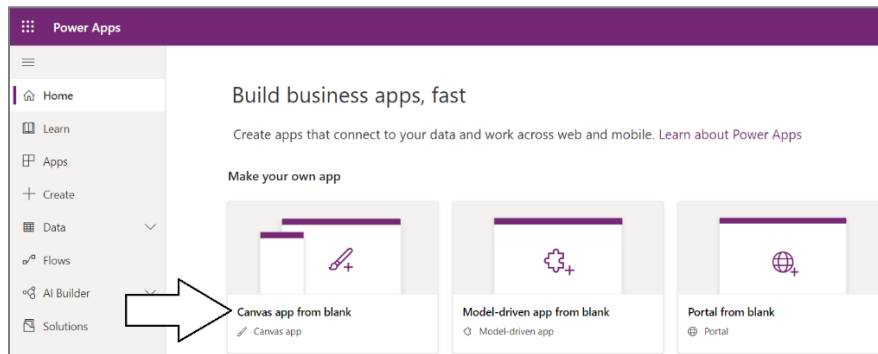
CustomerId	FirstName	LastName	Company	EmailAddress
501	Trent	Dixon	Michael Scott Paper Company	Trent.Dixon@
502	Saul	Snider	Biffco	Saul.Snider@B
503	Arron	Barrera	Groovy Smoothie	Arron.Barrera
504	Ezra	Bradley	Deon International	Ezra.Bradley@
505	Wilbur	Cote	Cyberdyne Systems	Wilbur.Cote@
506	Kelley	Gray	Total Bastard Airlines	Kelley.Gray@

A key learning point in this exercise is gain an understanding about the behavior of data sources that do not support delegation. When a connector does not support delegation, it will only return a maximum of 500 records by default. Therefore, you will not see any customer records with a **CustomerId** over 500. Later in this lab you will search for customers with the last name "Barrera". What you will see later in this exercise is that you will not be able to find this customer with the default canvas app settings which limits the tables it returns to a maximum of 500 records.

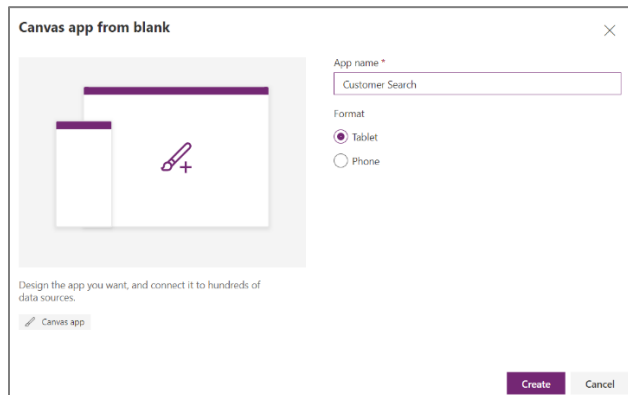
- d) Close the Excel workbook named **Customers.xlsx** without saving any changes and then close Microsoft Excel.
2. Upload the Excel workbook named **Customers.xlsx** to OneDrive for Business.
 - a) Drop down the Microsoft 365 app launcher menu and select **OneDrive** to navigate to your **Files** collection.
 - b) Click the **Upload** button and then select **Customers.xlsx** to upload this file to OneDrive for Business.
 - c) Verify that **Customers.xlsx** has been uploaded to your **Files** folder.

Name	Modified	Modified By	File Size	Sharing
Customers.xlsx	A few seconds ago	Ted Pattison	148 KB	Private
Expenses.xlsx	4 hours ago	Ted Pattison	12.9 KB	Private

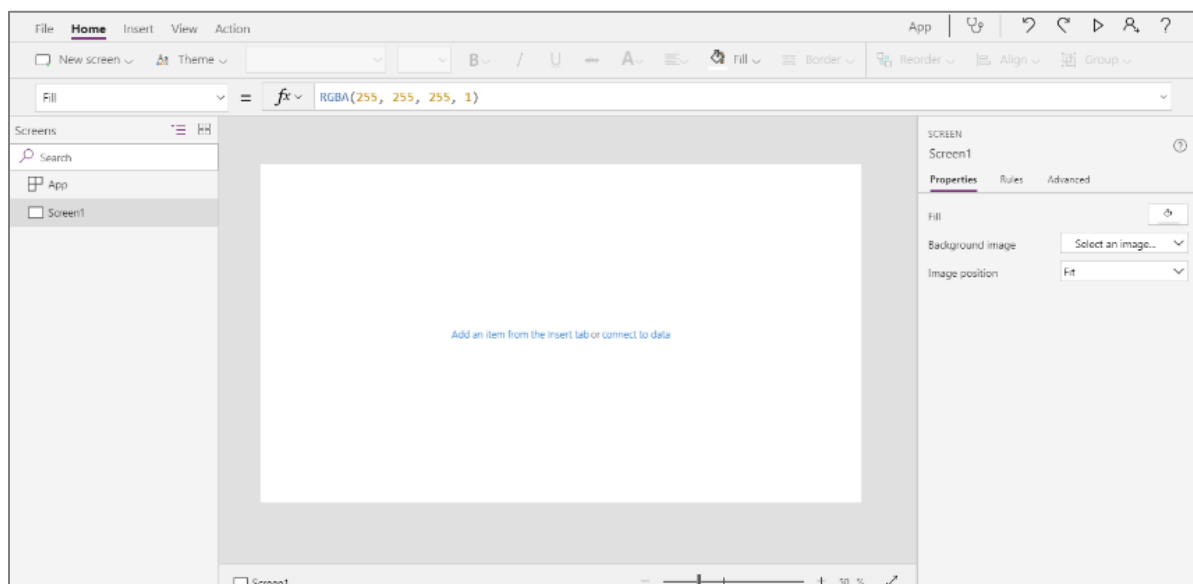
3. Create a new canvas app using the **Canvas app from blank** data template.
 - a) Navigate to the Power Apps portal at <https://make.powerapps.com>
 - b) Create a new canvas app by clicking the **Canvas app from blank** button.



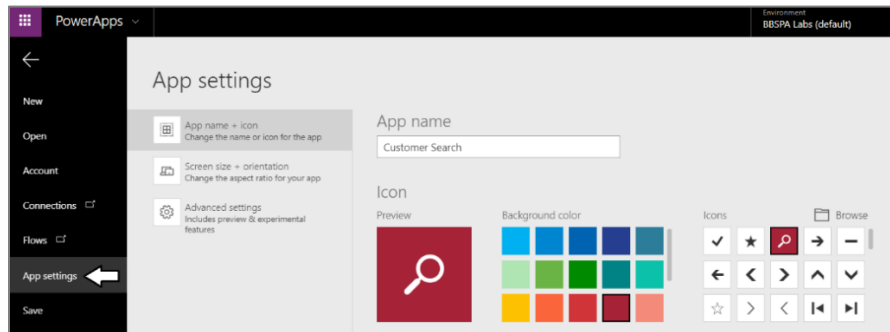
- c) When prompted with the Canvas app from blank dialog, enter an **App name** of **Customer Search**.
 - d) Make sure to select a **Format** of the **Tablet** and then click **Create**.



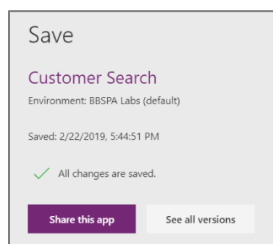
- e) After a few seconds, you should see your new canvas app in Power Apps Studio.



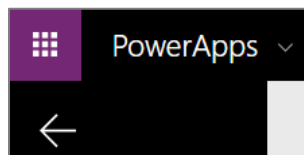
4. Configure the App settings and save your new app.
 - a) Click the **File** menu and then click **App settings**.
 - b) Make sure the **App name** is set to **Customer Search**.
 - c) Select an icon and color of your choosing as shown in the following screenshot.



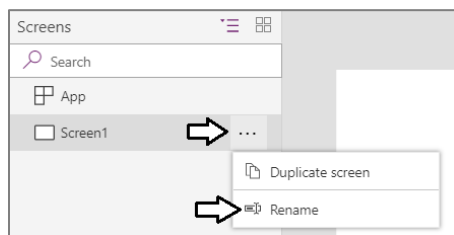
- d) Click Save in the left navigation and then click the Save button in the bottom right corner of the Power Apps Studio window.



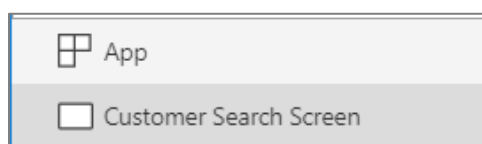
- e) Click the back arrow at the top left corner of Power Apps Studio to return to the canvas app in edit mode.



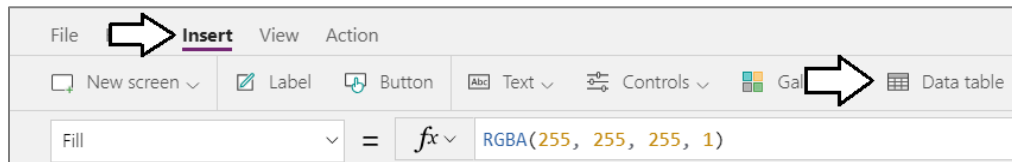
5. Update the name of **Screen1** to **Customer Search Screen**.
 - a) Drop down the ellipse menu (...) for **Screen1** and select the **Rename** command.



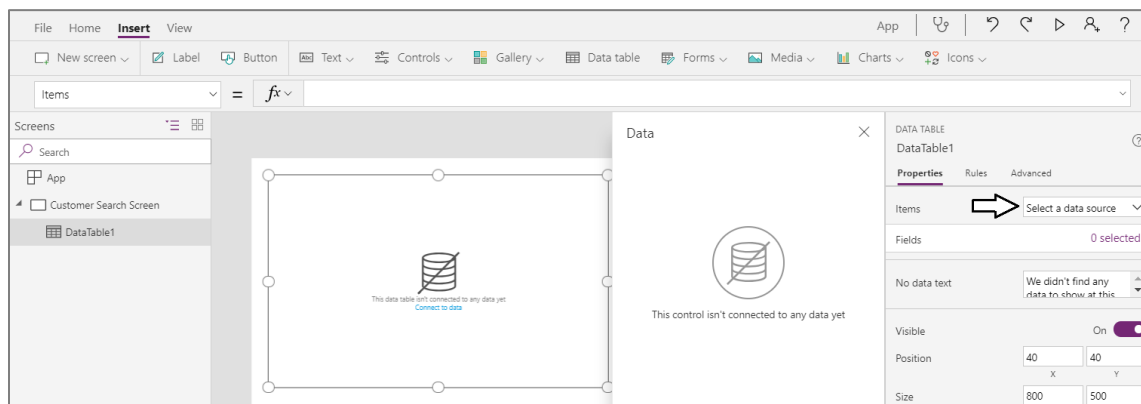
- b) Rename the screen to **Customer Search Screen**.



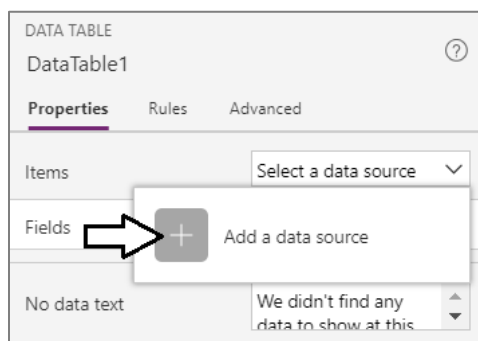
6. Add a new **Data table** control to display customer data.
 - a) Navigate to the **Insert** tab in the ribbon.
 - b) Click the **Data table** button to add a new Data table control to **Customer Search Screen**.



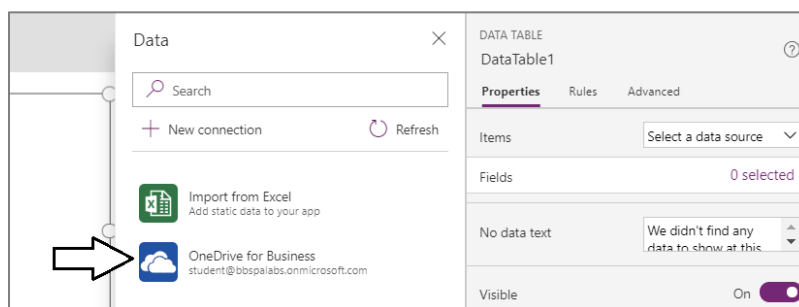
- c) Once the Data table control has been added, inspect its properties in the Properties pan on the right.
 - d) Locate the **Items** property which has a dropdown menu with the caption **Select a data source**.



- e) Drop down the menu for the **Items** property and select **Add a data source** to display the **Data** pane.

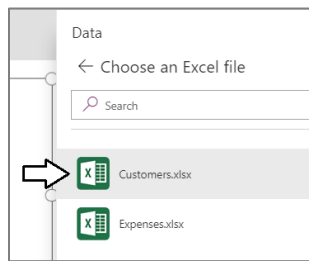


- f) When the **Data** pane appears, select the connection named **OneDrive for Business**.

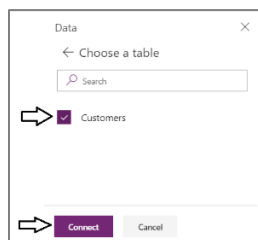


The **OneDrive for Business** connection should automatically appear because you created it when building the **Expense Tracker** app.

- g) The **Data** pane should prompt you to **Choose an Excel file** and display the Excel workbooks at your OneDrive root library.
- h) Select the Excel workbook named **Customers.xlsx**.

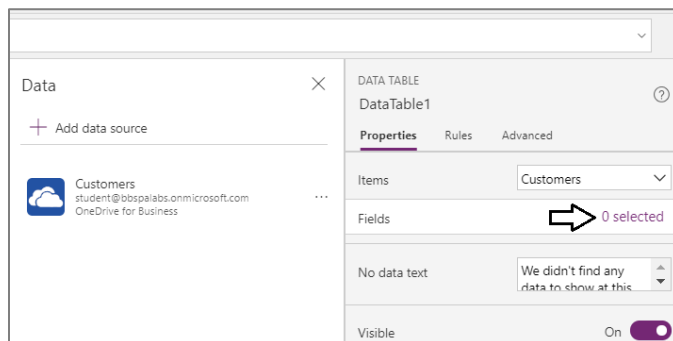


- i) When prompted to **Choose a table**, select the **Customers** table and then click **Connect**.

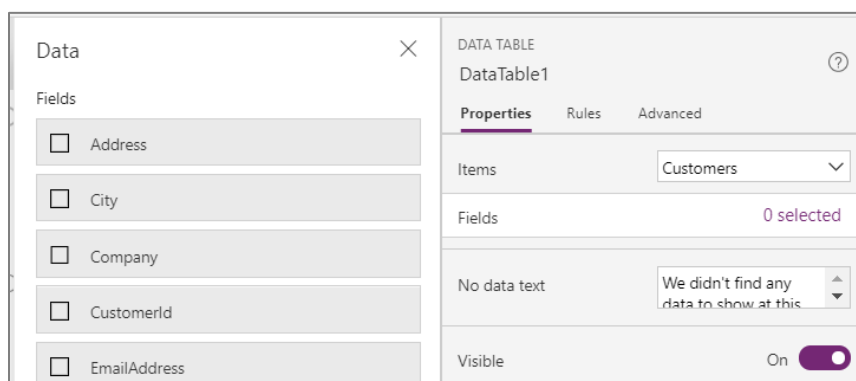


At this point, you have bound the **Customers** table to the Data table control but you have not selected any fields to display.

- j) Inspect the **Items** property and the **Fields** property of the Data table control.
- k) You should be able to verify that **Items** property has a value of **Customers** and the **Fields** property shows **0 selected**.



- l) Click the link for the **Fields** property with the caption **0 selected** to display the **Fields** list in the **Data** pane.



- m) Select the fields **CustomerId**, **FirstName**, **LastName**, **Company**, **EmailAddress**, **WorkPhone** and **HomePhone**.

- n) You should now see those fields appear in the Data table control.
o) Using the mouse, resize the Data table to take up an entire height and width of the screen.
p) Move the top of the Data table down leaving a small amount of empty space at the top as shown in the following screenshot.

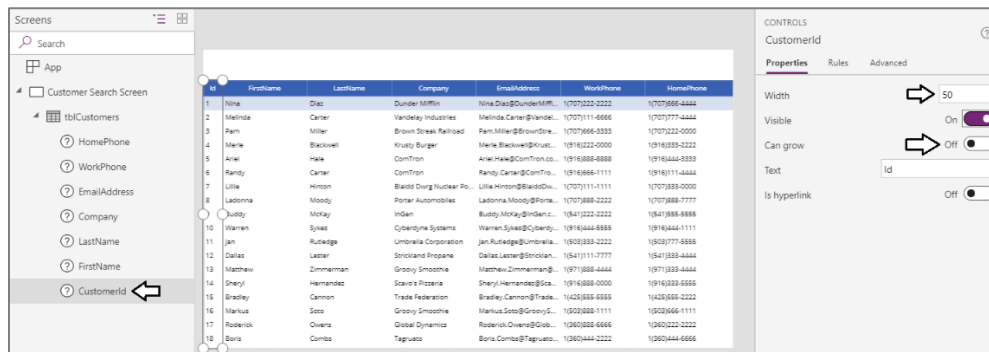
CustomerId	FirstName	LastName	Company	EmailAddress	WorkPhone	HomePhone
1	Nina	Diaz	Dunder Mifflin	Nina.Diaz@Dunder...	1(707)222-2222	1(707)666-4444
2	Melinda	Carter	Vandelay Industries	Melinda.Carter@Van...	1(707)111-6666	1(707)777-4444
3	Pam	Miller	Brown Streak Railroad	Pam.Miller@BrownS...	1(707)666-3333	1(707)222-0000
4	Merle	Blackwell	Krusty Burger	Merle.Blackwell@Kr...	1(916)222-0000	1(916)333-2222
5	Ariel	Hale	ComTron	Ariel.Hale@ComTro...	1(916)888-8888	1(916)444-3333
6	Randy	Carter	ComTron	Randy.Carter@Com...	1(916)666-1111	1(916)111-4444
7	Lillie	Hinton	Blaid Dwg Nuclear...	Lillie.Hinton@Blaid...	1(707)111-1111	1(707)333-0000
8	Ladonna	Moody	Porter Automobiles	Ladonna.Moody@Po...	1(707)888-2222	1(707)888-7777

- q) Rename the Data table control to **tblCustomers**.

- r) Select the **CustomerId** column and then look at this selection in the property pane on the right.

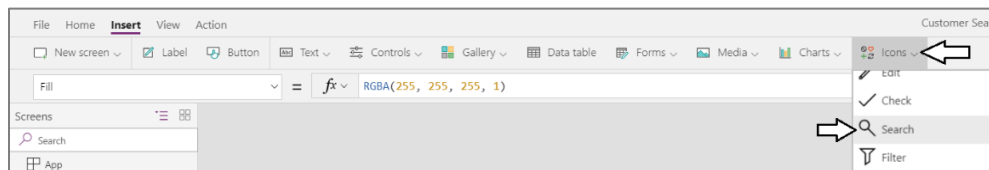
CustomerId	FirstName	LastName	Company	EmailAddress	WorkPhone	HomePhone
1	Nina	Diaz	Dunder Mifflin	Nina.Diaz@Dunder...	1(707)222-2222	1(707)666-4444
2	Melinda	Carter	Vandelay Industries	Melinda.Carter@Van...	1(707)111-6666	1(707)777-4444
3	Pam	Miller	Brown Streak Railroad	Pam.Miller@BrownS...	1(707)666-3333	1(707)222-0000
4	Merle	Blackwell	Krusty Burger	Merle.Blackwell@Kr...	1(916)222-0000	1(916)333-2222
5	Ariel	Hale	ComTron	Ariel.Hale@ComTro...	1(916)888-8888	1(916)444-3333
6	Randy	Carter	ComTron	Randy.Carter@Com...	1(916)666-1111	1(916)111-4444
7	Lillie	Hinton	Blaid Dwg Nuclear...	Lillie.Hinton@Blaid...	1(707)111-1111	1(707)333-0000
8	Ladonna	Moody	Porter Automobiles	Ladonna.Moody@Po...	1(707)888-2222	1(707)888-7777
9	Buddy	McKay	InGen	Buddy.McKay@InGe...	1(541)222-2222	1(541)555-5555
10	Warren	Sykes	Cyberdyne Systems	Warren.Sykes@Cybe...	1(916)444-5555	1(916)444-1111

- s) Set the **Width** property of the **CustomerId** column to **50** and set the **Can grow** property to **Off**.

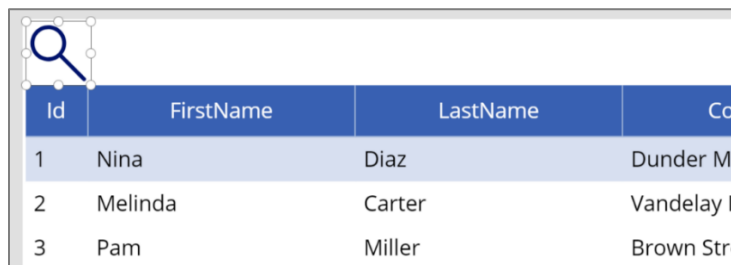


7. Add a search box

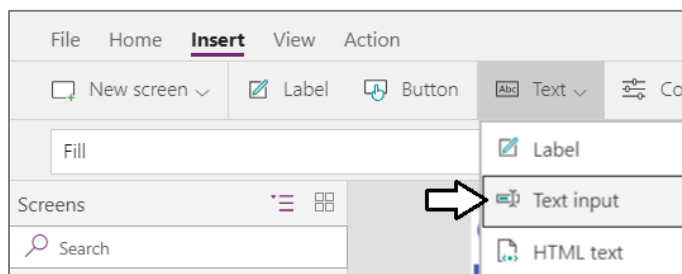
- a) Add a search icon onto the **Customer Search Screen**. and rename it to **icoSearch**.



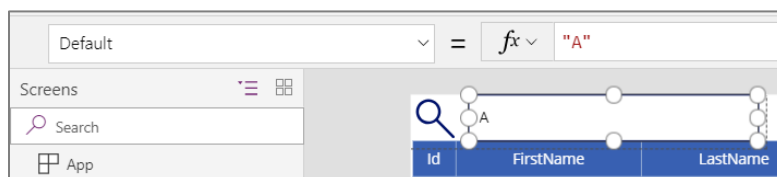
- b) Position **icoSearch** in the upper, left as shown in the following screenshot.



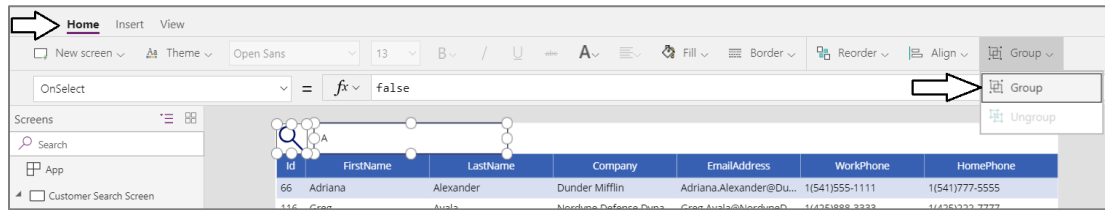
- c) Add a **Text input** control to the **Customer Search Screen** and rename it to **txtSearchInput**.



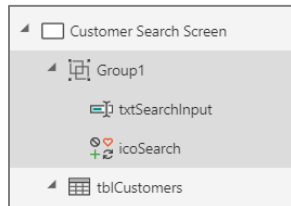
- d) Set the **Default** property of **txtSearchInput** to **"A"**.



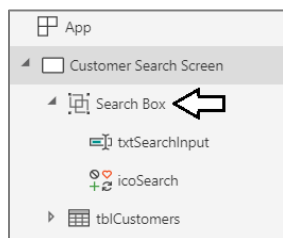
- e) Select **icoSearch** and **txtSearchInput** together at the same time and group them using the **Group > Group** command.



- f) You should have created a new group named **Group1**.

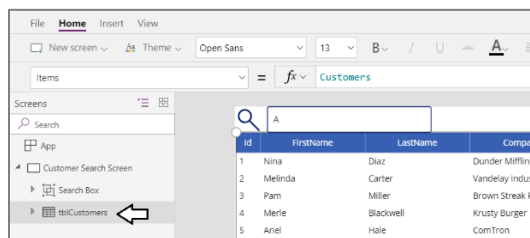


- g) Rename **Group1** to **Search Box**.



8. Add filtering and sorting behavior.

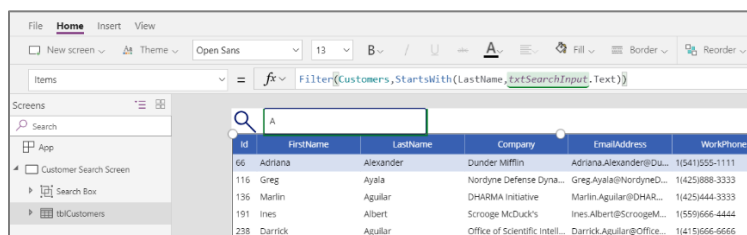
- a) Inspect the **Items** property of **tblCustomers** which should currently have a value of **Customers**.



- b) Modify the value for the **Items** property of **tblCustomers** using the following expression to add filtering behavior.

```
Filter( Customers, StartsWith(LastName, txtSearchInput.Text))
```

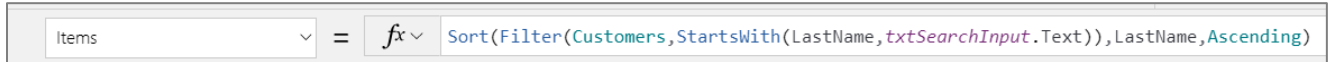
- c) The items displayed by **tblCustomers** should now only include customers whose last name starts with "A".



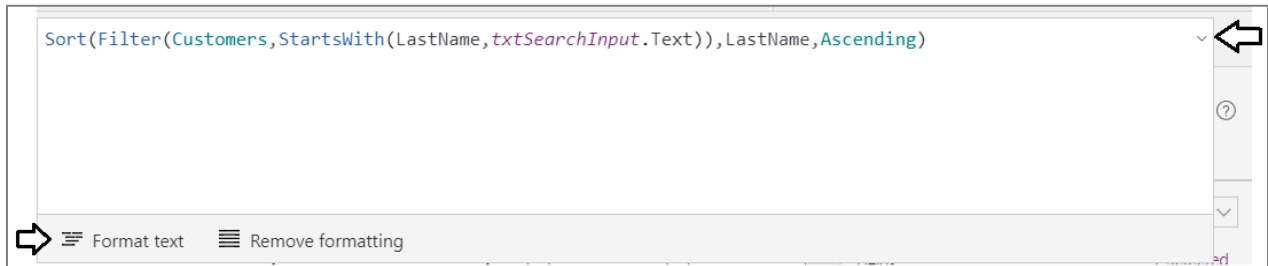
- d) Modify the value for the **Items** property of **tblCustomers** using the following expression to add sorting behavior.

```
Sort(Filter(Customers,StartsWith(LastName,txtSearchInput.Text)),LastName,Ascending)
```

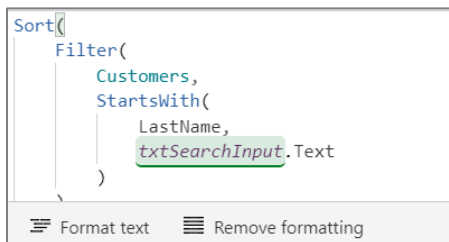
- e) Your formula bar should match the following screenshot.



- f) Experiment with formula formatting by dropping down the formula bar vertically and clicking the **Format text** button.



- g) You should see that Power Apps Studio is able to format expressions to make them more readable.



9. Test out the filter functionality

- a) Start the app and try running a search using a search input string of "Bar". You should see the following results.

Search: Bar			
Id			
FirstName			
LastName			
Company			
172	Mable	Barber	Yoyodyne Propulsion S...
101	Sondra	Barker	Ewing Oil
299	Gustavo	Barker	Astromech
446	Jerry	Barker	Dinoco
216	Jermaine	Barlow	Groovy Smoothie
267	Jessica	Barnes	Metacortex
96	Bobbie	Barton	Dirk Gently's Holistic D...
254	Eduardo	Barton	ARCAM Corporation

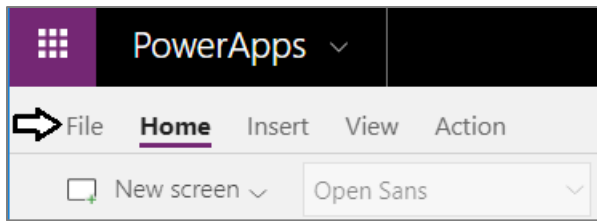
- b) Search for "Barr" to find customers with the last name Barrera. Note that you cannot find any customers with this last name.

Search: Barr					
Id					
FirstName					
LastName					
Company					
EmailAddress					
WorkPhone					
We didn't find any data to show at this time					

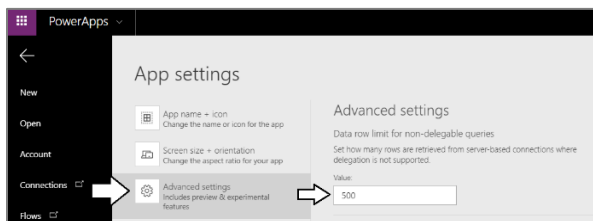
You are only searching through the first 500 records because of the way delegation works in Power Apps. You will never find any records that exist after the first 500. The key point is that you can get into trouble if you don't understand how delegation works.

10. Configure the **Customer Search** app to discover records beyond the first 500 record limitation.

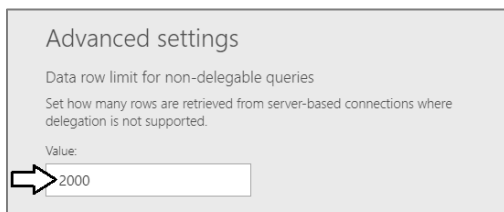
- a) Navigate to Backstage area in Power Apps Studio by clicking the **File** menu.



- b) Click **App settings** in the left navigations and then click **Advanced settings**.
c) The **Data row limit for non-delegable queries** setting should have the default value of **500**.



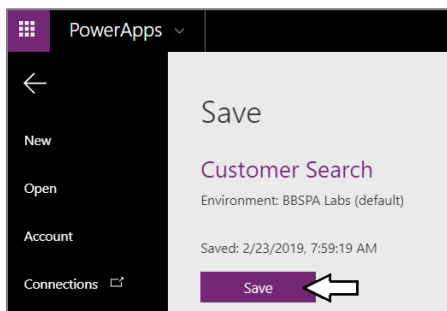
- d) Modify the value for the **Data row limit for non-delegable queries** setting to the maximum value of **2000**.



- e) Click File to leave the Backstage area.
f) Start up the app and run a search for **Barrera**.
g) You should now find **Barrera** because you are looking through all the records in the underlying Excel table.

Barrera						
	Id	FirstName	LastName	Company	EmailAddress	WorkPhone
	502	Arron	Barrera	Groovy Smoothie	Arron.Barrera@Groovy...	1(408)555-2222
	655	Irene	Barrera	Trioptimum Corporation	Irene.Barrera@Triopti...	1(949)111-1111
	919	Amado	Barrera	Prescott Pharmaceutic...	Amado.Barrera@Presc...	1(323)444-2222
	938	Shaun	Barrera	The Hanso Foundation	Shaun.Barrera@TheHa...	1(360)555-8888

- h) Quit the app from running and then save your changes to the app.



11. Student challenge (if you have time)

- a) Resize the columns in the Data table to make the data more readable.

Bar

	Id	FirstName	LastName	Company	EmailAddress	WorkPhone	HomePhone
	172	Mable	Barber	Yoyodyne Propulsion Systems	Mable.Barber@YoyodynePropulsionSystems.c...	1(206)111-5555	1(206)444-4444
	101	Sondra	Barker	Ewing Oil	Sondra.Barker@EwingOil.com	1(916)555-6666	1(916)888-8888
	299	Gustavo	Barker	Astromech	Gustavo.Barker@Astromech.com	1(425)777-4444	1(425)777-4444
	446	Jerry	Barker	Dinoco	Jerry.Barker@Dinoco.com	1(541)555-4444	1(541)666-5555
	10...	Daron	Barker	Zorg Industries	Daron.Barker@ZorgIndustries.com	1(707)444-3333	1(707)666-5555
	216	Jermaine	Barlow	Groovy Smoothie	Jermaine.Barlow@GroovySmoothie.com	1(503)222-6666	1(503)555-5555
	820	Chris	Barlow	Shelbyville Nuclear Power Pl...	Chris.Barlow@ShelbyvilleNuclearPowerPlant.c...	1(661)444-3333	1(661)666-7777
	14...	Angeline	Barlow	Fabrikam	Angeline.Barlow@Fabrikam.com	1(971)666-8888	1(971)555-8888
	367	Jessica	Barnes	Metacortex	Jessica.Barnes@Metacortex.com	1(559)666-8888	1(559)111-5555
	574	Shelby	Barnes	The Hanso Foundation	Shelby.Barnes@TheHansoFoundation.com	1(415)555-8888	1(415)666-6666
	725	Wes	Barnes	Roxxon	Wes.Barnes@Roxxon.com	1(949)777-4444	1(949)444-2222
	682	Sang	Barr	Wallaby Airlines	Sang.Barr@WallabyAirlines.com	1(509)666-3333	1(509)444-1111

Congratulations. You have now completed this lab.