Creating Approvals in SharePoint Online using Microsoft Flow

Lab Time: 60 minutes

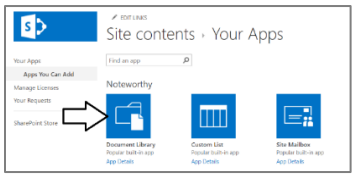
Lab Folder: C:\Student\Modules\02_Approvals\Lab

Lab Overview: In this lab you will begin by creating a canvas app named **Photo Tracker** that uploads photos to a SharePoint document library named **Submitted**. After that you will create a flow named Photo Approval which automates an approval process.

Exercise 1: Create Two SharePoint Document Libraries for Uploading Photos

In this exercise, you will create two document libraries in your SharePoint site to store photos.

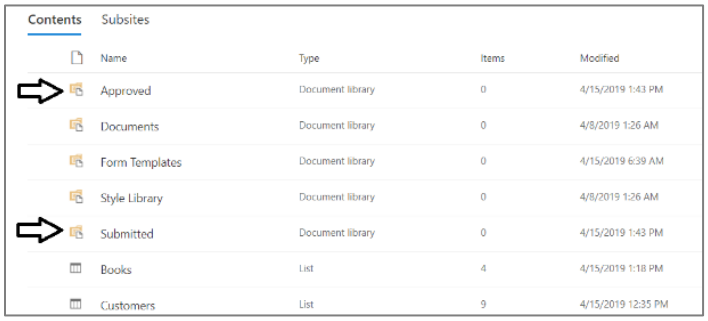
1. Navigate to the root SharePoint site for your trial Office 365 tenancy.
2. Create a document library named **Submitted**.
   a) Click on the gear icon and then click on **Add an app**.
   b) Click on the **Document Library** tile to create a new document library.



   c) In the **Adding a Document Library** dialog, enter a **Name** of **Submitted** and then click the **Create** button to create the library.

3. Create a document library named **Approved**.
   a) Click on the gear icon and then click on **Add an app**.
   b) Click on the **Document Library** tile to create a new document library.
   c) In the **Adding a Document Library** dialog, enter a **Name** of **Approved** and then click the **Create** button to create the library.

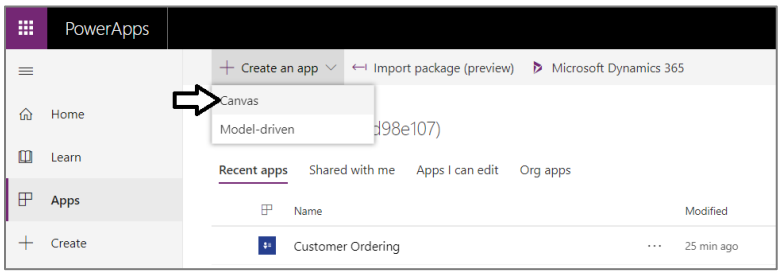
   d) You should now have two new document libraries named **Submitted** and **Approved**.



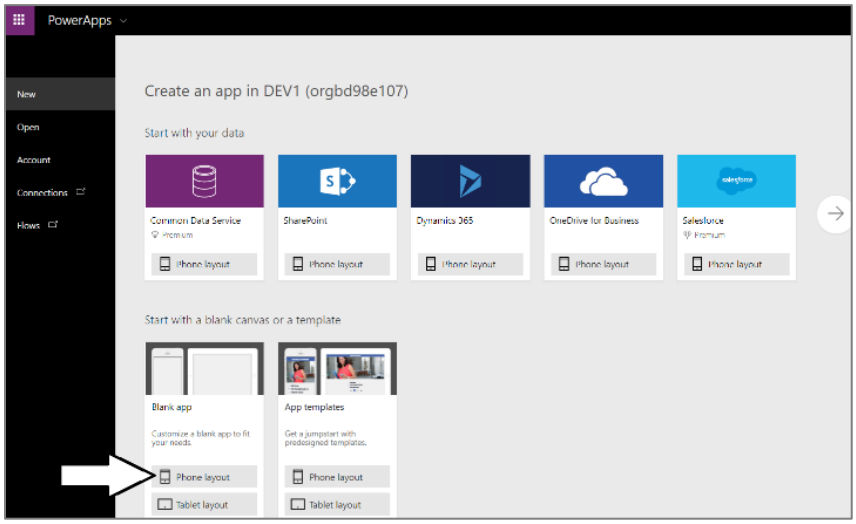
Exercise 2: Create a New Canvas App to Upload Photos to SharePoint

In this exercise, you will create a new canvas app and a flow that work together to upload a photo to the **Submitted** document library.

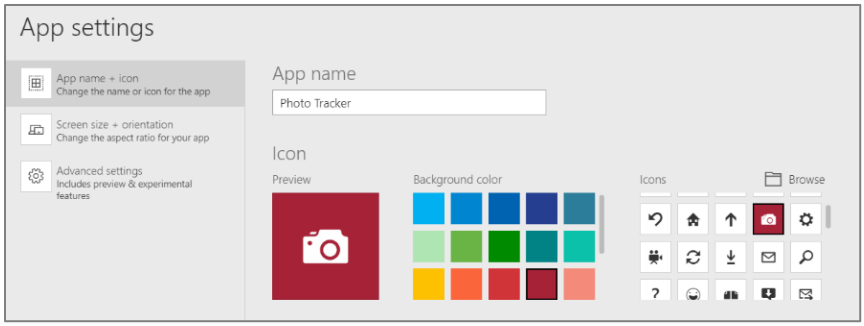
1. Create the Photo Tracker canvas app.
   a) Navigate to the PowerApps portal at https://web.PowerApps.com.
   b) Click the **Apps** link in the left navigation.
   c) Drop down the **Create an app** menu and click **Canvas** to create a new canvas app.



   d) On the **Create an app** page, select **Blank app > Phone layout** as shown in the following screenshot.



   e) Once the app has been created, click the **File** menu and **App settings** to get to the **App settings** page.
   f) On the **App settings** page, give the app a name of **Photo Tracker** and select an icon and color of your choosing.

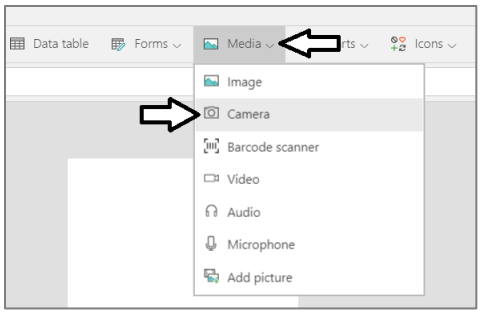


   g) Click **Save** in the left navigation and then click the **Save** button in the bottom right to save your changes to the app.
   h) Click the **File** menu to move back to the PowerApps Studio editor.

2. Given the screen a better name.
   a) The app has a single screen which has a default name of **Screen1**.

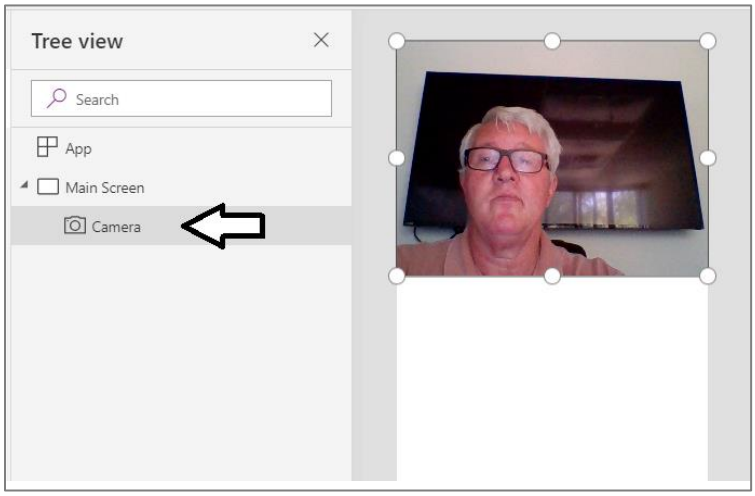
   b) Rename the screen to Main **Screen**.

3. Add a camera control to **Main Screen**.
   a) Click the **Insert** tab on the ribbon and then select **Media > Camera** to add a new camera control.



If you are working on a laptop computer or a desktop computer with a camera, the camera control should display what the camera is looking at. If you are working on a computer without a camera, you will not be able to see any image at all.

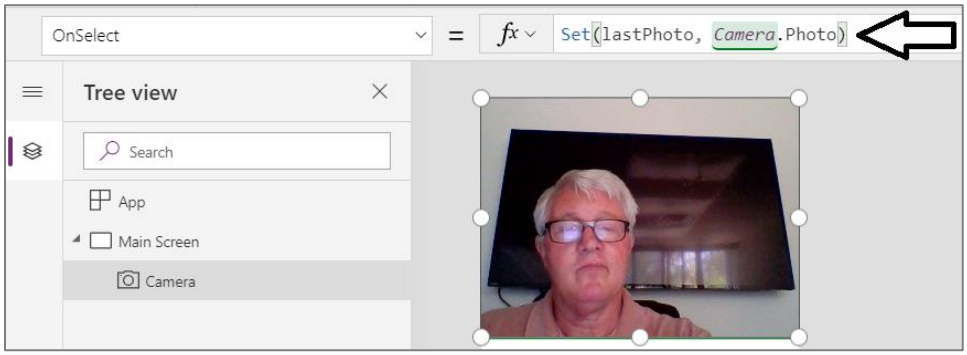
   b) Rename the camera control from **Camera1** to **Camera**. and reposition it to take up the entire width of the screen.



c) Update the **OnSelect** property of **Camera** with the following expression to save a photo into a global variable

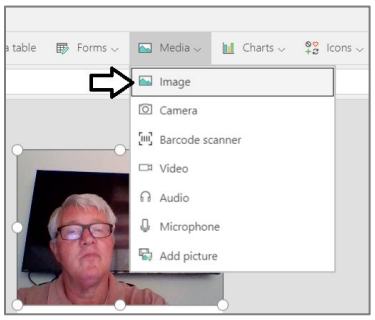
```
Set(lastPhoto, Camera.Photo)
```

d) The formula you have entered for **OnSelect** property for **Camera** should match the following screenshot.



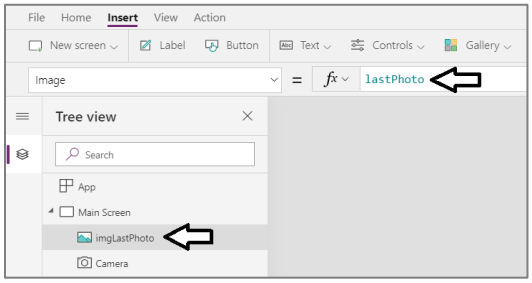
4. Add an image control to display the photo stored in the variable named **lastPhoto**.

a) From the **Insert** tab, select **Media > Image** to add a new **Image** control to the screen.



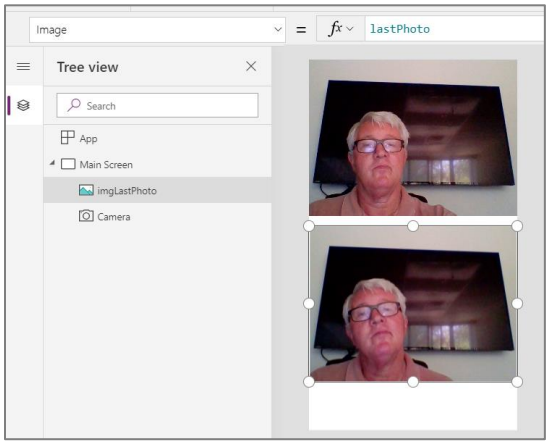
b) Rename the **Image** control to **imgLastPhoto**.

c) Set the **Image** property of **imgLastPhoto** to the variable named **lastPhoto**.

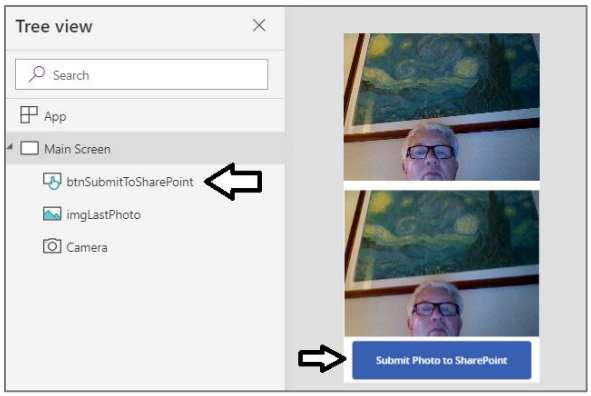


When you are using the camera control in PowerApps Studio, you must hold down the **Alt** key and then click on the **Camera** control to simulate taking a photo with a mobile device. When you hold down the **Alt** key and click the **Camera** control, it will also have the effect of evaluating the **OnSelect** property of the **Camera** control which will write a photo image into the variable named **lastPhoto**.

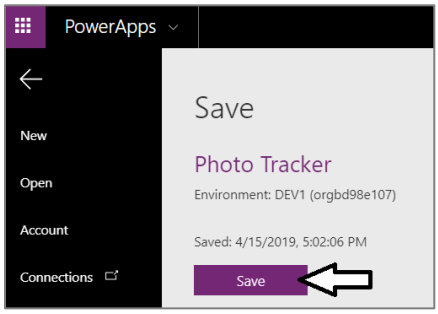
d) Hold down the **Alt** key and then click on the **Camera** control. When you do this, you should then see the photo image displayed below in the **imgLastPhoto**.



5. Add a new button to the screen which allows the user to save the last photo to SharePoint.
   a) Add a new button to the screen and rename it to **btnSubmitToSharePoint**.
   b) Update the Text property of the button to **Submit Photo to SharePoint**.

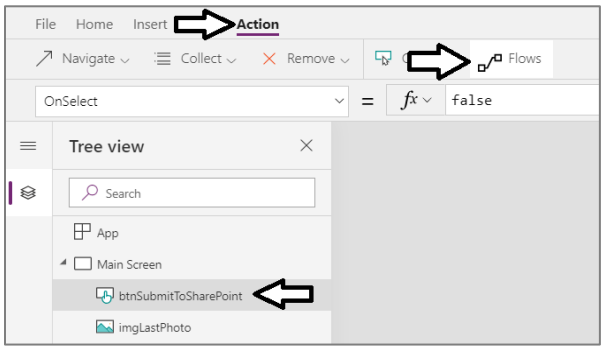


6. Save your work.
   a) Click the **File** menu and the **Save** button to save your work.

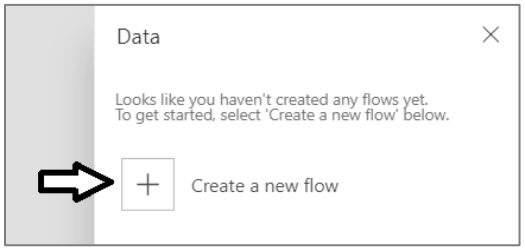


   b) Click the **File** menu again to return to the editor in PowerApps Studio.

7. Connect **btnSubmitToSharePoint** to a new flow.
   a) Select **btnSubmitToSharePoint** in the left tree view.
   b) Click the **Actions** tab and then click **Flows** as shown in the following screenshot.

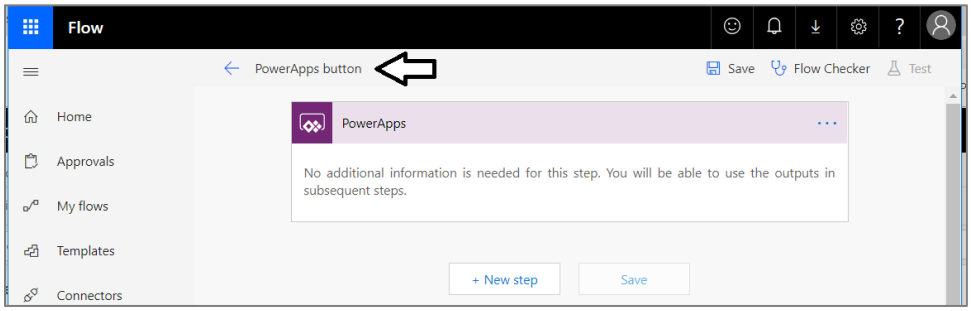


   c) On the **Data** pane. select the option to **Create a new flow**.

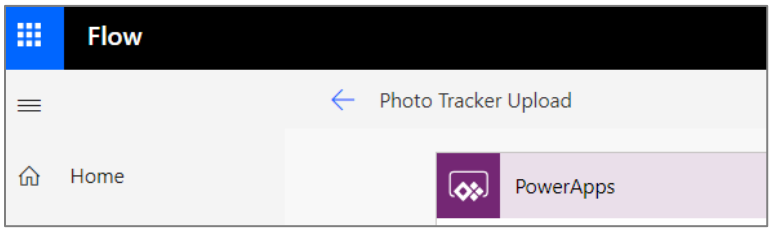


At this point, you will be redirected to Microsoft Flow in a separate browser tab so you can work on the new flow that has been created. You will first rename the flow that has been created. After that, you will implement the behavior in the flow to upload a photo to the SharePoint document library named **Submitted**. After you have implemented the flow and saved your changes, you will return to **Photo Tacker** canvas app and add support to execute the flow by clicking the **btnSubmitToSharePoint** button.

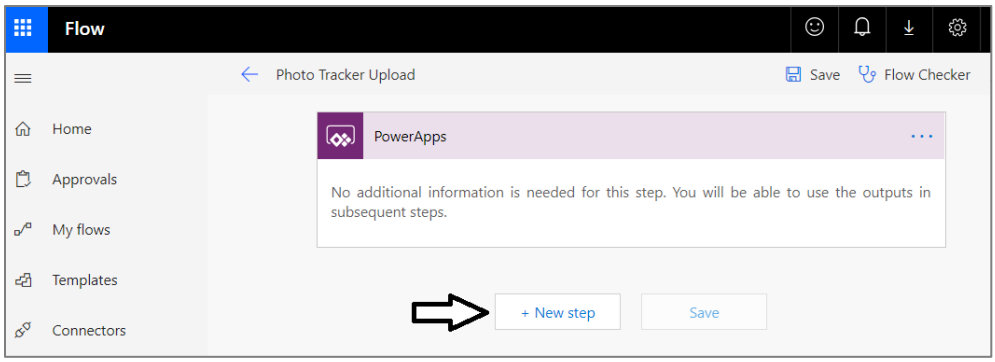
8. Change the name of the flow.
   a) You should be on a page with a new flow that has been given the name **PowerApps button**.



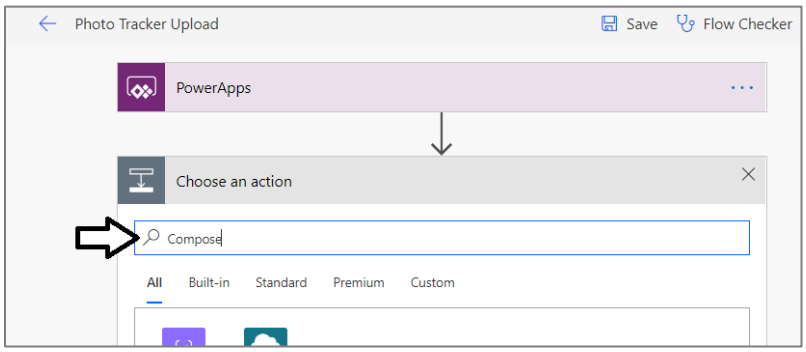
   b) Rename the flow from **PowerApps button** to **Photo Tracker Upload**.



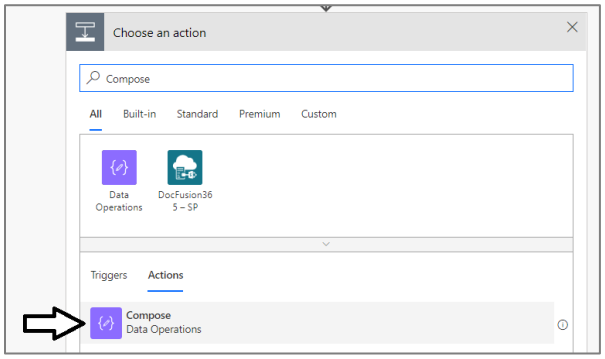
9. Add a new compose action to generate a unique file name for each photo.
   a) Click **New Step** to add a new step.



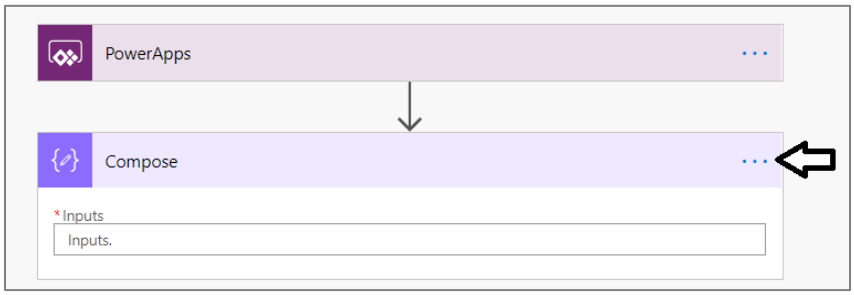
   b) Type **Compose** into the search box.



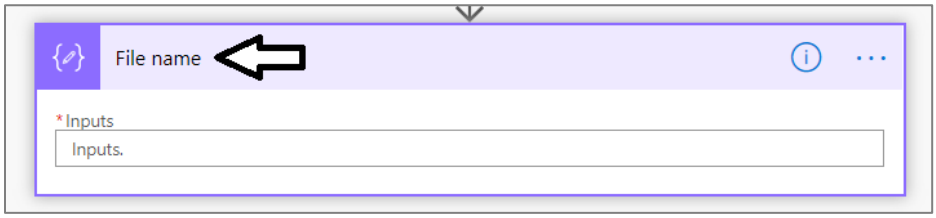
   c) Find and select the **Compose** action to add a new step.



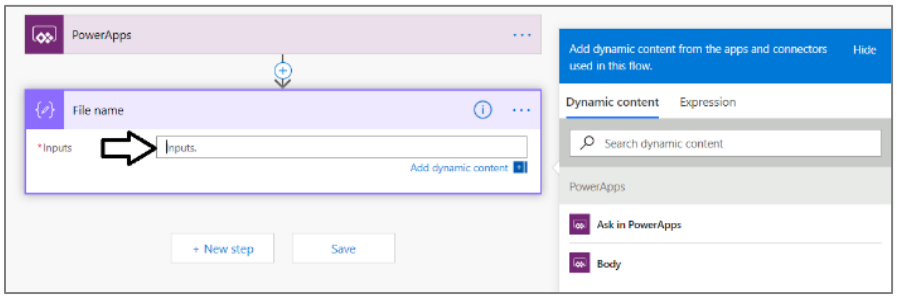
   d) Click the context menu (…) for the new **Compose** action and select the **Rename** command.



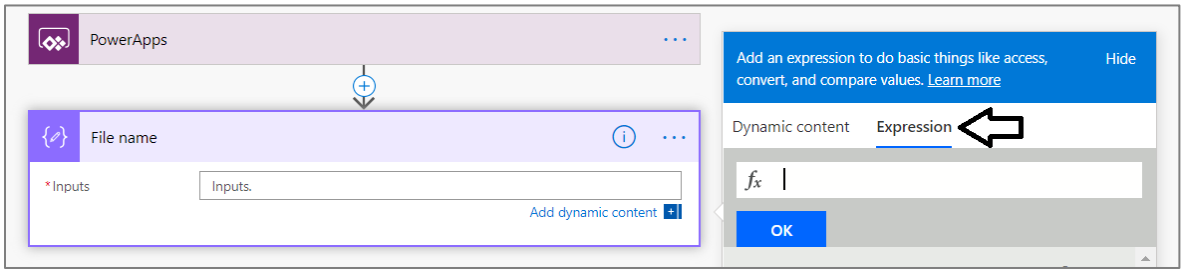
e) Give the **Compose** action the name **File name**.



f) Click in the **Inputs** textbox so you can modify its value.



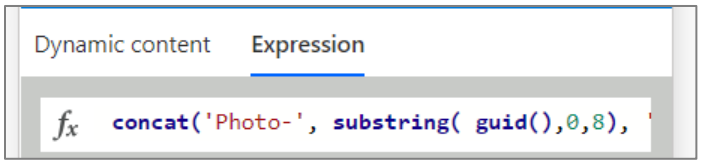
g) With the **Inputs** textbox selected, click **Expressions** link to the right so you can add an expression for the **Inputs** parameter.



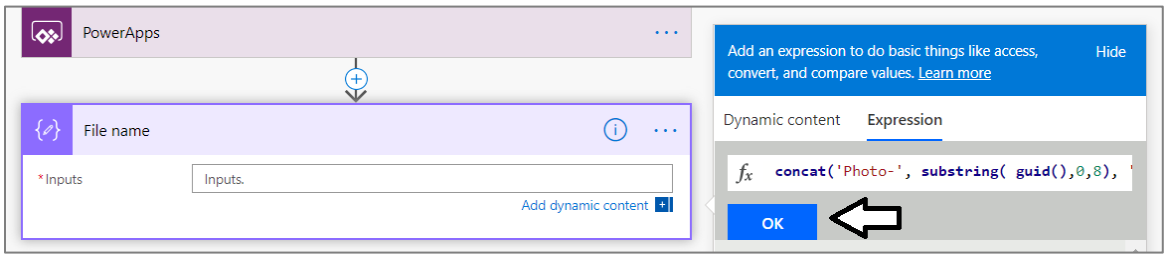
h) Add the following expression into the **Expressions** textbox .

```
concat('Photo-', substring( guid(),0,8), '.png')
```

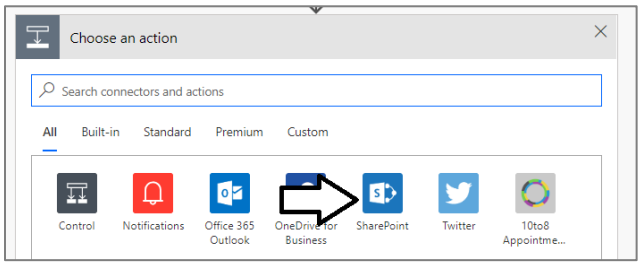
i) The expression you have entered in the **Expressions** textbox is should match the following screenshot.



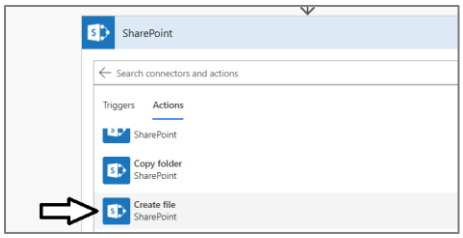
j) Click the **OK** button to save the expression for the **Inputs** parameter of the **File name** action.



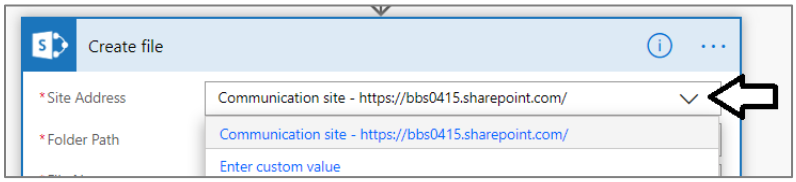
10. Add a SharePoint **Create file** action to upload photos to SharePoint.
    a) Click **New Step** to add a new step.
    b) Select **SharePoint** to filter the actions displayed below.



    c) Find and select the SharePoint **Create file** action.

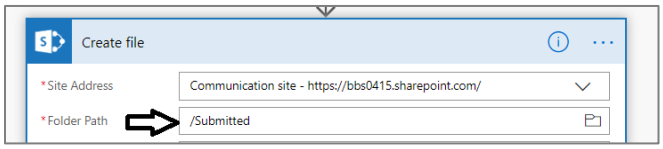


    d) Update the **Site Address** input parameter with the URL to your SharePoint site.

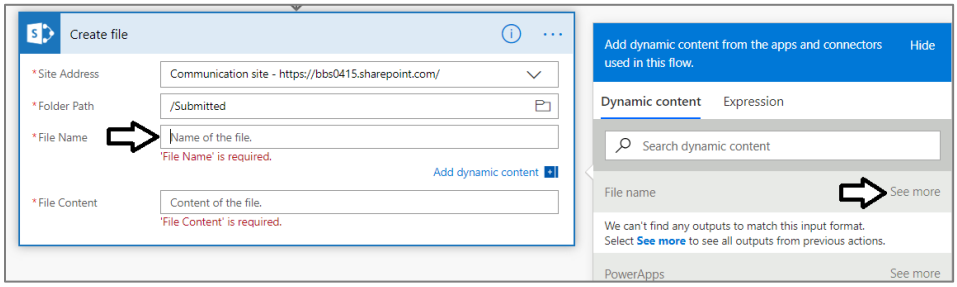


If the drop down does not automatically include your SharePoint site, you should select **Enter custom value** and then you must manually enter the URL to your SharePoint site.

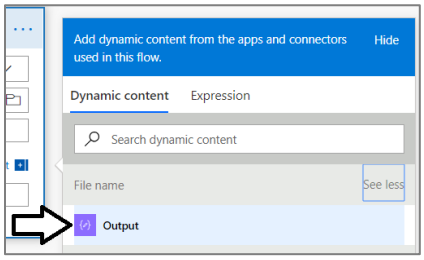
    e) For **Folder Path**, select **/Submitted** to reference the **Submitted** document library you created earlier.



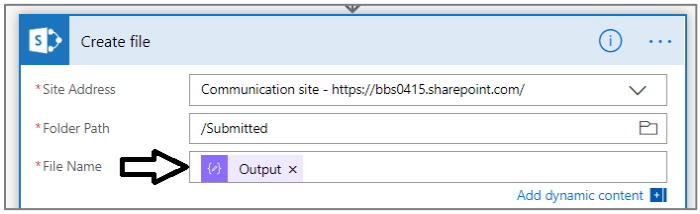
    f) Place your cursor into the **File Name** textbox.
    g) In the **Dynamic content** panel on the right, click the **see more** link for the **File name** action.



h) Click on the **Output** property of the **File name** action.



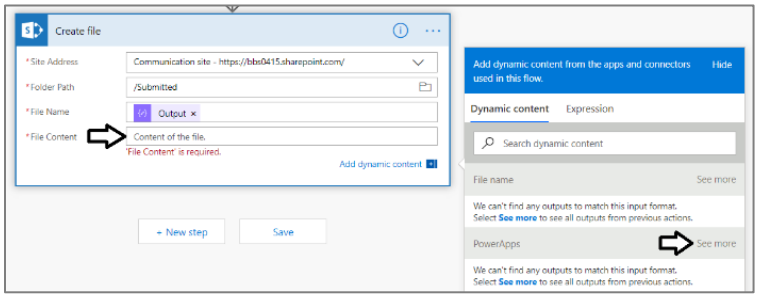
i) You should be able to confirm that the **File Name** textbox contains the **Output** property from the **File name** action.



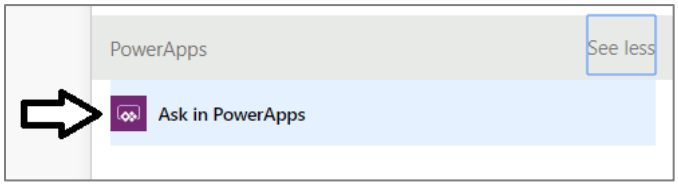
11. Create a PowerApps trigger parameter to pass the photo image from PowerApps to Flow.

a) Click on the textbox for the **File Content** property to place the cursor inside.

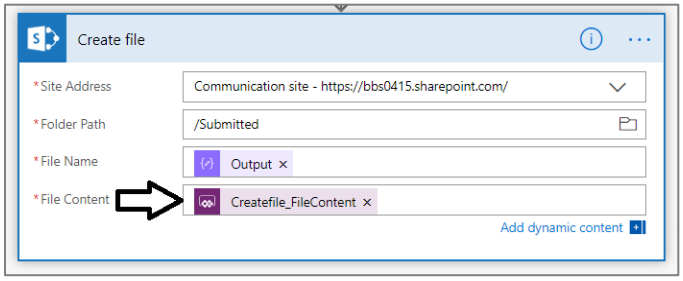
b) Click on the **see more** link for the **PowerApps** trigger.



c) Click on **Ask in PowerApps** to create a new parameter for the PowerApps trigger.



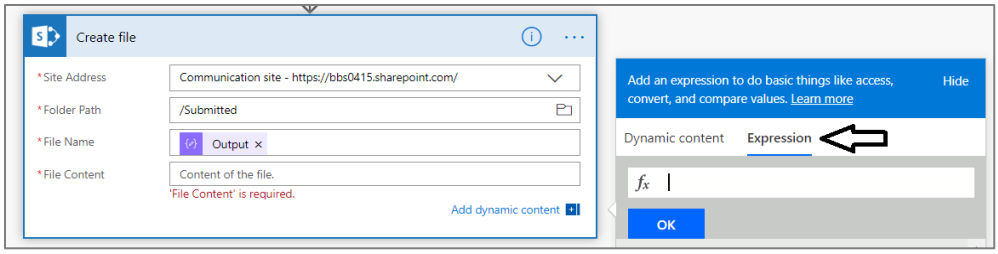
d) You should see that a new trigger output parameter has been created named **Createfile_FileContent**.



In the last step you created the new parameter named **Createfile_FileContent** so PowerApps can pass a photo image to this flow. However, you cannot pass the **Createfile_FileContent** output parameter directly to the **File Contents** input parameter. Instead, you must convert the photo image using the **dataUriToBinary** function before you can upload the photo to SharePoint.

e) Delete the parameter named **Createfile_FileContent** from the **File Content** textbox.

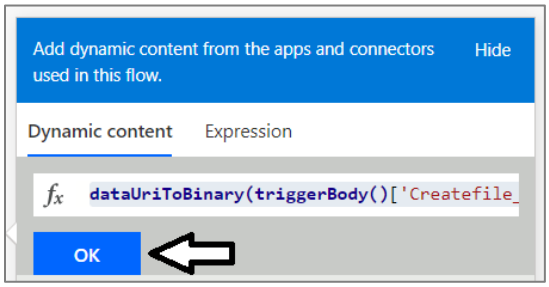
f) Click the **Expressions** link on the right so you can enter an expression for the **File Contents** parameter.



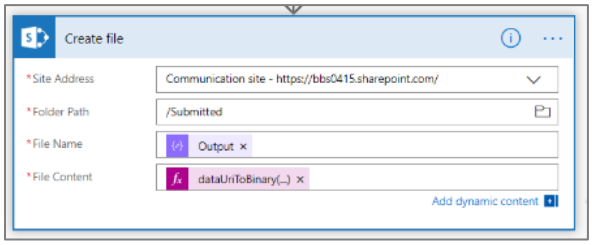
g) Enter the following expression to convert the value passed to **Createfile_FileContent** using the **dataUriToBinary** function.

```
dataUriToBinary(triggerBody()['Createfile_FileContent'])
```

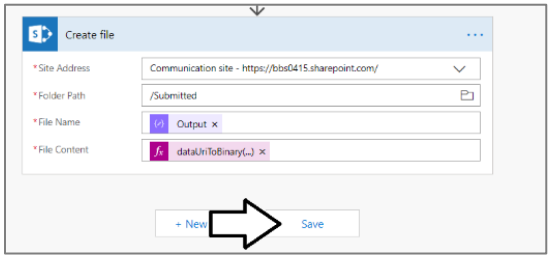
h) Click the **OK** button to save the expression for the **File Contents** parameter.



i) The **File Contents** parameter should now be configured with the expression which calls **dataUriToBinary**

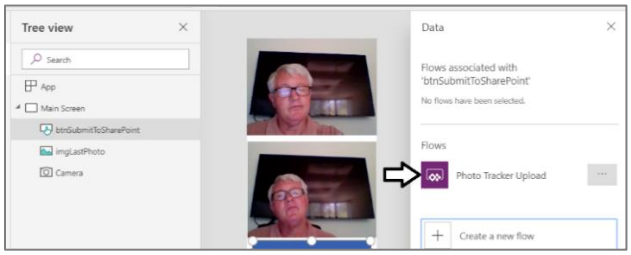


j) Click the **Save** button to save your changes to the **Photo Tracker Upload** flow.

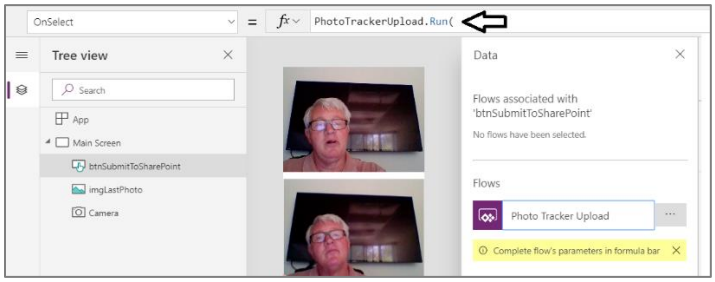


You have now finished creating the **Photo Tracker Upload** flow. Now you will return back to the **Photo Tracker** canvas app in PowerApps Studio to configure the **btnSubmitToSharePoint** button control to execute the flow each time it is clicked.

12. Configure the **btnSubmitToSharePoint** button to execute the **Photo Tracker Upload** flow.
    a) In the browser, switch back to the tab with the **Photo Tracker** app in PowerApps Studio.
    b) In the **Flows** section on the **Data** tab, you should now see the new flow named **Photo Tracker Upload**.
    c) Click on the flow named **Photo Tracker Upload**.



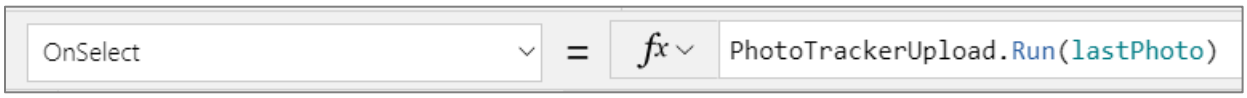
    d) PowerApps Studio will update the **OnSelect** property of the button with **PhotoTrackerUpload.Run(** .



    e) Complete the expression by passing the **lastPhoto** variable in the call to **Run**.

```
PhotoTrackerUpload.Run(lastPhoto)
```

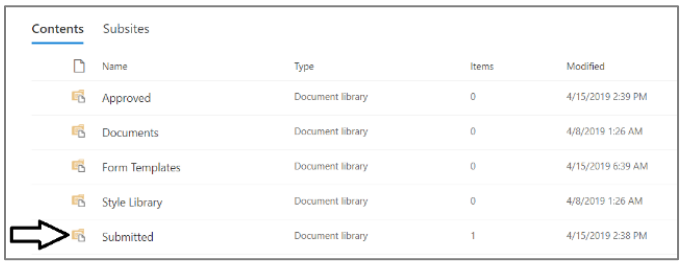
    f) The formula bar for the **OnSelect** property should match the following screenshot.



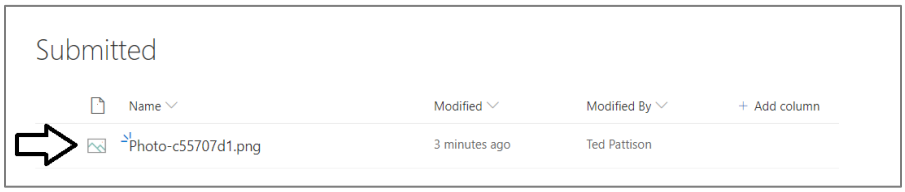
13. Test the **Photo Tracker** app to verify you can use the app to upload photos.
    a) Run the app and then click the camera control so that a photo is displayed in the image control on the bottom of the screen.
    b) Click the **Submit Photo to SharePoint** button to execute the **Photo Tracker Upload** flow.

At this point, the **Photo Tracker** app should have uploaded the photo to the **Submitted** library in SharePoint.

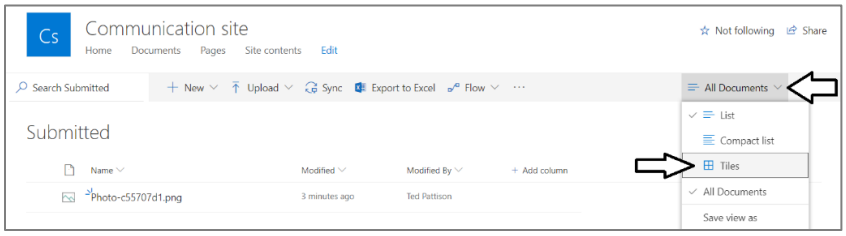
14. Confirm that a photo was uploaded to the **Submitted** document library.
    a) Navigate to the **Site contents** page of your SharePoint site.
    b) Click on the link for the **Submitted** document library.



    c) You should see that a file with a **png** extension has been created.



    d) Change the view for the document library by dropping down the view menu on the right and selecting the **Tiles** view.



    e) You should now see the photo image.

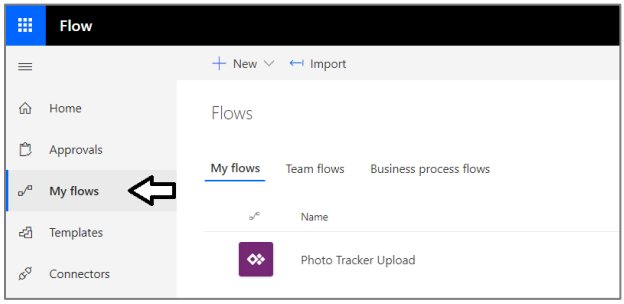
    f) Return to the Photo Track canvas app and upload a few more photos.

15. **Optional *step***: launch the **Photo Tracker** app using PowerApps mobile and upload a photo from your mobile phone.

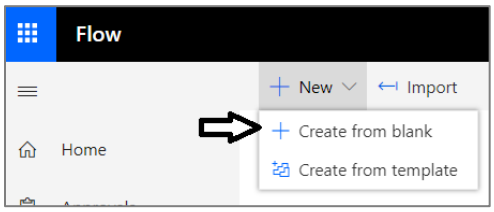
Exercise 3: Create A Flow to Automate a Photo Approval Process

In this exercise, you'll use Microsoft Flow to create a an approval workflow associated with a photo that has been added to the Submitted document library.

1. Sign in to Microsoft Flow.
   a) Navigate to http://flow.microsoft.com and sign in using your Office 365 trial account.
2. Create a new blank flow named **Photo Approval**.
   a) Click the **My flows** link to see your existing flows.

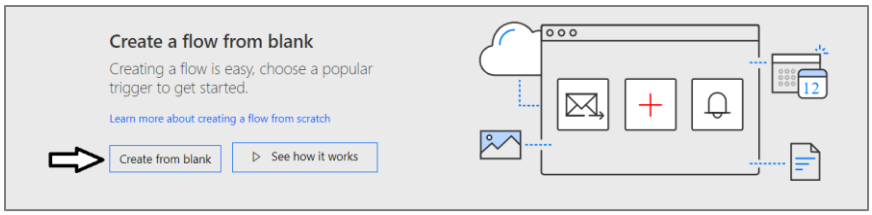


   b) Drop down the **+ New** menu select the **+ Create from blank** menu command.

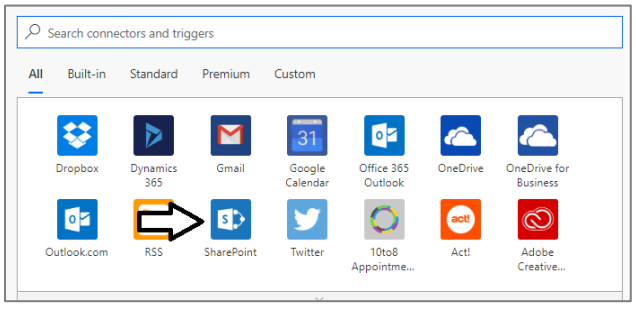


You must start by selecting a trigger for your new flow. In this scenario, you will configure the flow to trigger when a new file is added to the **Submitted** document library that you created in your SharePoint site.

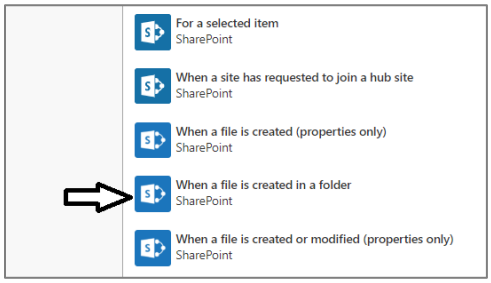
   c) Select the **Create from blank** trigger.



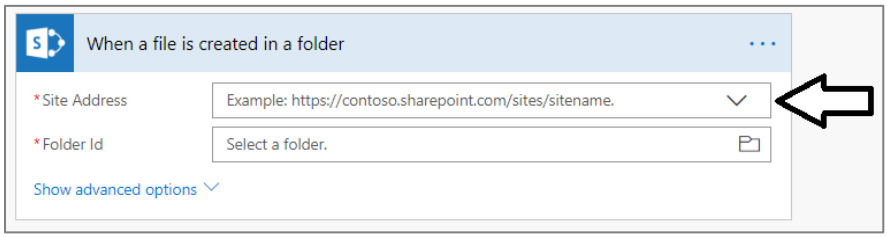
   d) Click on **SharePoint** to filter the set of riggers below.



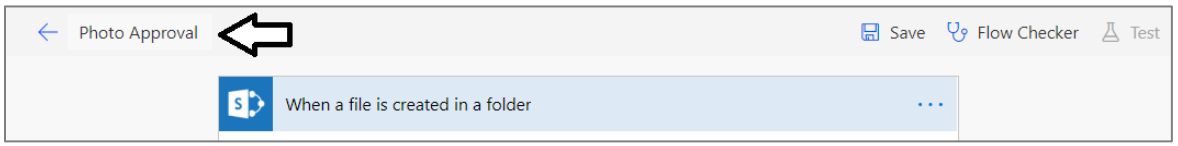
e) Select the trigger named **When a file is created in a folder**.



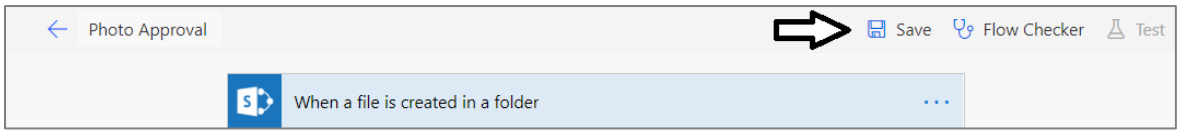
f) The new flow should now container the **When an item is created** trigger for a SharePoint list.



g) Update the name of the flow to **Photo Approval**.

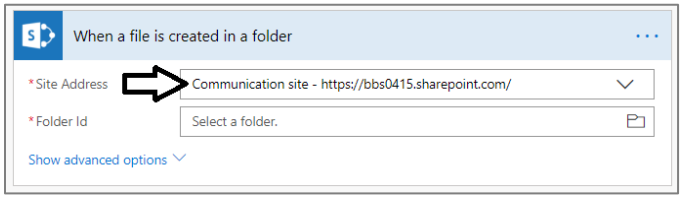


h) Click the **Save** button to save your flow.

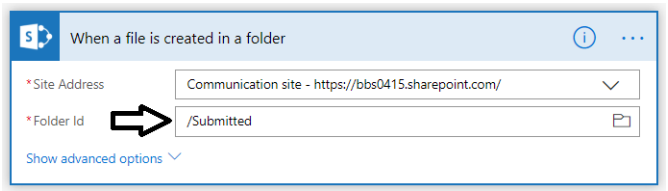


3. Configure the **When a file is created** trigger.

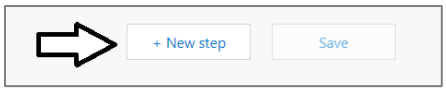
a) Select your site URL from the **Site Address** dropdown.



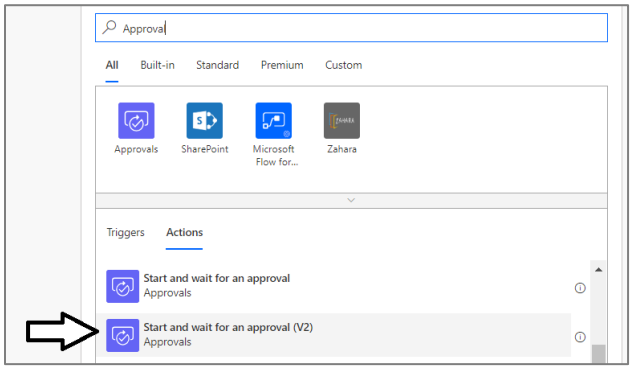
b) Configure the **Folder Id** to reference the **/Submitted** document library.



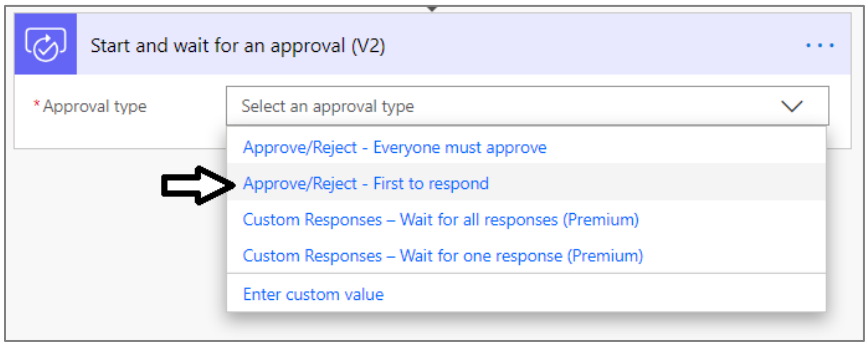
4. Add an **Approvals** action.
   a) Click +New step and then Add an action.



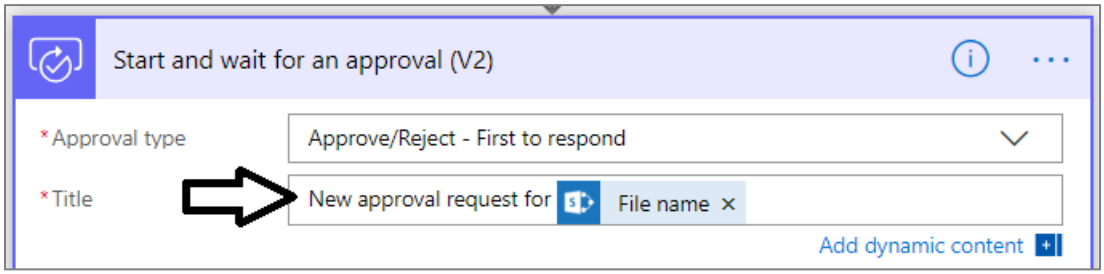
   b) Search for: **Approvals**.
   c) Select the **Start and wait for an approval (V2)** action.



5. Configure the approval so anyone from the assigned list should be able to approve the request.
   a) Set the **Approval type** with the option **Approve/Reject – First to respond**.



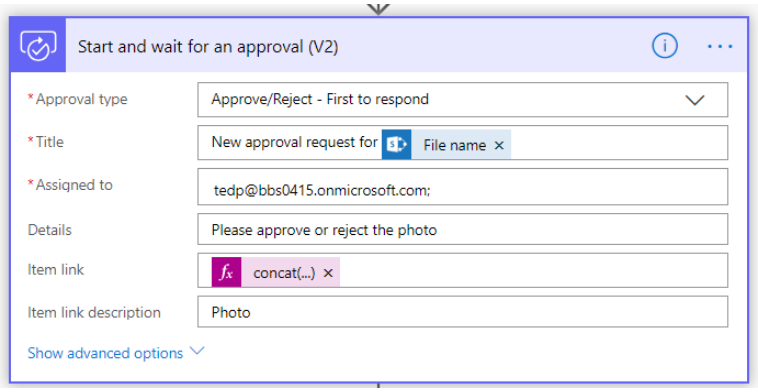
6. Set the **Title** of the approval request.
   a) Click on the **Title** edit box and type "**New approval request for** ". (*Do not type quotation mark, but add a space at the end*)
   b) In the Dynamic content box on the right, select **File name**.
   c) Your screen should match the following screenshot.



7. Fill in the remaining input parameters for the **Start and wait for approval (V2)** action.
   a) Assign the email address for your Office 365 user account to the **Assigned to** parameter.
   b) For the **Details** parameter, enter **Please approve or reject the photo**.
   c) For the **Item link** parameter, enter the following expression.

```
concat('https://bbs0415.sharepoint.com/',triggerOutputs()['headers']['x-ms-file-path'])
```

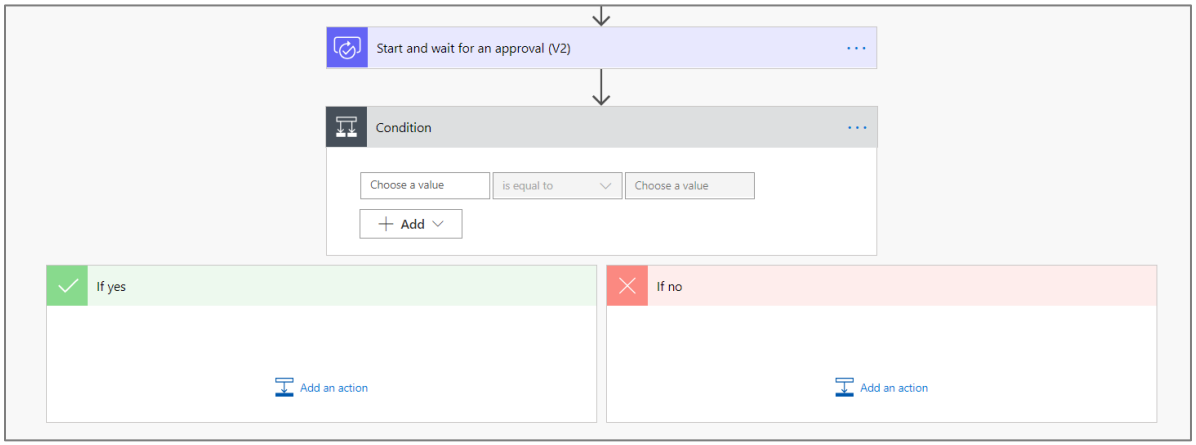
   d) For the **Item link description** parameter, enter **Photo**.
   e) The **Start and wait for approval (V2)** action in your flow should match the following screenshot.



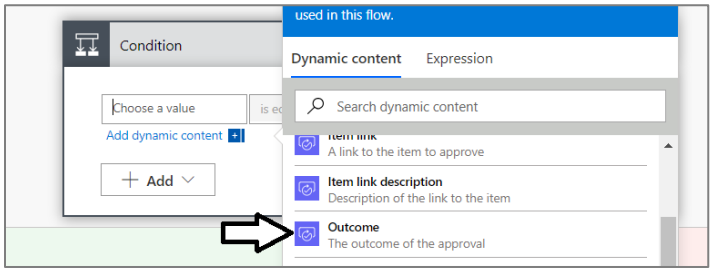
Now that we have an approval, we can customize the actions to take based on the result of the approval. The **Output** parameter of the **Start and wait for approval (V2)** action will have a value of **Approve** if the photo has been approved and a value of **Reject** if the photo was rejected.

8. Add a condition to the flow to determine is the **Output** parameter of the approval is ***Approve*** or ***Reject***.
   a) Click the **+ New Step** button to add a new step at the bottom of the flow.
   b) Search for **Condition** and then select the **Condition** action.

   c) You should now see a new **Condition** action in the flow designer as shown in the following screenshot.

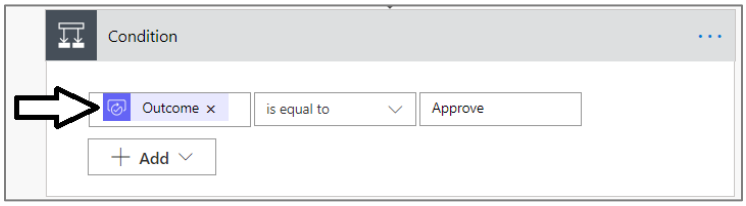


d) Click in the left edit box with the hint **Choose a value** and select **Output** from the dynamic content pane.

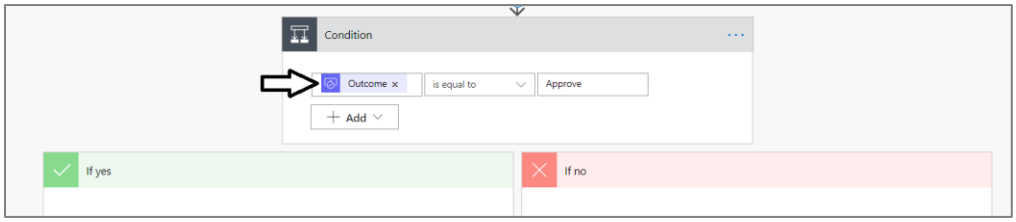


e) Make sure the dropdown menu in the middle is set to **is equal to**.

f) Click in the right edit box and type a string value of **Approve**.



g) You should see that below the **Condition** box, there are two more boxes with branches for **If yes** and **If no**.



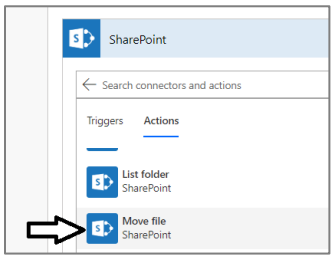
Over the next few steps you will implement the logic for the **If yes** branch. You will perform an action on the same photo that triggered the flow, based on information passed from the **When a file is added to a folder** trigger.

9. In the **If yes** branch, add an action to move the photo from the **Submitted** document library to the **Approved** document library.

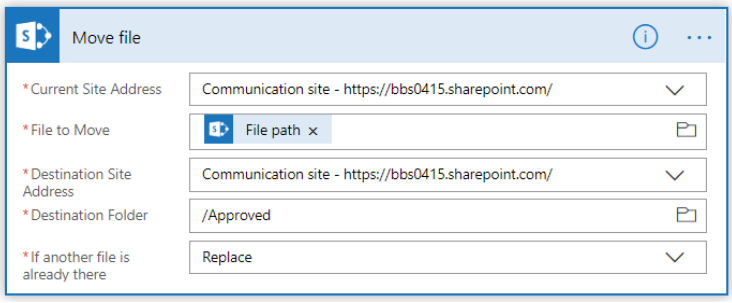
a) In the left **"If yes"** box, click **Add an action**

b) Search for **SharePoint Move File**.

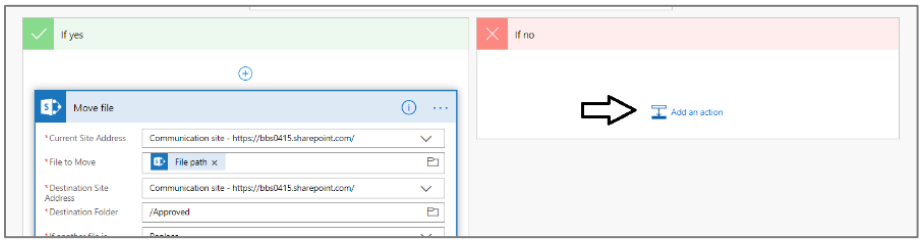
c) Select the SharePoint **Move file** action.



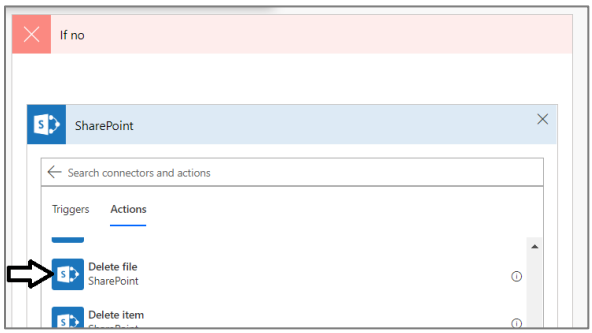
d) Set **Current Site Address** to your SharePoint URL.
e) Set **File to move** to the **File path** output parameter from the **When a file is added to a folder** trigger.
f) Set **Destination Site Address** to your SharePoint URL.
g) Set the **Destination folder** to the **/Approved** document library.
h) Set **If another file is already there** to **Replace**.



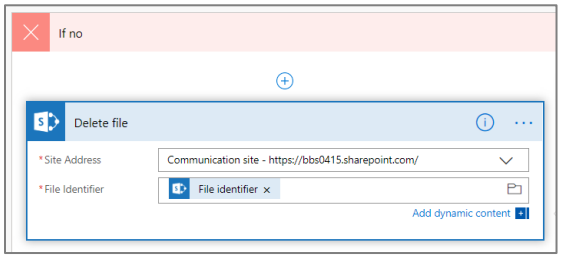
10. In the **If no** branch, add an action to delete the photo from the **Submitted** document library.
a) In the left **"If no"** box, click **Add an action**



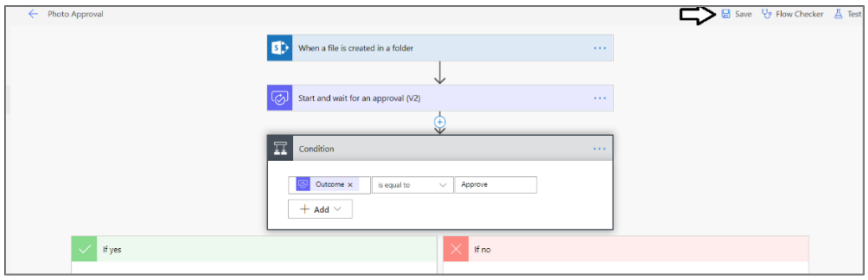
b) Add a SharePoint **Delete file** action.



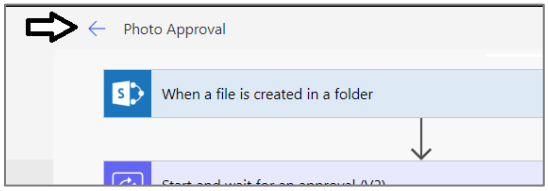
c) Set **Site Address** to your SharePoint URL.
d) Set the **File Identifier** parameter to the **File Identifier** output parameter from the **When a file is added to a folder** trigger



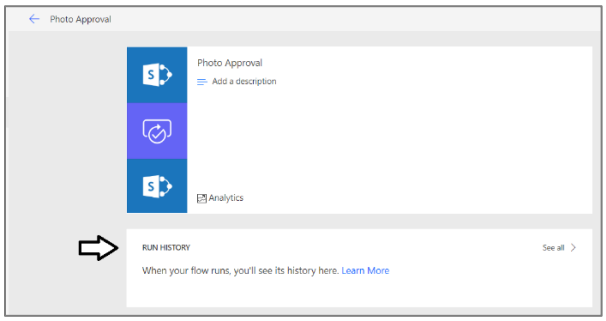
11. Save your work on the Photo Approval flow.
    a) Click the **Save** button at the top right to save your work.



    b) Click the back arrow button to move back to the page which shows the flow run history.



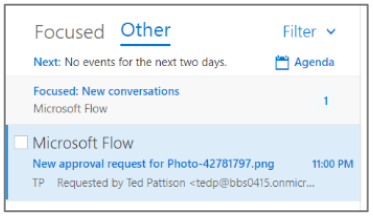
    c) At this point, the **RUN HISTORY** list should be empty.



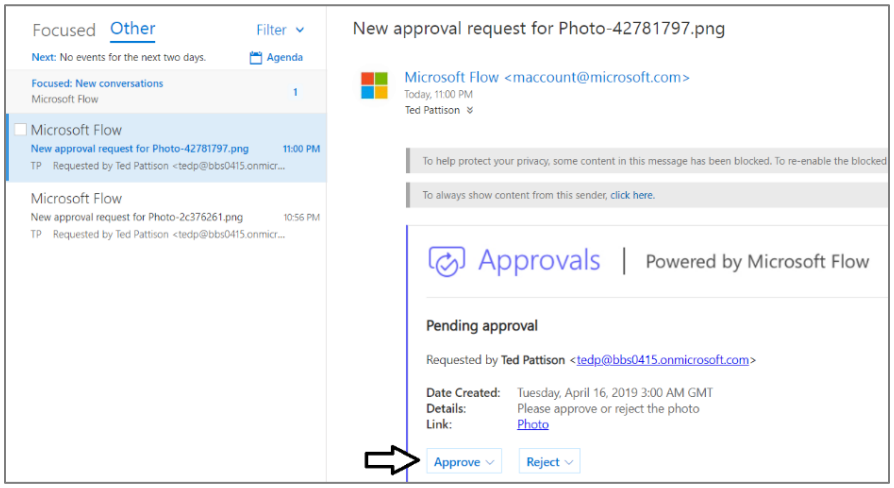
Exercise 4: Test the Photo Approval Flow

In this exercise, you will examine flow history to see what happened during the lifetime of each flow.

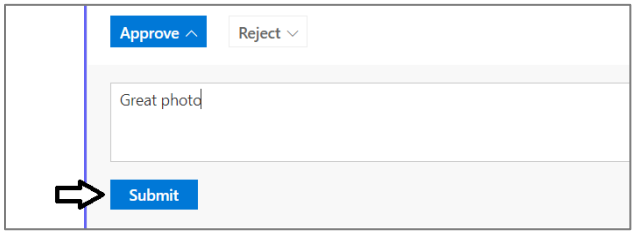
1. Delete all existing photos from the **Submitted** document library before you start your testing.
    a) In a separate browser tab, return to your SharePoint site and navigate to the **Submitted** document library.
    b) Delete all the files from the **Submitted** document library.
2. Run the **Photo Tracker** app and upload a new photo.
3. Return to the **Submitted** document library in SharePoint and verify the file has been uploaded.
4. Open Outlook and find the approval email.
    a) Open Outlook and find the email sent for the approval *(you might have to wait a minute or two before the message appears)* .



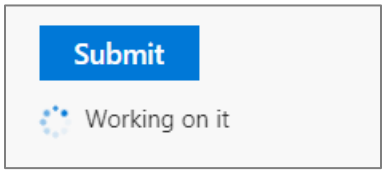
b) In the email body, locate and click the **Approve** button.



c) Add a comment and then click **Submit** to complete the approval process.

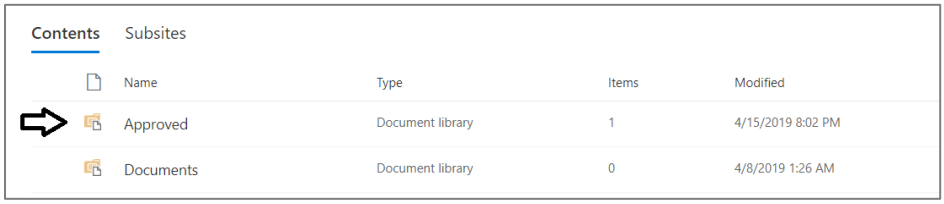


d) Wait while the approval process runs.

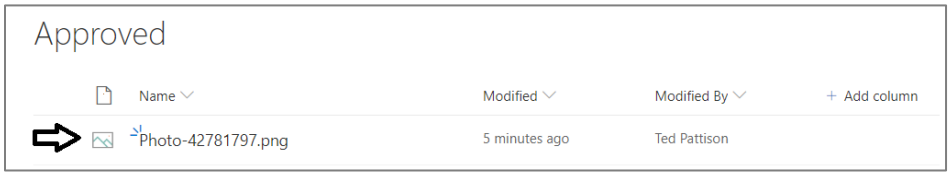


5. Check to verify the photo was moved to the **Approved** document library

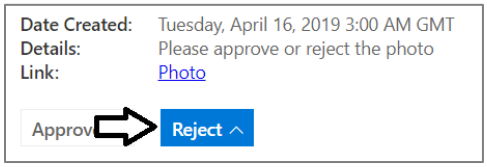
a) In your SharePoint site, navigate to the **Approved** document library



b) Verify you can see the photo has been moved.

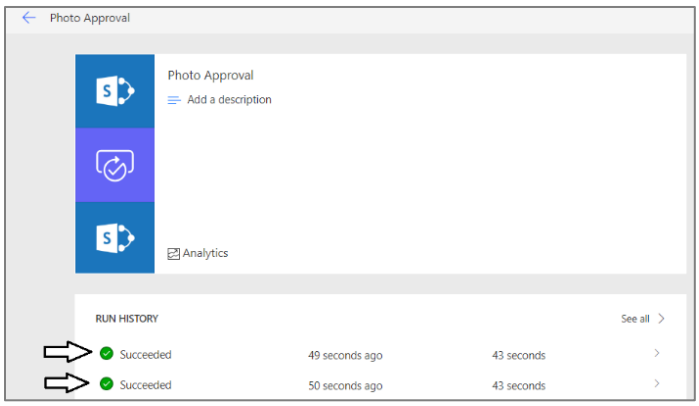


6. Test the flow for the scenario when a photo is rejected.
   a) Return to the **Photo Tracker** app and submit a new photo.
   b) Return to the **Submitted** document library in SharePoint and verify the file has been uploaded.
   c) Return to Outlook and find the approval email.
   d) In the email body, locate and click the **Reject** button.

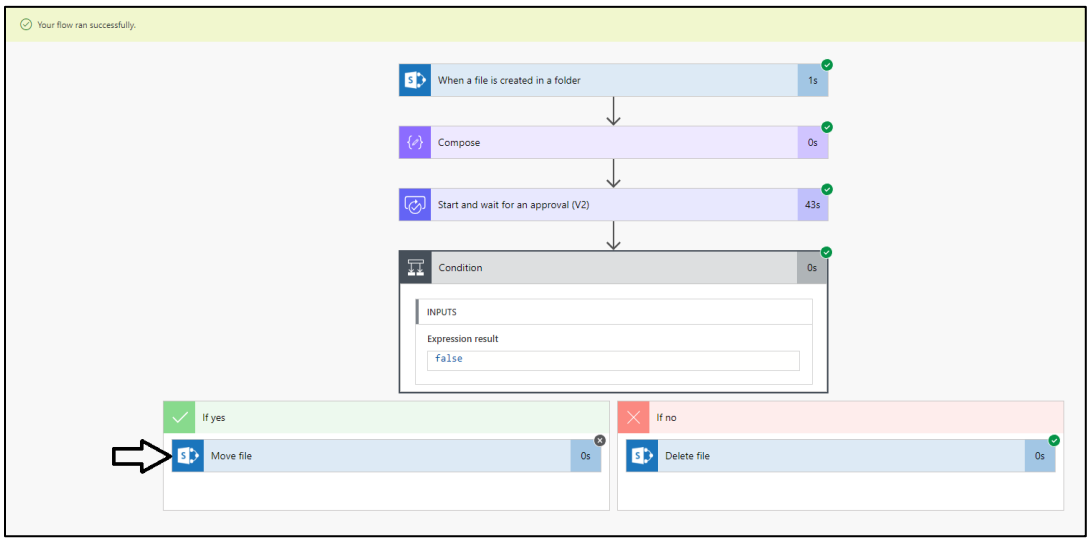


   e) Wait about a minute for the required processing to occur.
   f) Return to the **Submitted** document library in SharePoint and verify the rejected photo has been deleted.

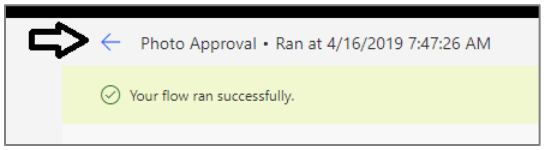
7. Inspect the RUN HISTORY for the two flows that have run.
   a) Return to the browser tab for the Photo Approval flow which shows the **RUN HISTORY**.
   b) When you refresh the page, you should see that two flows have run.



   c) Click on the bottom flow which ran first to see the history of a flow in which the photo was approved.
   d) You should be able to see in the run history that the flow of execution moved into the **If yes** branch.



e) Click the back arrow to move back to the page which shows all run history for all flows.



f) Click on the top flow which ran second to see the history of a flow in which the photo was rejected.

g) You should be able to see in the run history that the flow of execution moved into the **If no** branch.

You have now completed this lab.