**Task - 1 – Segregating Multiple file formats(JSON, XML**, **CSV,) in separate folders.**

**Copy multiple File formats from a folder in blob storage.**

Eg: I have 3 file format, i.e. **JSON, XML**, and **CSV**, But I want to filter only **JSON** and **CSV**, file format and store them in a separate folder.

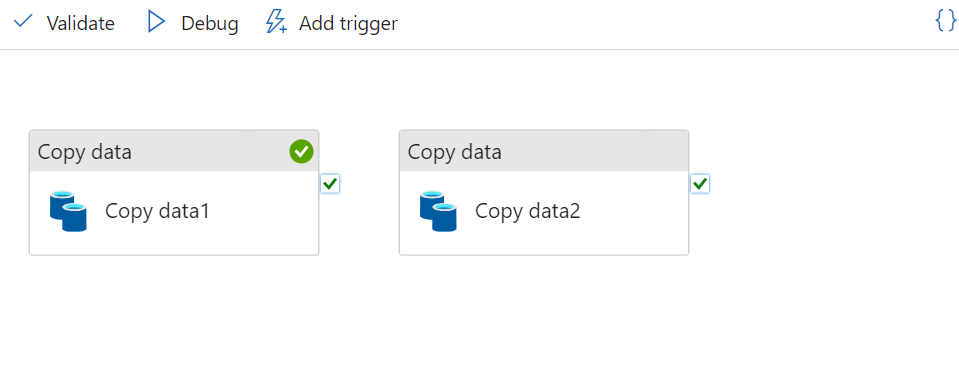
All CSV files in CSV Folder

All JSON files in JSON Folder.

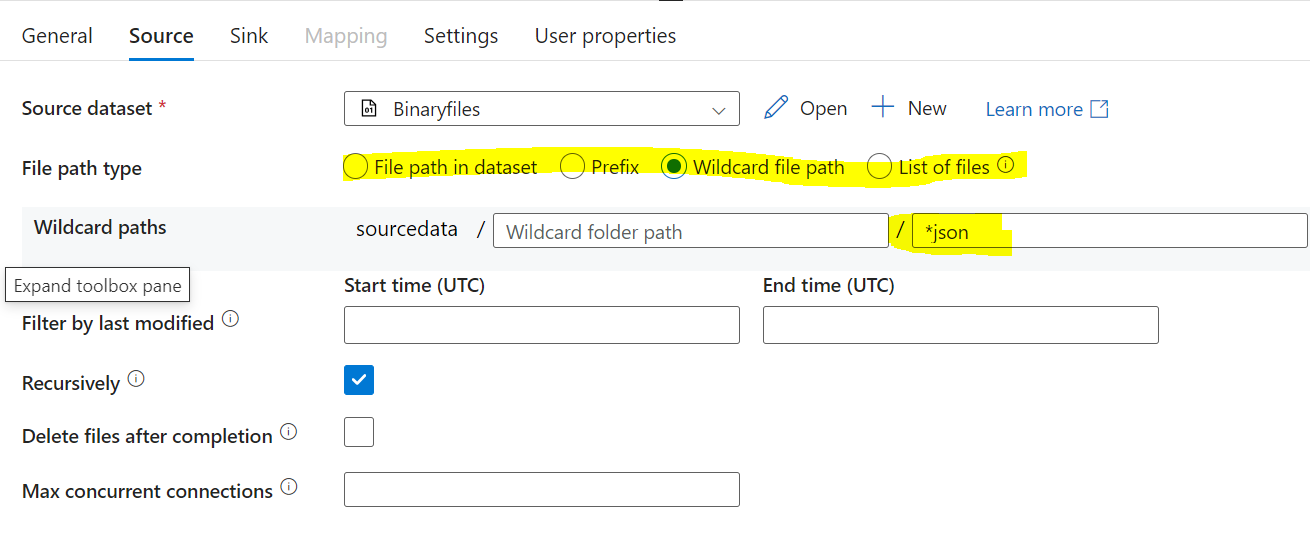
**There are 2 method to perform this task:**

**1st Method->**

create 2 **copy activity** and use wildcard - \*csv and \*json in another copy activity as shown below.

****

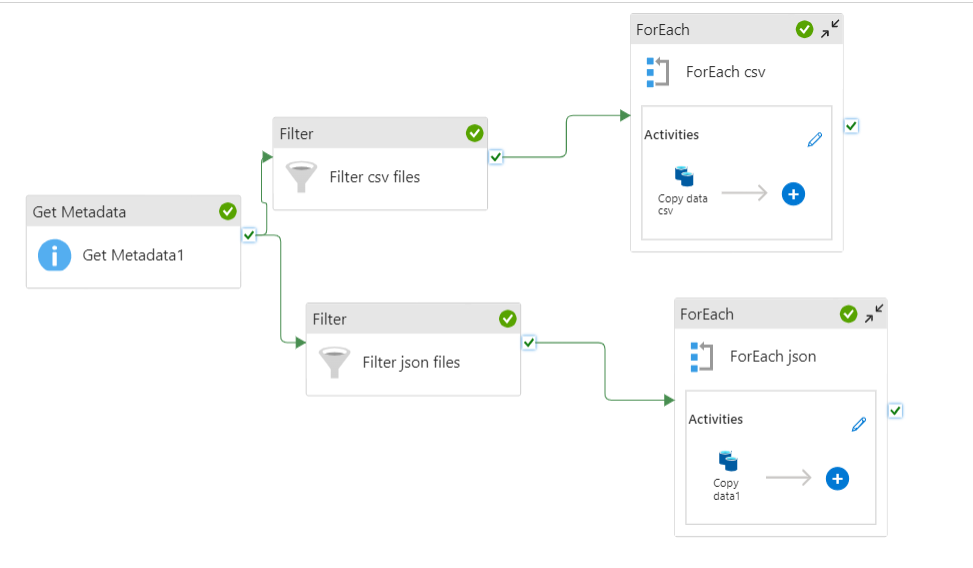
**Write the wild card properties**



**2nd Method->**

Creating a Get Metadata -> Filter Activity -> For Each activity (Inside ForEach activity insert Copy Activity)

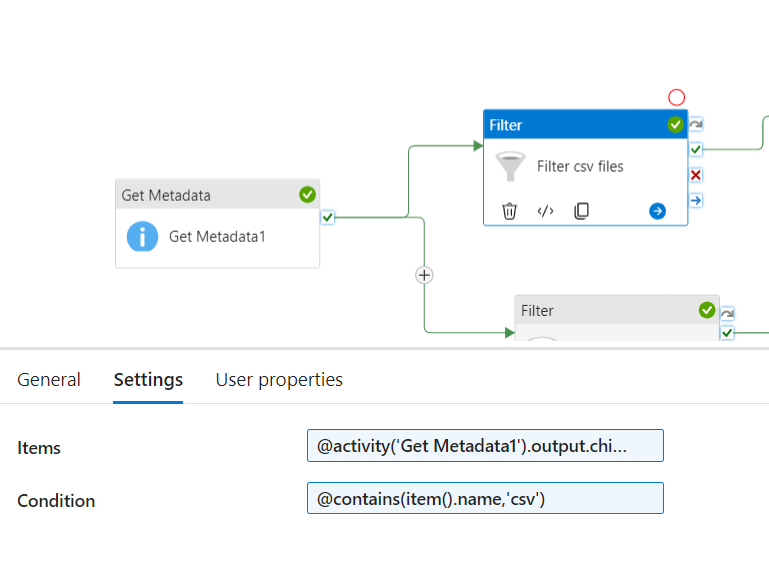
**Refer the steps shown in below picture:**



**Inside both the Filter Activity write the below expressions:**

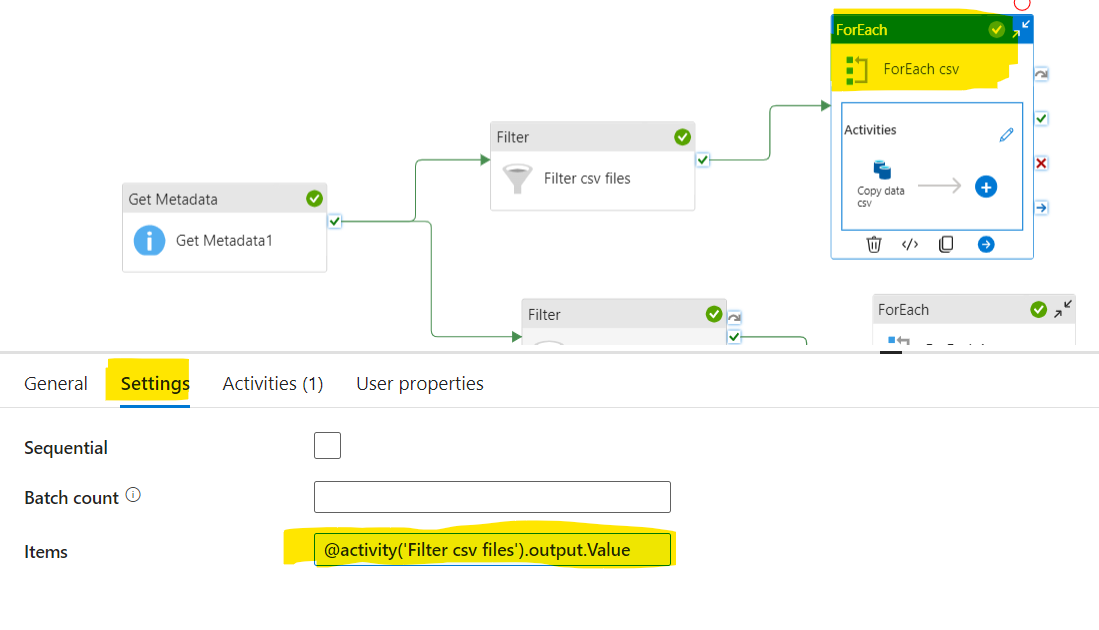
@activity('Get Metadata1').output.childItems

@contains(item().name,'csv')



**Inside ForEach activity write the below expression:**

@activity('Filter csv files').output.Value



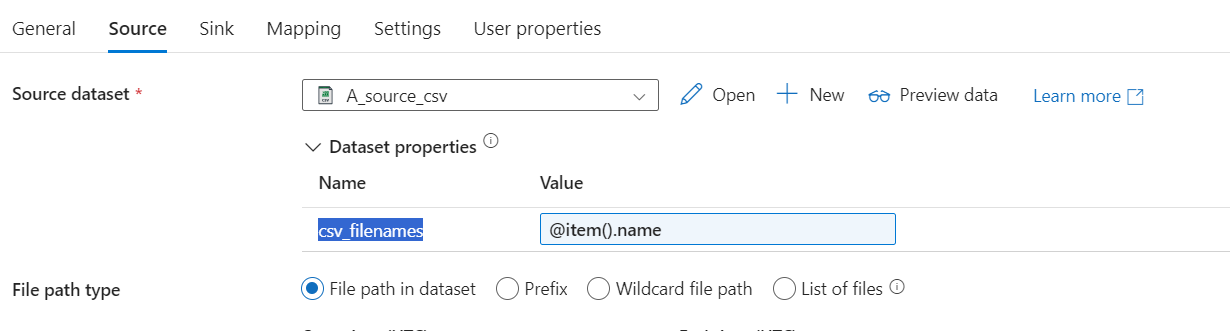
Create Copy activity inside ForEach

Source – select the file path from blob storage

Click o**pen** in **dataset**

Create **parameter** inside dataset – “**csv\_filenames”**

Come to pipeline level and write the value for parameter as shown below:

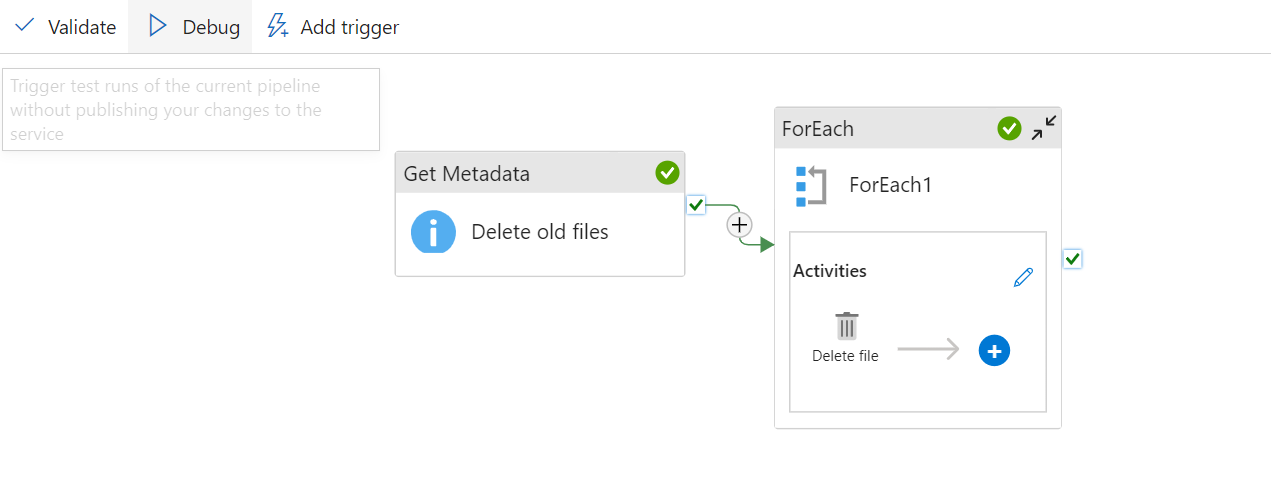


And select the Sink dataset.

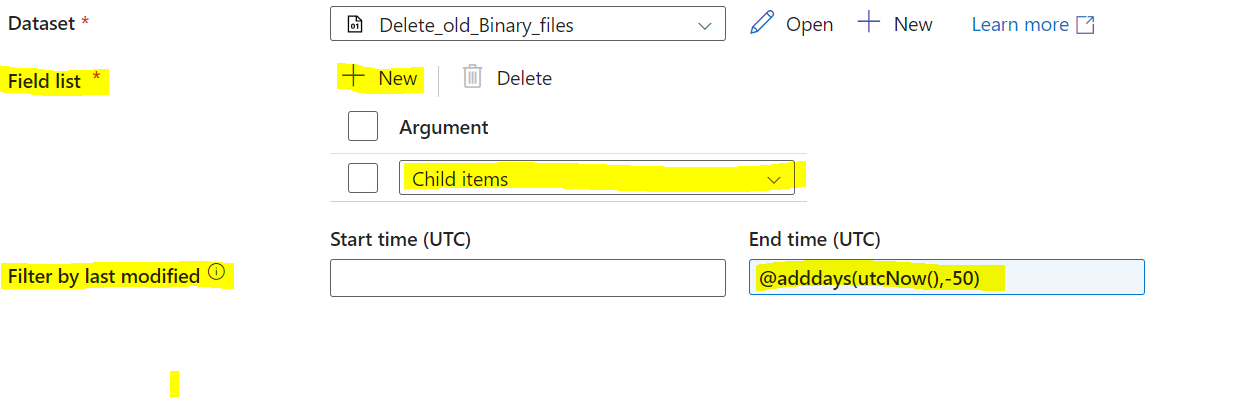
Run the pipeline, it will copy only CSV file format for 1st Filter activity and JSON file format for 2nd Filter activity.

**Task - 2 – Deleting the old files in blob storage.**

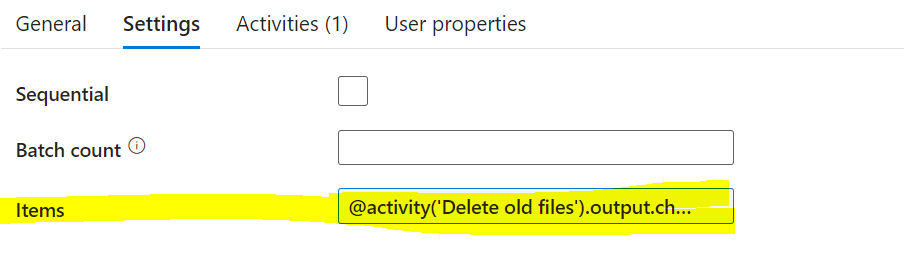
Eg: If I want to delete the files in the blob storage which are older than 50 days .



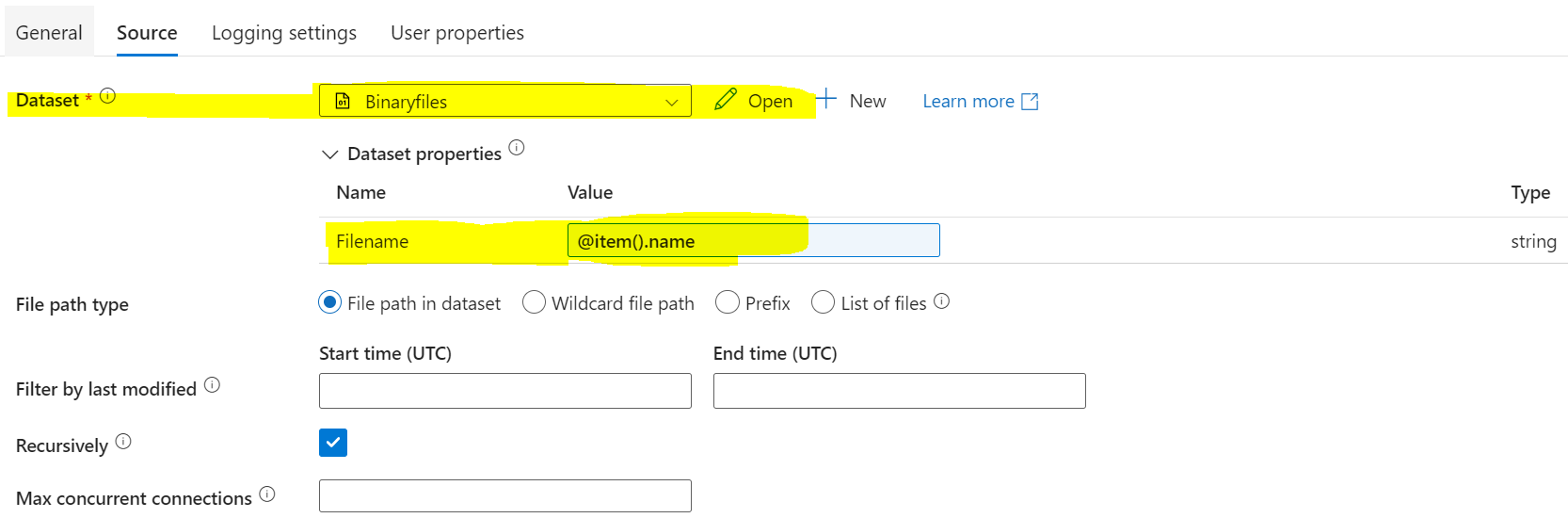
**Step-1** Create a **Get MetaData** activity and inside the **Get MetaData** activity create the “Field list” and enter expression in **“Filter by last modified”** as shown in below image.



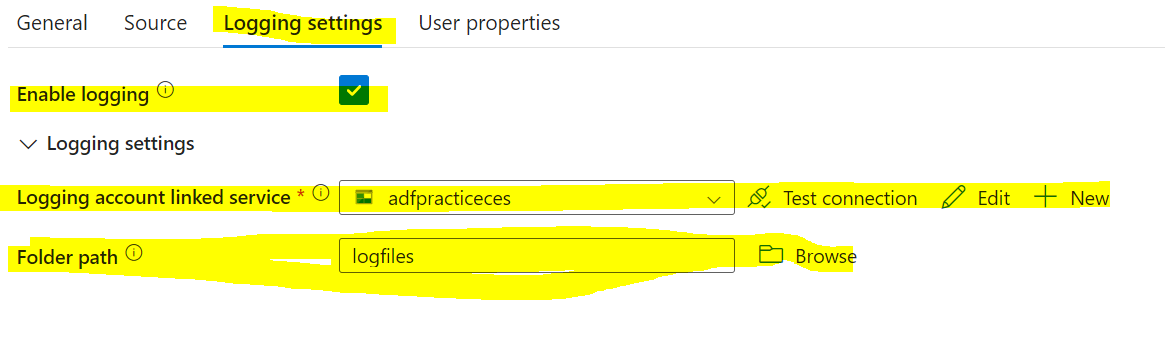
**Step-2** In **ForEach** activity Go to settings and enter the expression in items ( To fetch childItems from **Get MetaData** activity)  
  
@activity('Delete old files').output.childItems



Step-3 Inside the **ForEach** activity add the **Delete** activity and select the source dataset and create the **parameters** and call the item name as shown in below image

****

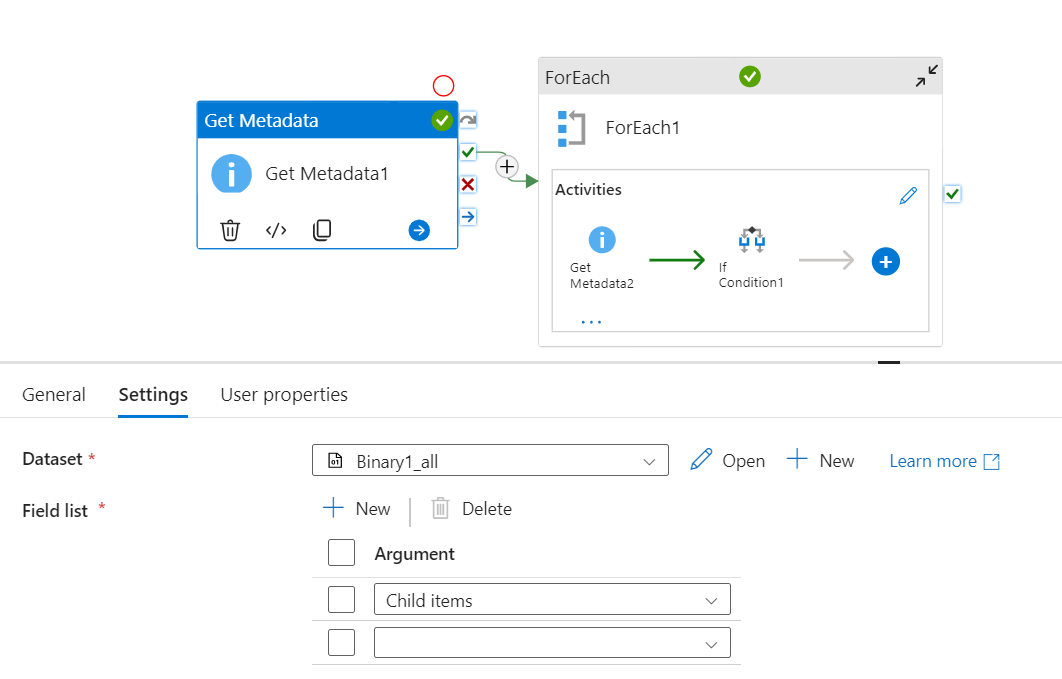
**Step-4** Now enable **“Logging” and select the log files path to store them** as shown in below image.

****

**Finally now “Debug” .**

**Task - 3 – Segregating Multiple file based on File size (KB, MB, GB) in separate folders.**

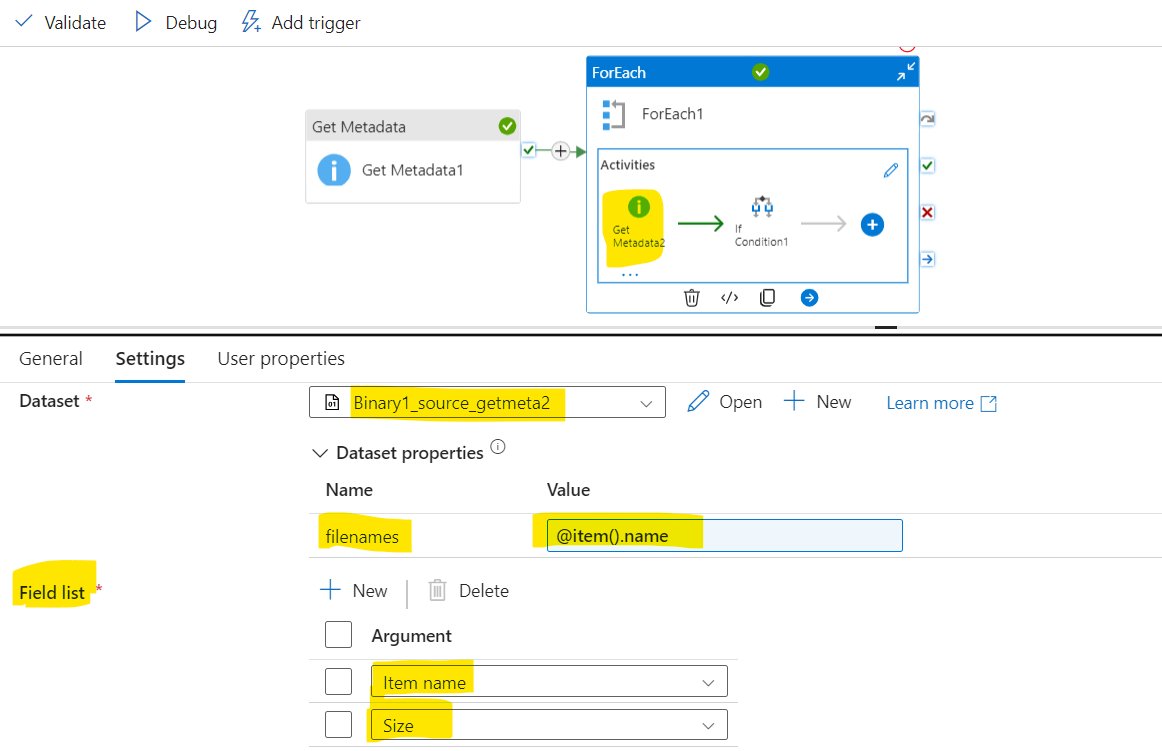
1. Create GetMetadata activity and get the childitems of the source folder, select File source as “Binary” while creating Dataset.



1. Take **ForEach Activity** and in **settings**

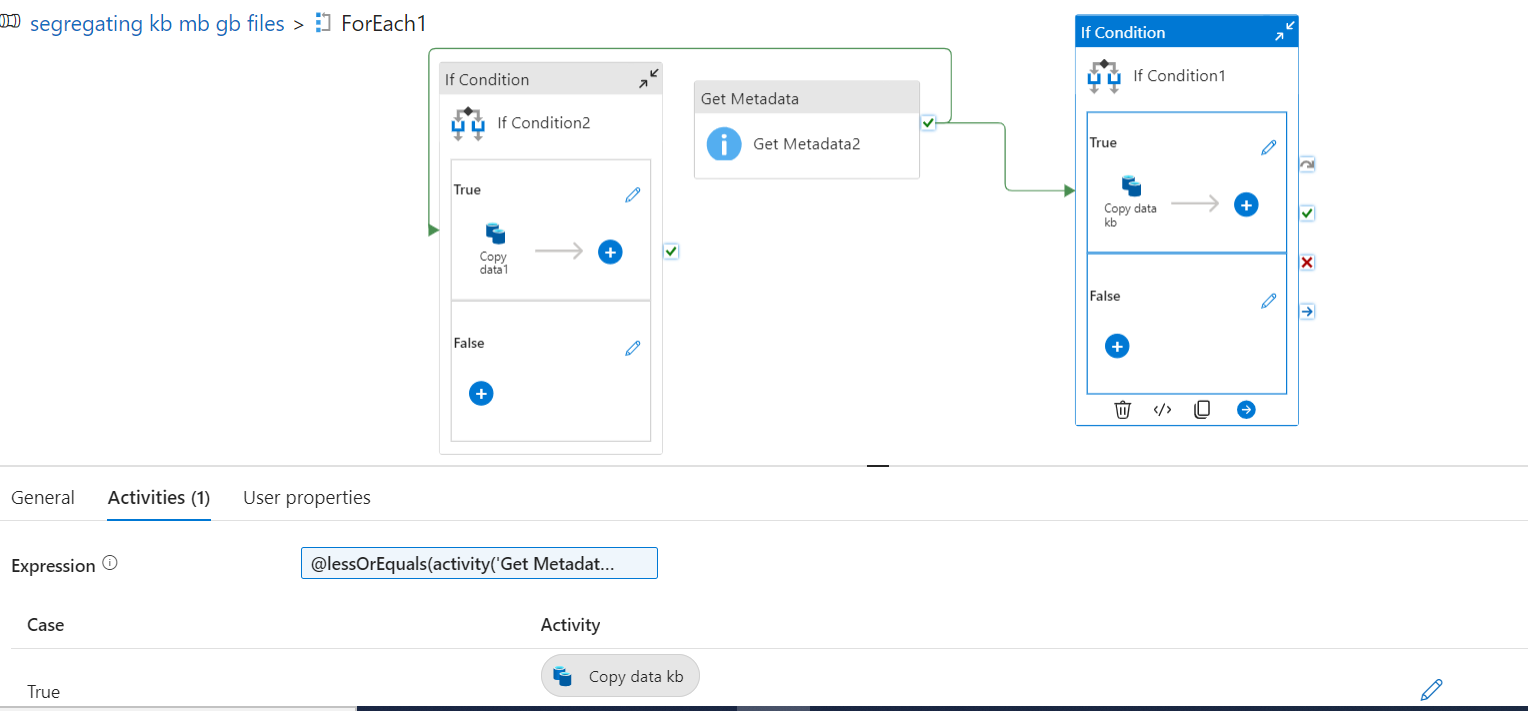
Write this: @activity('Get Metadata1').output.childItems

1. Add **GetMetaData** activity inside **ForEach Activity.**
2. Add fresh dataset source of the same path in container.
3. Create parameter – **filenames**
4. Pass **@item().name** in parameter in value



1. Add **If condition Activity** inside **ForEach Activity (as shown in below fig)**
2. Write this Expression **in If Condition** – (This filters files less than **999KB**)

@lessOrEquals(activity('Get Metadata2').output.size,999999)



1. Inside **If condition Activity,** add **copy** **Activity** in True
2. In **copy** **Activity** -> Under **Source**, take the **same dataset** which is used in **GetMetadata Activity 2.**
3. Create **Parameter** and the below value in parameter.

