CLOUD PLATFORMS AND ANALYTICS – WORKSHOP

Azure Logic Apps

This workshop will give you a hands-on introduction to logic apps by walking you through a use case for creating an integration workflow with Azure Logic Apps using the Azure portal.

IMPORTANT:

- The services covered in this course are only a subset of a much larger family of Azure services. Similar outcomes can be achieved by leveraging other services and/or features not covered by this workshop. Specific business requirements may also ask for the use of different services or features not included in this workshop.
- Some concepts presented in this course can be quite complex and you may need to seek for more information from different sources to compliment your understanding of the Azure services covered.

Document Structure

This document contains detailed step-by-step instructions on how to automate workflows using Azure Logic Apps. It is recommended you carefully read the detailed description contained in this document for a successfully complete this workshop.

You will see the label **IMPORTANT** whenever a there is a critical step. Please pay close attention to the instructions given.

Table of Contents

Azure Logic Apps 1
Document Structure 1
Table of Contents 2
Introduction 3
Prerequisites 3
Create a storage account and containers 4
Create a Logic App resource 7
Select the blank template 9
Add a Recurrence trigger 10
Parse the CSV files to JSON and calculate the total sales 12
Run your workflow 22
Clean up resources 22

Introduction

This workshop aims to provide a comprehensive understanding of leveraging Azure Logic Apps to automate workflows.

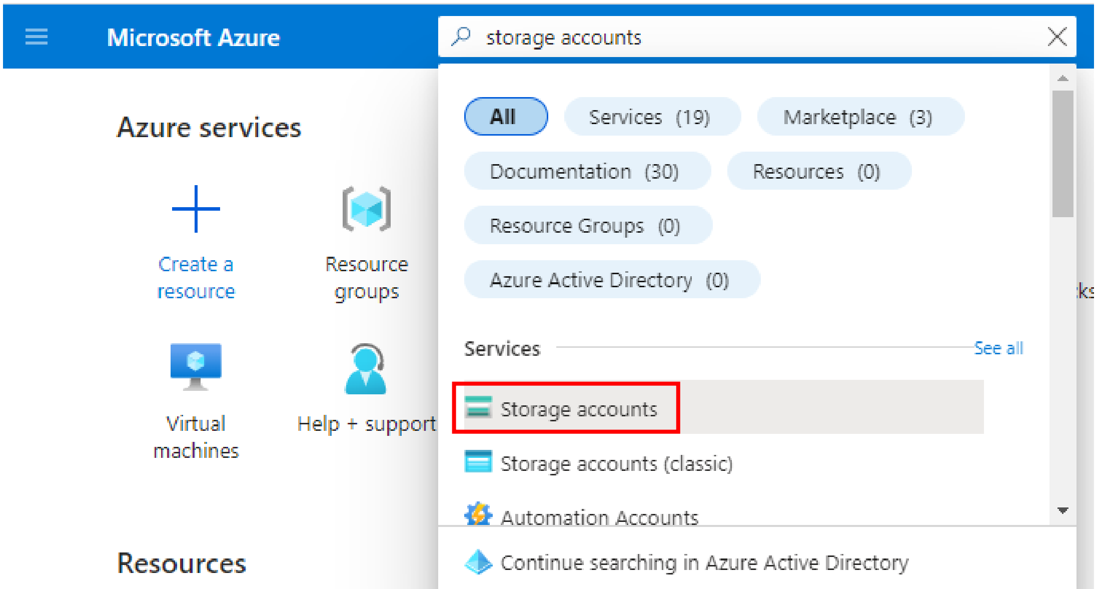
The workshop will cover Azure Logic Apps, a powerful cloud service that allows you to automate and orchestrate tasks, workflows, and business processes. You will specifically learn how to create a Logic App workflow designed to automate the process of parsing CSV files to JSON format.

Prerequisites

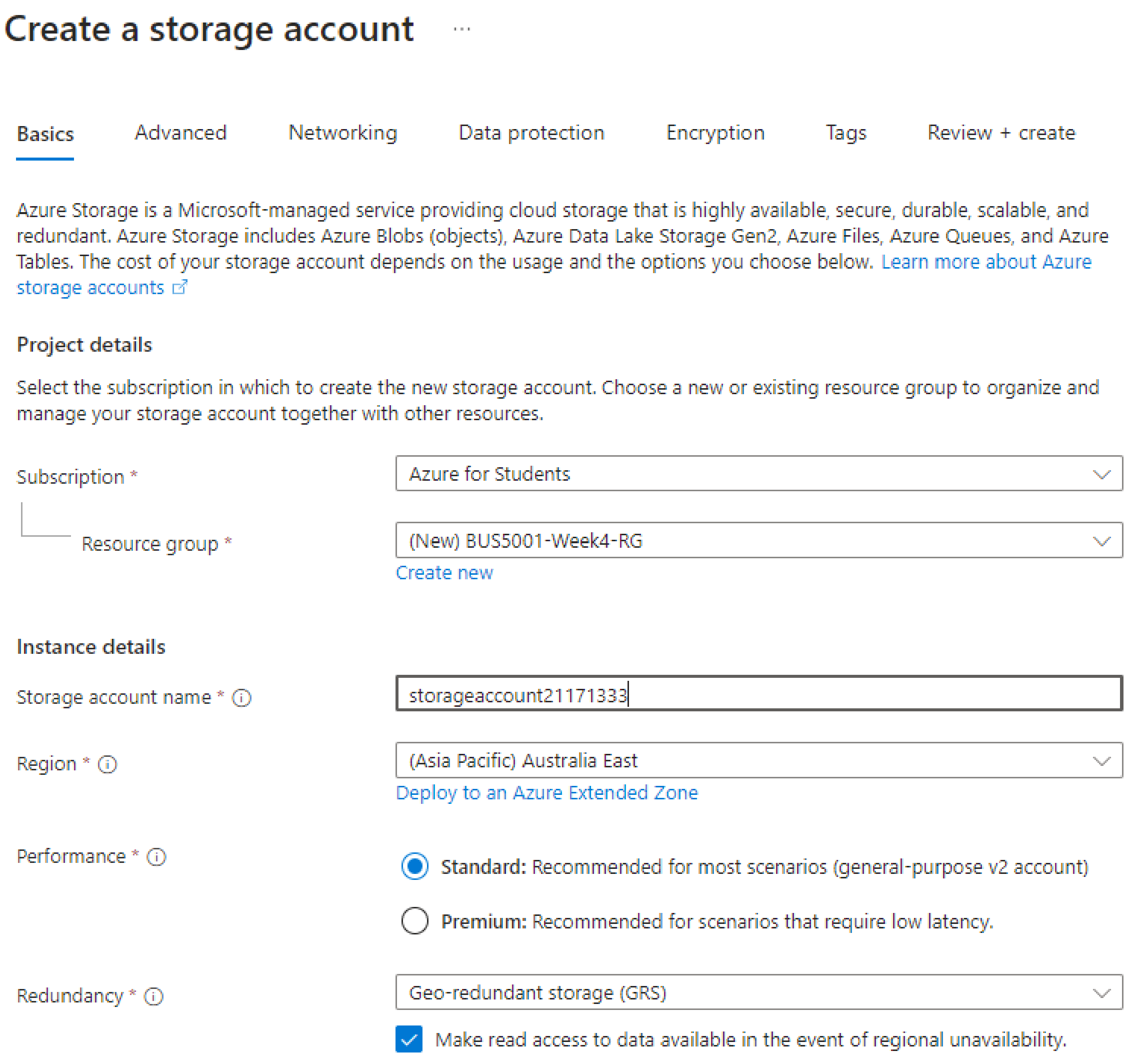
- An Azure account and subscription. If you don't have a subscription, sign up for a free Azure account using your La Trobe student account.

Create a storage account and containers

1. Sign in to the Azure portal with your Azure account.
2. In the Azure search box, enter 'storage account', and select Logic apps



3. On the Storage accounts page, select **+ Create**.
4. On the **Create a storage account** pane, on the **Basics** tab, provide the following basic information about your logic app:
   - **Subscription**: Your Azure subscription name. *Ex: Azure for students*
   - **Resource Group**: Select **Create new** to create a new resource group: *Ex: BUS5001-Week4-RG.* (ensure the name is 1-90 characters long, contains only alphanumeric characters, periods, underscores, hyphens, and parentheses (no spaces), and is unique within the subscription)
   - **Storage account name:** *Ex: storageaccount<student_id>*. (ensure the name is 3-24 characters long, contains only lowercase letters and numbers, and is unique across all Azure)
5. Use default values for other settings. When you're done, your settings look similar to this version:



Create a storage account ...

Basics | Advanced | Networking | Data protection | Encryption | Tags | Review + create

Azure Storage is a Microsoft-managed service providing cloud storage that is highly available, secure, durable, scalable, and redundant. Azure Storage includes Azure Blobs (objects), Azure Data Lake Storage Gen2, Azure Files, Azure Queues, and Azure Tables. The cost of your storage account depends on the usage and the options you choose below. Learn more about Azure storage accounts

Project details

Select the subscription in which to create the new storage account. Choose a new or existing resource group to organize and manage your storage account together with other resources.

Subscription * : Azure for Students

Resource group * : (New) BUS5001-Week4-RG

Create new

Instance details

Storage account name * : storageaccount21171333

Region * : (Asia Pacific) Australia East

Deploy to an Azure Extended Zone

Performance * :

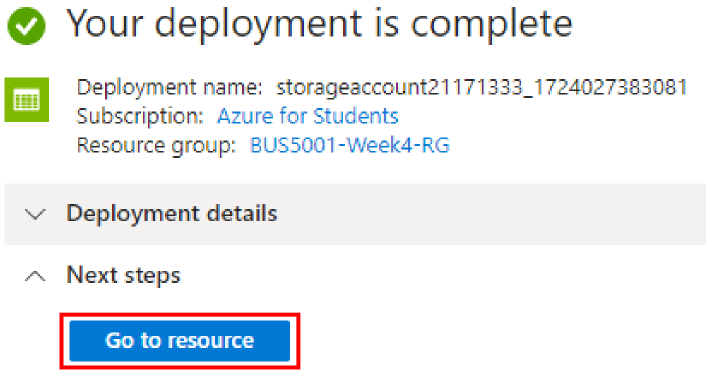
● **Standard:** Recommended for most scenarios (general-purpose v2 account)

○ **Premium:** Recommended for scenarios that require low latency.

Redundancy * : Geo-redundant storage (GRS)

☑ Make read access to data available in the event of regional unavailability.

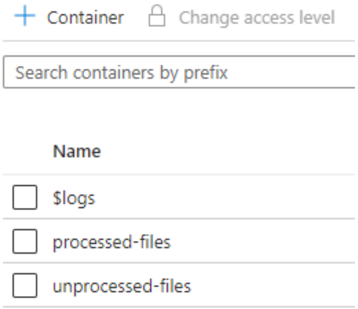
6. Select **Review + Create**. On the validation page that appears, confirm all the information that you provided, and select **Create.**
7. After the deployment is completed. Select **Go to resource**.



Your deployment is complete

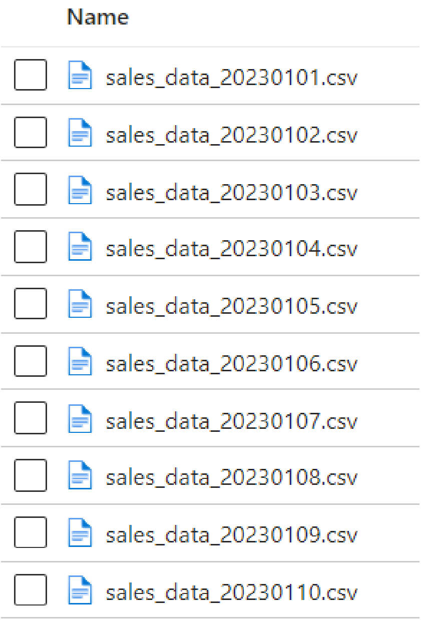
Deployment name: storageaccount21171333_1724027383081
Subscription: Azure for Students
Resource group: BUS5001-Week4-RG

8. In the storage account, navigate to **Containers** under Data storage. Add two containers named *unprocessed-files and processed-files*. Access level can be default Private for both.



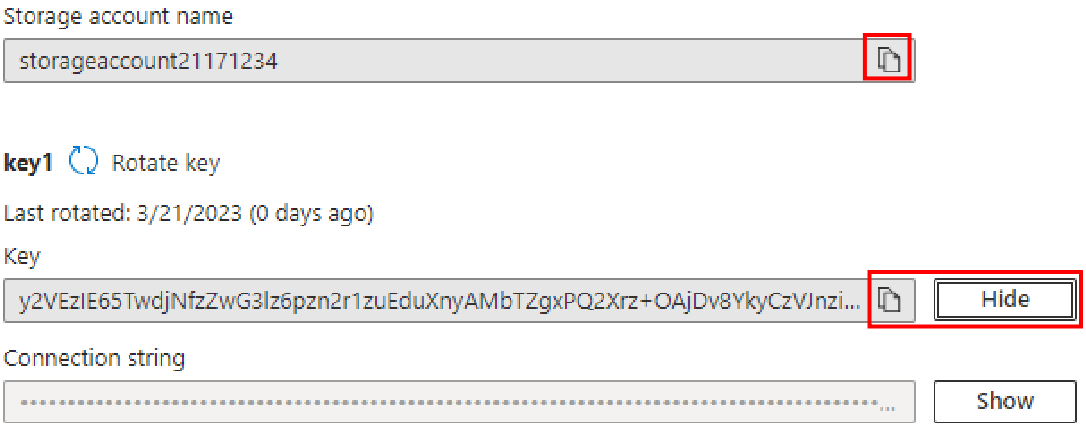
	Name
□	$logs
□	processed-files
□	unprocessed-files

9. Upload the csv files that you downloaded from LMS, into unprocessed-files container.



	Name
□	sales_data_20230101.csv
□	sales_data_20230102.csv
□	sales_data_20230103.csv
□	sales_data_20230104.csv
□	sales_data_20230105.csv
□	sales_data_20230106.csv
□	sales_data_20230107.csv
□	sales_data_20230108.csv
□	sales_data_20230109.csv
□	sales_data_20230110.csv

10. On the left navigation panel, select **Access keys** under Security + networking.
11. In a future step you will need the Storage account name and an access key. You will access them here.



Storage account name
storageaccount21171234

key1 Rotate key

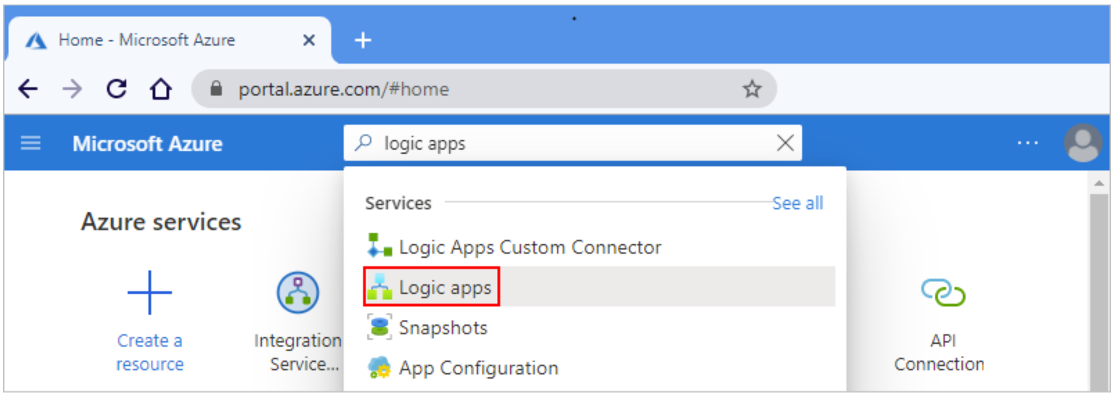
Last rotated: 3/21/2023 (0 days ago)

Key
y2VEzIE65TwdjNfzZwG3lz6pzn2r1zuEduXnyAMbTZgxPQ2Xrz+OAjDv8YkyCzVJnzi...

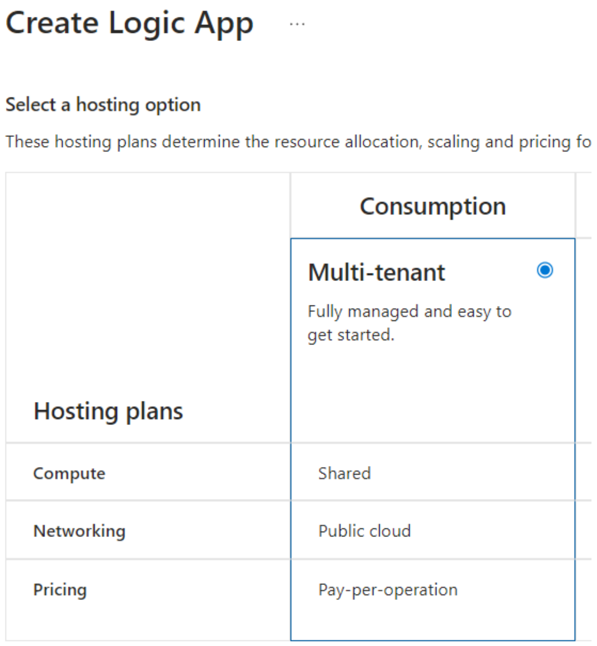
Connection string
••...

Create a Logic App resource

12. In the Azure search box, enter logic apps, and select Logic apps.



13. On the Logic apps page, select **+ Add**.
14. On the **Create Logic App** pane, select **Consumption, Multi-tenant** option and click **Select**.



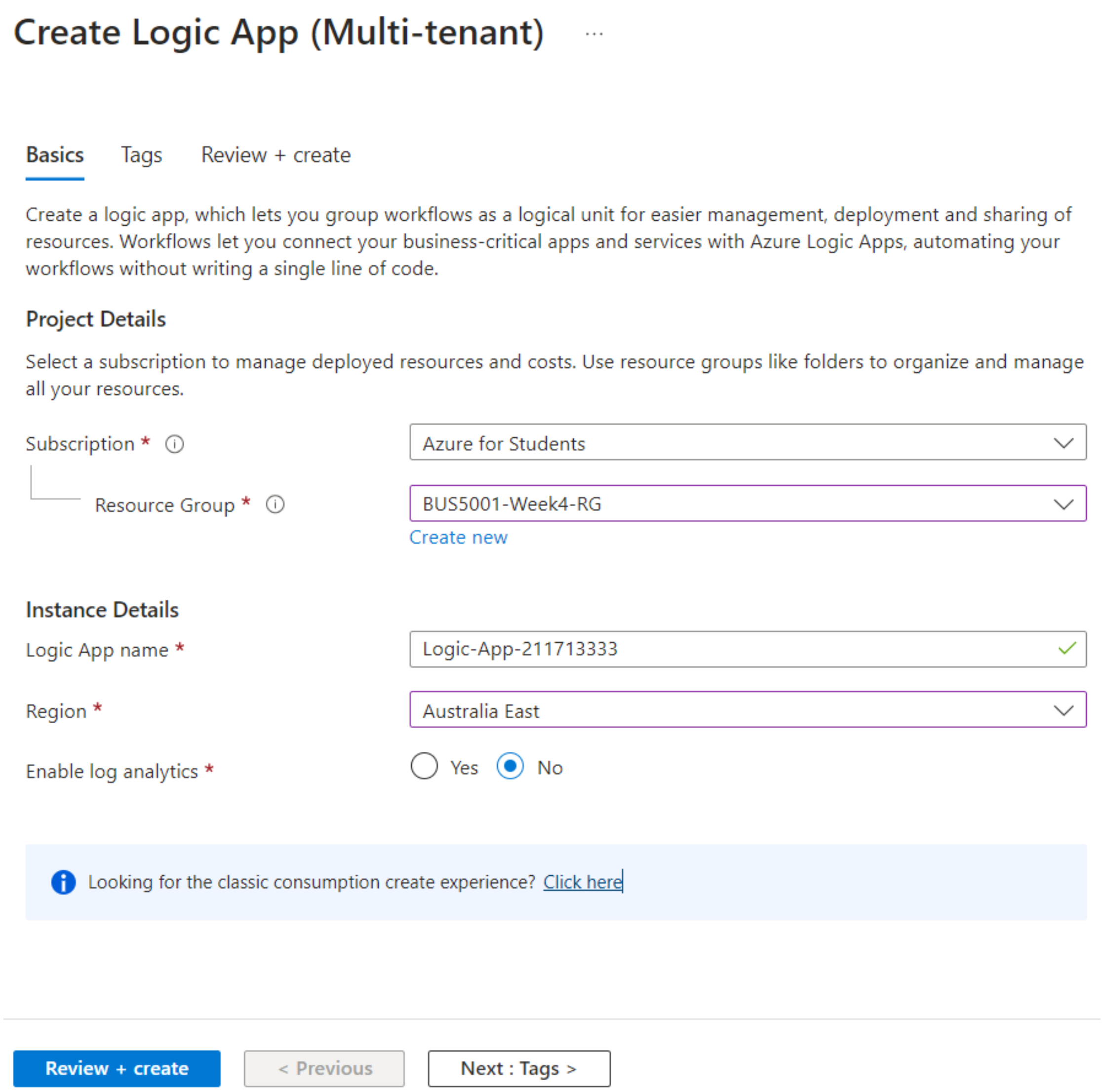
Create Logic App ···

Select a hosting option

These hosting plans determine the resource allocation, scaling and pricing fo

Hosting plans	Consumption Multi-tenant ● Fully managed and easy to get started.
Compute	Shared
Networking	Public cloud
Pricing	Pay-per-operation

15. On the **Create Logic App** pane, on the **Basics** tab, provide the following basic information about your logic app:
    a. **Subscription**: Your Azure subscription name. *Ex: Azure for students*
    b. **Resource Group**: Select the resource group that you used with the storage account.
    c. **Logic App name:** *Ex: storageaccount<student_id>*. (can contain only letters, numbers, hyphens (-), underscores (_), parentheses ((,)), and periods (.) and is unique across all regions). *Ex: Logic-App-<student_id>.*
    d. **Region:** Australia East
    e. **Enable log analytics:** No
16. When you're done, your settings look similar to the following:



Create Logic App (Multi-tenant)

Create a logic app, which lets you group workflows as a logical unit for easier management, deployment and sharing of resources. Workflows let you connect your business-critical apps and services with Azure Logic Apps, automating your workflows without writing a single line of code.

Project Details

Select a subscription to manage deployed resources and costs. Use resource groups like folders to organize and manage all your resources.

Subscription * : Azure for Students

Resource Group * : BUS5001-Week4-RG

Instance Details

Logic App name * : Logic-App-211713333

Region * : Australia East

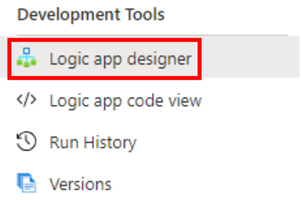
Enable log analytics * : ○ Yes ● No

Looking for the classic consumption create experience? Click here

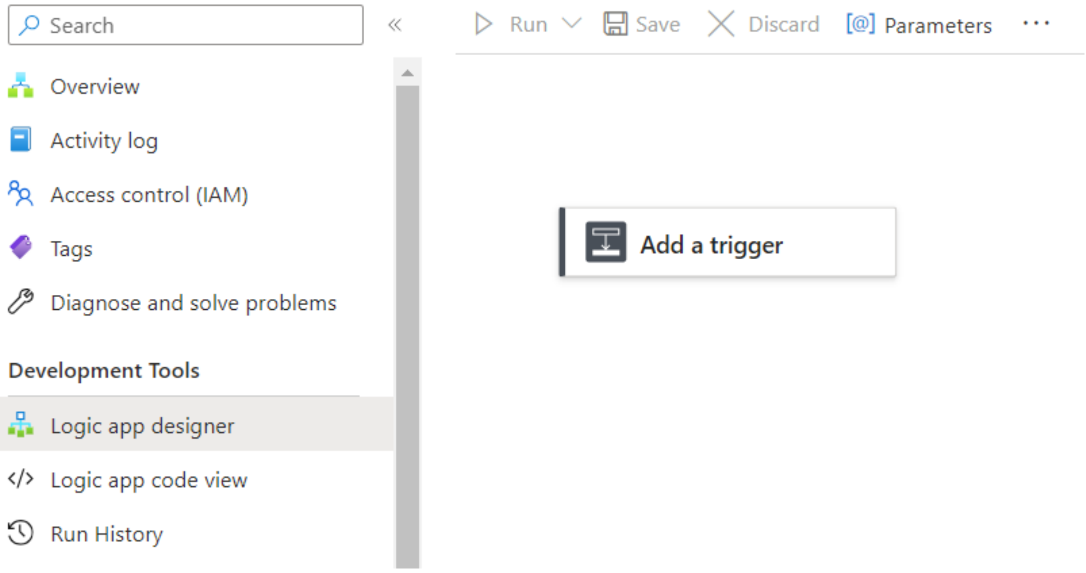
17. Select **Review + Create**.
18. On the validation page that appears, confirm all the information that you provided, and select **Create**.

Select the blank template

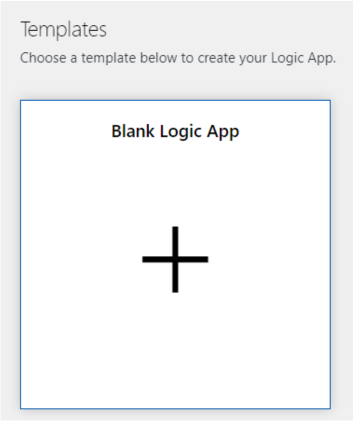
1. After Azure successfully deploys your app, select **Go to resource**. Or find and select your logic app resource by typing the name in the Azure search box.
2. Select **Logic app designer** under **Development Tools** on the left navigation pane.

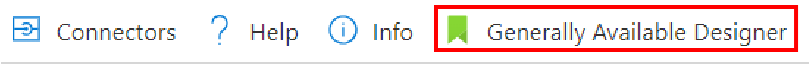


3. The designer should show an empty workflow where you can add the trigger.



<u>If not</u>, scroll down and select **Blank Logic App** under Templates, Select **Generally Available Designer** from the upper pane.

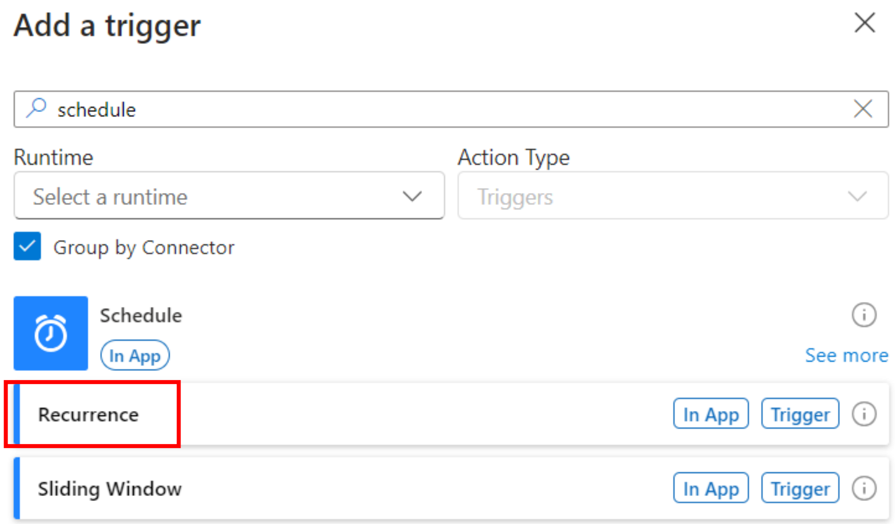




Add a Recurrence trigger

This workshop uses a Recurrence trigger that activates at specified intervals. At each scheduled interval, the trigger starts a new instance of the workflow, ensuring the automation runs according to the predefined frequency. This allows the workflow to execute regularly, regardless of any new input, making it ideal for scheduled tasks that need to run periodically.

1. Select **Add a trigger** option.
2. In the search box, enter **’schedule’**. Select **Recurrence** under Schedule.



Add a trigger

schedule

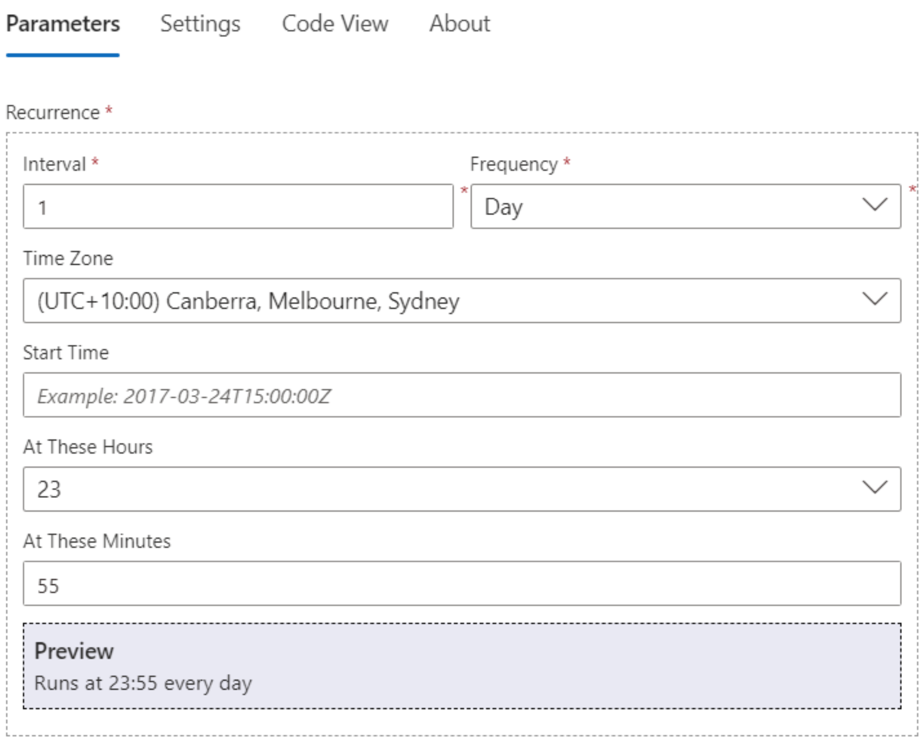
Runtime: Select a runtime

Action Type: Triggers

☑ Group by Connector

3. In the **Parameters** pane, enter following details,

Property	**Required**	**Value**
Interval	Yes	1
Frequency	Yes	Day
Time Zone	No	UTC+ 10:00
At These Hours	No	23
At These Hours	No	55



Recurrence *

Interval * 1

Frequency * Day

Time Zone (UTC+10:00) Canberra, Melbourne, Sydney

Start Time *Example: 2017-03-24T15:00:00Z*

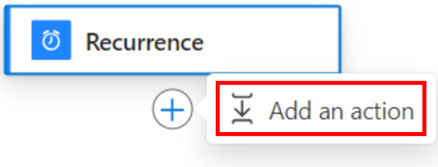
At These Hours 23

At These Minutes 55

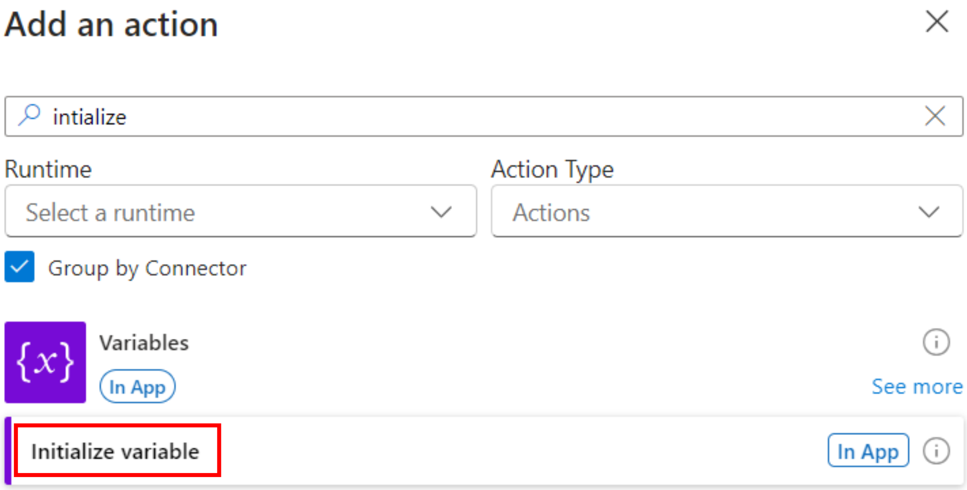
Preview
Runs at 23:55 every day

Parse the CSV files to JSON and calculate the total sales

1. Under the **Recurrence** trigger, select **+**, then **Add an action.**



2. In the search box, enter **'initialize'** and select **Initialize variable** under Variables.



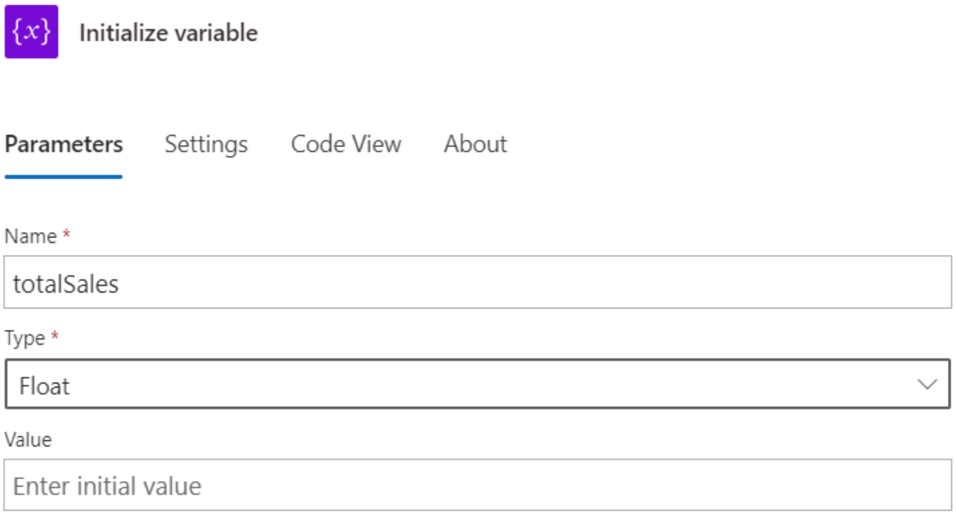
intialize

Runtime: Select a runtime

Action Type: Actions

☑ Group by Connector

3. Enter following values to the fields,
   a. Name: **totalSales**
   b. Type: **Float**

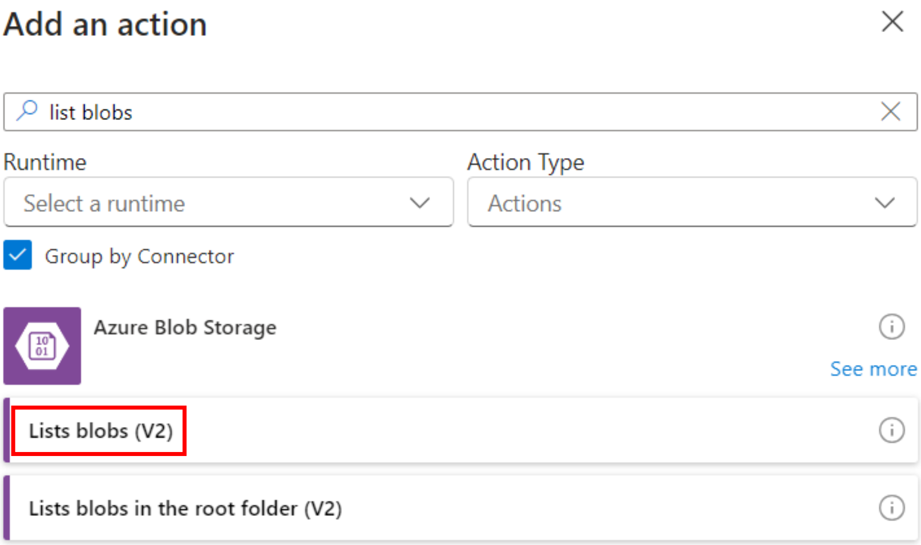


Name *: totalSales

Type *: Float

Value: Enter initial value

4. Under the previous action, select **+** and then **Add an action**.
5. Search for **'list blobs'** and select **List blobs (V2)** under **Azure Blob Storage**.



Add an action

list blobs

Runtime: Select a runtime

Action Type: Actions

☑ Group by Connector

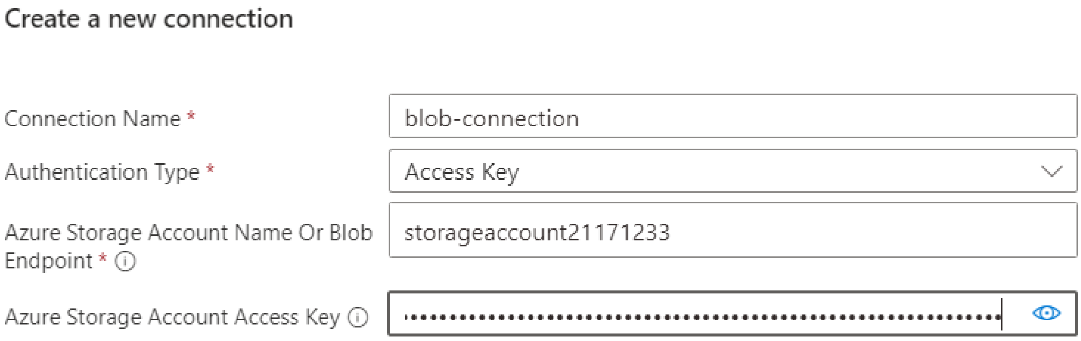
Azure Blob Storage

See more

Lists blobs (V2)

Lists blobs in the root folder (V2)

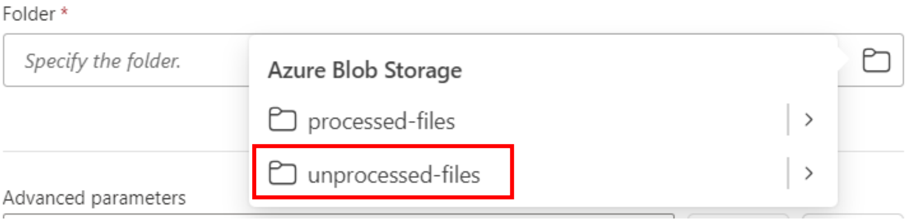
6. In the Azure Blob Storage action, change **Authentication type** to **Access Key**.
7. Enter following values to the fields,
    a. **Connection Name**: Name for the connection to the storage account. *Ex: blob-connection*.
    b. **Authentication Type**: **Access Key**
    c. **Azure Storage Account Name or Blob Endpoint**: Storage account name that you created. ('Create Storage Account' section step 10)
    d. **Azure Storage Account Access Key**: Storage Account Access Key. ('Create Storage Account' section step 10)



Create a new connection

Connection Name *	blob-connection
Authentication Type *	Access Key
Azure Storage Account Name Or Blob Endpoint * ⓘ	storageaccount21171233
Azure Storage Account Access Key ⓘ	••

8. Select **Create New**.
9. In the **List blobs (V2)** action,
    a. For **Storage account name or blob endpoint,** select the connection settings for the created storage account.
    b. For **Folder**, click Open folder icon and select **unprocessed-files.**



Folder *

Specify the folder.

Azure Blob Storage

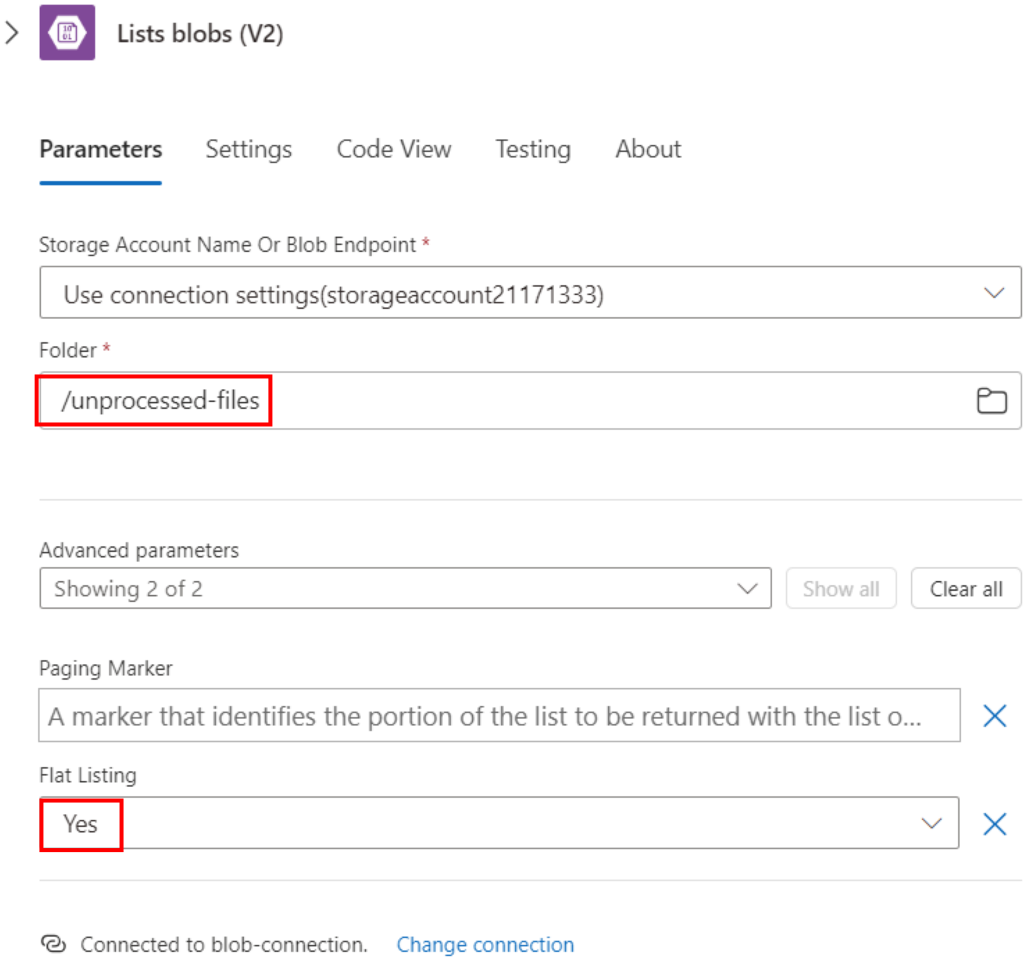
processed-files

unprocessed-files

Advanced parameters

    c. Select **Yes** for **Flat Listing**.

10. Final action window should look like the following.



Lists blobs (V2)

Storage Account Name Or Blob Endpoint *

Use connection settings(storageaccount21171333)

Folder *

/unprocessed-files

Advanced parameters

Showing 2 of 2

Paging Marker

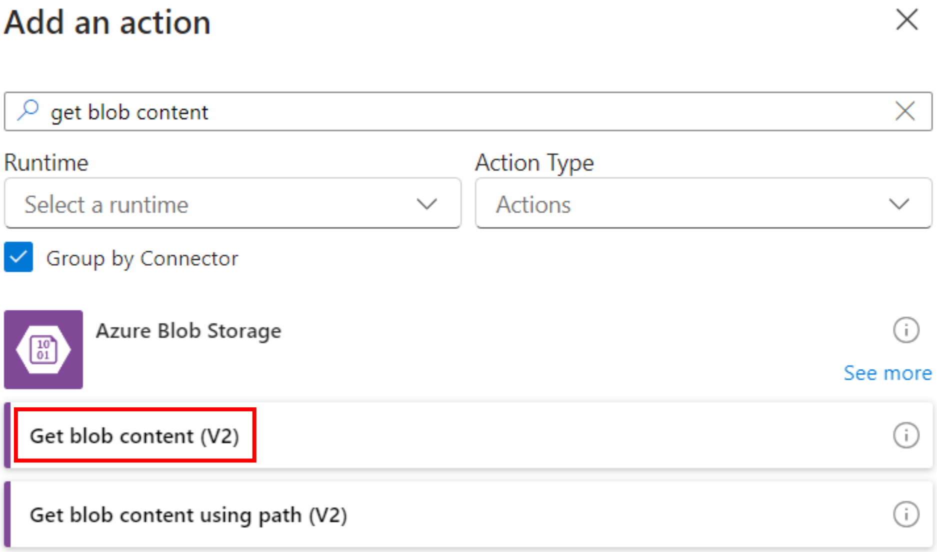
A marker that identifies the portion of the list to be returned with the list o...

Flat Listing

Yes

Connected to blob-connection. Change connection

11. Under the previous action, select **+** and then **Add an action**.
12. Search for ‘Get blob content’ and select **Get blob content (V2)** under **Azure Blob Storage**.



Add an action

get blob content

Runtime

Select a runtime

Action Type

Actions

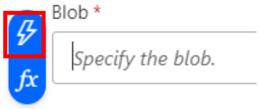
☑ Group by Connector

Azure Blob Storage

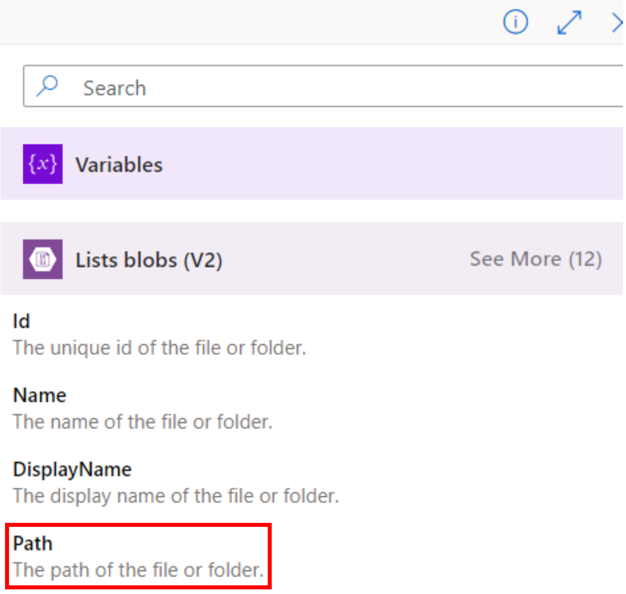
Get blob content (V2)

Get blob content using path (V2)

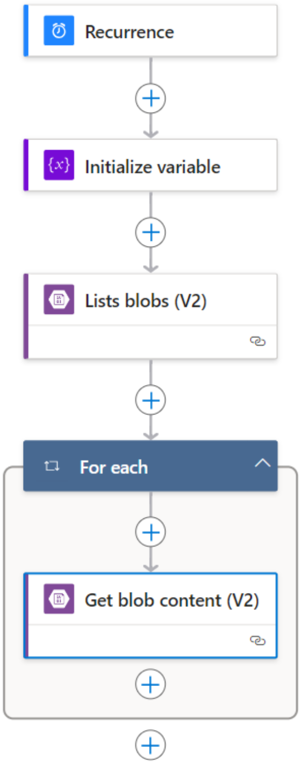
13. In the **Get blob content (V2)** action,
    a. For **Storage account name or blob endpoint,** select the connection settings for the created storage account.
    b. For **Blob,** open Dynamic content window, clicking the icon.



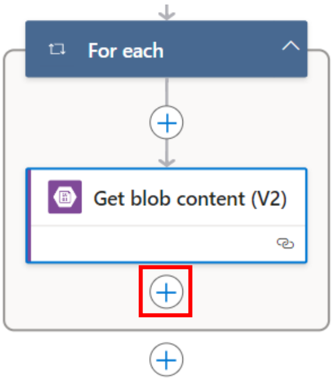
    c. Select **Path** under List blobs (V2).


Search
Variables
Lists blobs (V2)
See More (12)
Id
The unique id of the file or folder.
Name
The name of the file or folder.
DisplayName
The display name of the file or folder.
Path
The path of the file or folder.

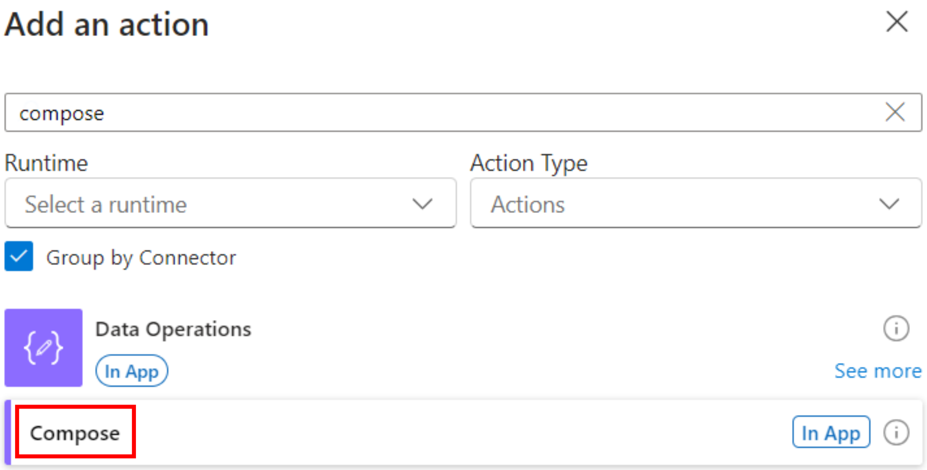
    d. You will see now this resulted in a **For each** in the workflow. The work flow should look similar to the following.



14. Inside the **For each**, select **+** and then **Add an action.**



15. Search for **'compose'** and select **Compose** under **Data Operations**.


Add an action

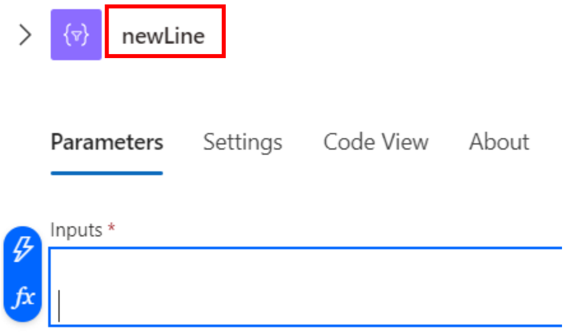
compose

Runtime: Select a runtime

Action Type: Actions

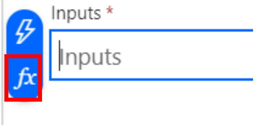
☑ Group by Connector

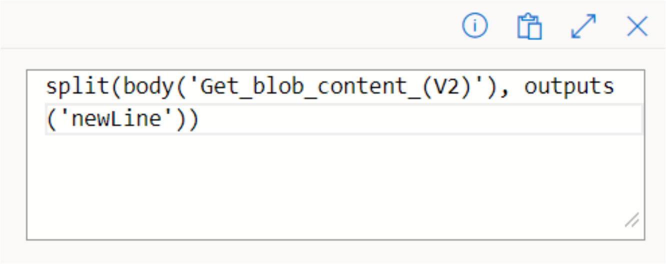
16. Rename the action to **newline** and add a new line (press enter) in **Inputs** field.



17. Inside the **For each**, select **+** and then **Add an action.**
18. Add another compose action and rename it to **splitByLines**.
19. For the inputs select expressions and add the following expression and click **Add**.

```
split(body('Get_blob_content_(V2)'), outputs('newLine'))
```





```
split(body('Get_blob_content_(V2)'), outputs
('newLine'))
```

20. Inside the **For each**, select **+** and then **Add an action**.
21. Add another compose action and rename it to **fieldNames**.
22. Add the following expression for the Inputs field.

```
split(first(outputs('splitByLines')), ',')
```

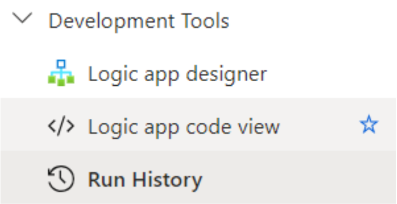
23. Inside the **For each**, select **+** and then **Add an action.**
24. Search for **'select'** and select **Select** under Data operations.
25. In the parameters pane, add following expressions.
    a. For **From**: `skip(outputs('splitByLines'), 1)`
    b. Under Map,

`outputs('fieldNames')[0]`	`split(item(), ',')?[0]`
`outputs('fieldNames')[1]`	`split(item(), ',')?[1]`
`outputs('fieldNames')[2]`	`split(item(), ',')?[2]`

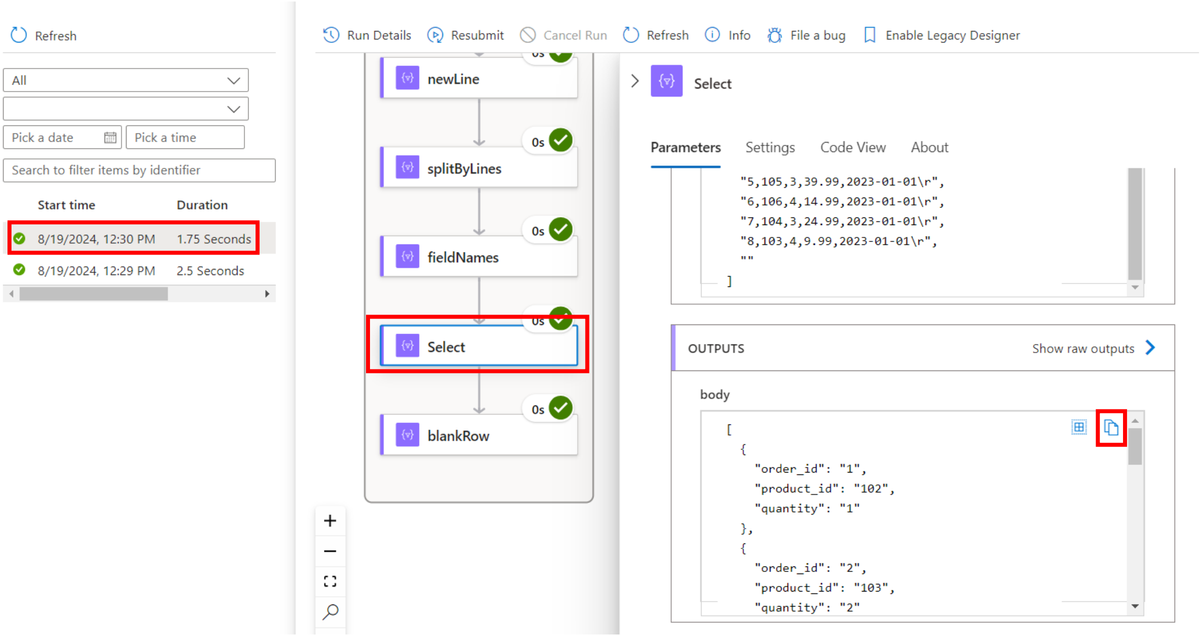
26. Inside the For each, select **+** and then **Add an action**.
27. Add another compose action and rename it to **blankRow**.
28. For the Inputs field, add the following expression,

```
take(body('Select'), sub(length(body('Select')), 1))
```

29. Click **Save** and run the workflow.
30. On the left pane, select **Run History** under **Development Tools**.



31. Select the successfully executed run and select the **Select** action.
32. In the **Parameters** pane, copy the content in the **Outputs** field.

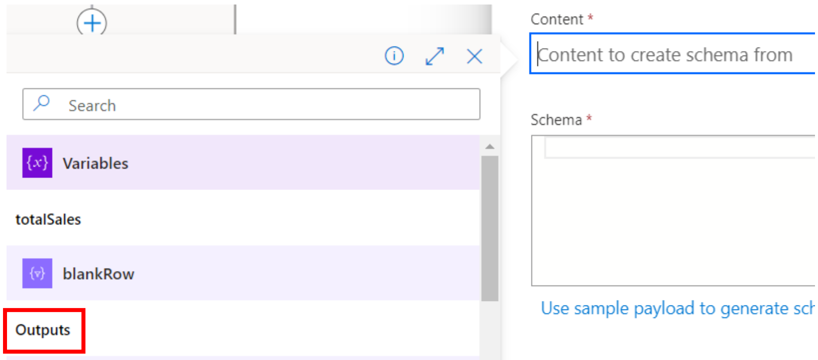


Start time	Duration
8/19/2024, 12:30 PM	1.75 Seconds
8/19/2024, 12:29 PM	2.5 Seconds

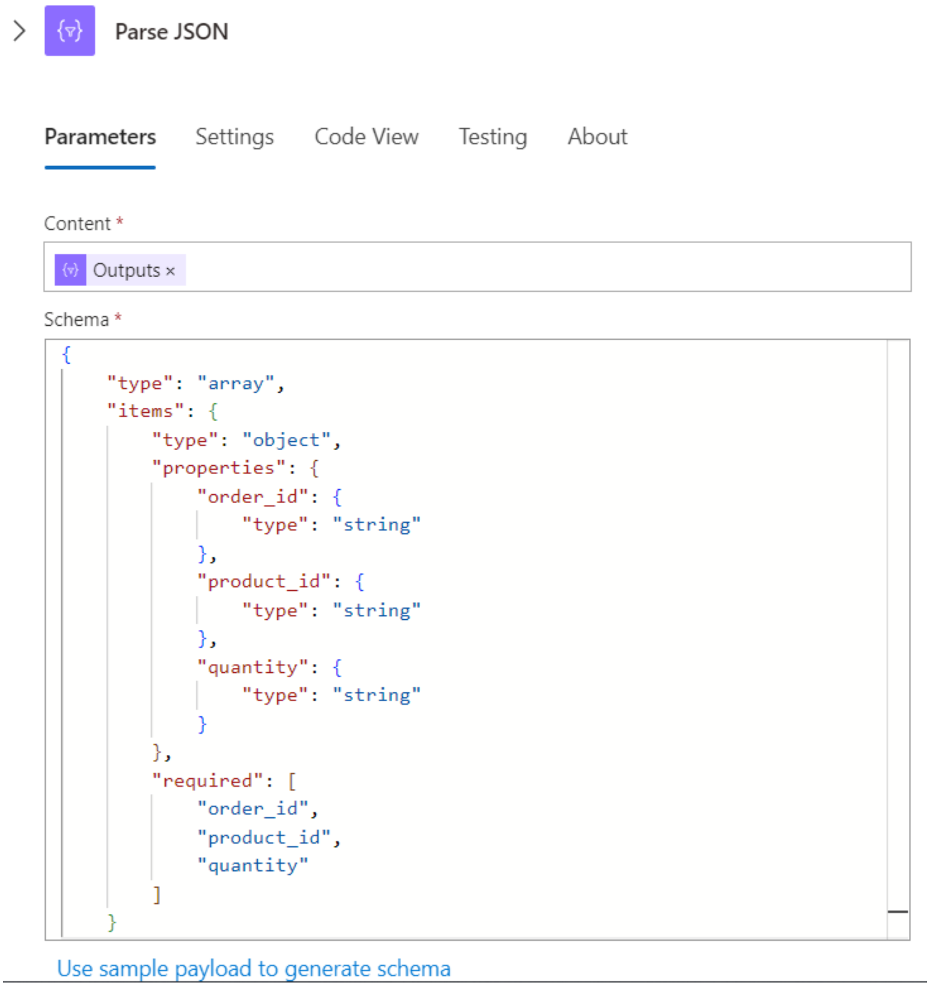
```
"5,105,3,39.99,2023-01-01\r",
"6,106,4,14.99,2023-01-01\r",
"7,104,3,24.99,2023-01-01\r",
"8,103,4,9.99,2023-01-01\r",
""
]
```

```
[
  {
    "order_id": "1",
    "product_id": "102",
    "quantity": "1"
  },
  {
    "order_id": "2",
    "product_id": "103",
    "quantity": "2"
```

33. On the left pane, select **Logic app designer** to return to the designer canvas.
34. Inside the For each, select **+** and then **Add an action**.
35. Search for **'parse json'** and select **Parse JSON** under **Data operations**.
36. In the Parameters pane,
    a. For **Content**: in dynamic content window, select Outputs under blankRow.


Search
Variables
totalSales
blankRow
Outputs
Content *
Content to create schema from
Schema *
Use sample payload to generate sch

    b. Select **Use sample payload to generate schema**.
    c. Paste the content that you copied in a previous step and click **Done**.
    Parse JSON action should look similar to the following,



Content *

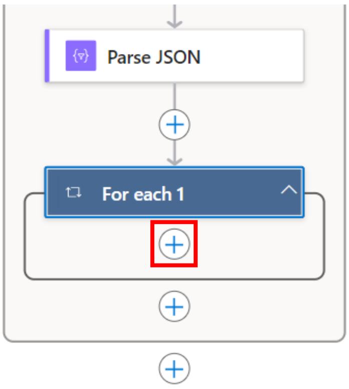
Outputs ×

Schema *

```
{
    "type": "array",
    "items": {
        "type": "object",
        "properties": {
            "order_id": {
                "type": "string"
            },
            "product_id": {
                "type": "string"
            },
            "quantity": {
                "type": "string"
            }
        },
        "required": [
            "order_id",
            "product_id",
            "quantity"
        ]
    }
```

Use sample payload to generate schema

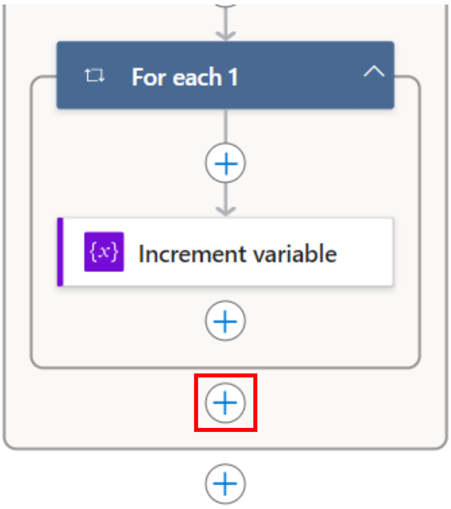
37. Inside the For each, select **+** and then **Add an action**.
38. Search for **‘for each’** and select **For each** under **Control**.
39. Select the second For each action added in the previous step, for the **Select An Output From Previous Steps** field, open dynamic content window and select **Body** under **Parse JSON**.
40. Inside the second For each, select **+** and then **Add an action**.



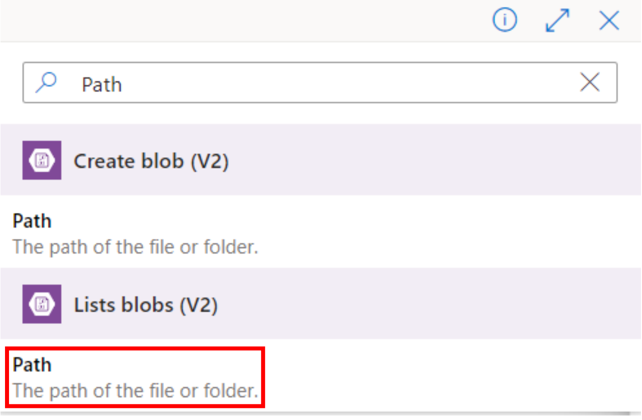
41. Search for **‘increment variable’** and select **Increment variable** under **Variables**.
42. In the parameters pane,
    a. For **Name**: select **totalSales** from the dropdown.

   b. For **Value**: enter the following expression, `float(items('For_each_1')['quantity'])`

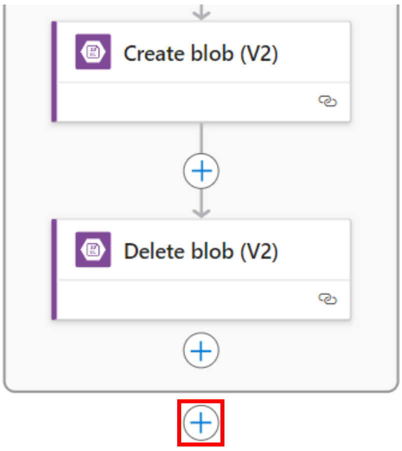
43. Inside the first for each, select **+** and then **Add an action.**



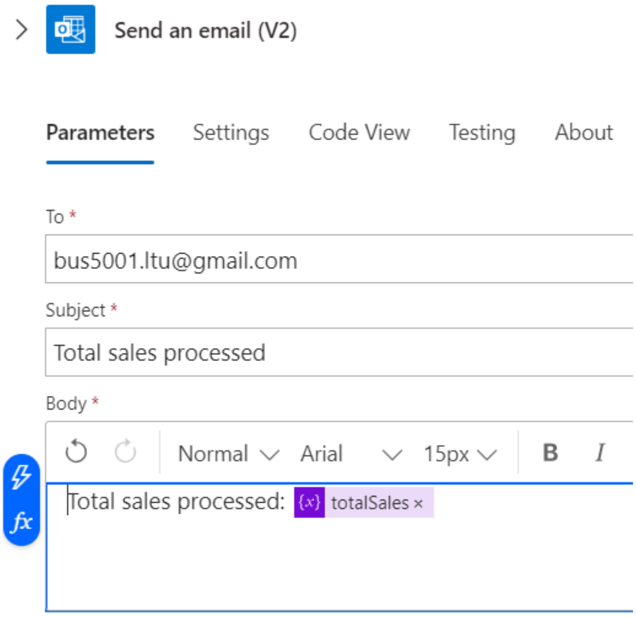
44. Search for **'create blob'** and select **Create blob (V2)** under **Azure Blob Storage.**
45. In the **Create Blob (V2)** action, on the **Parameters** pane,
    a. For **Storage account name or blob endpoint**, select the connection settings for the created storage account.
    b. For **Folder Path**, select **processed-files** container.
    c. For **Blob Name**, open the dynamic content window and select **Display Name** under **Lists Blobs (V2)**.
    d. For Blob Content, open the dynamic content window and select **File Content** under **Get blob content (V2)**.
46. Inside the first for each, select **+** and then **Add an action.**
47. Search for **'delete blob'** and select **Delete blob (V2)** under **Azure Blob Storage.**
48. In the **Delete Blob (V2)** action, on the **Parameters** pane,
    a. For **Storage account name or blob endpoint**, select the connection settings for the created storage account.
    b. For Blob, open dynamic content window and select **Path** under **Lists blobs (V2).** **IMPORTANT**: Make sure you select under **Lists blobs (V2)** not Create blob (V2).


Path
Create blob (V2)
Path
The path of the file or folder.
Lists blobs (V2)
Path
The path of the file or folder.

49. Outside the for each, select **+** and then **Add an action.**



50. Search for **'send an email'** and select **Send an email (V2)** under **Office 365 Outlook.**
51. **Sign in** using your student account.
52. In **Send an email (V2)** action, under **Parameters** pane,
    a. For **To**, enter the receiver's email address (except for student email, you can use a gmail address).
    b. For **Subject**, enter 'Total Sales Processed'
    c. For **Body**, enter 'Total Sales value: ', open dynamic content window and select totalSales under Variables.

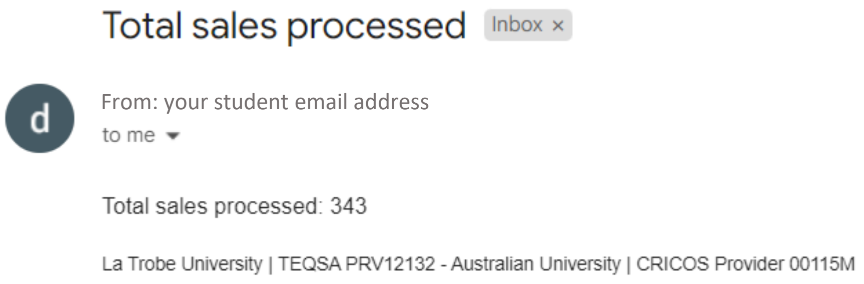


To *
bus5001.ltu@gmail.com
Subject *
Total sales processed
Body *
Total sales processed: totalSales ×

53. Click **Save** and run your workflow.

Run your workflow

After a successful run, the csv files in the **unprocessed-files** container should be moved to **processed-files** container and you should receive an email mentioning the total sales values to the Gmail account that you entered.



Total sales processed

From: your student email address

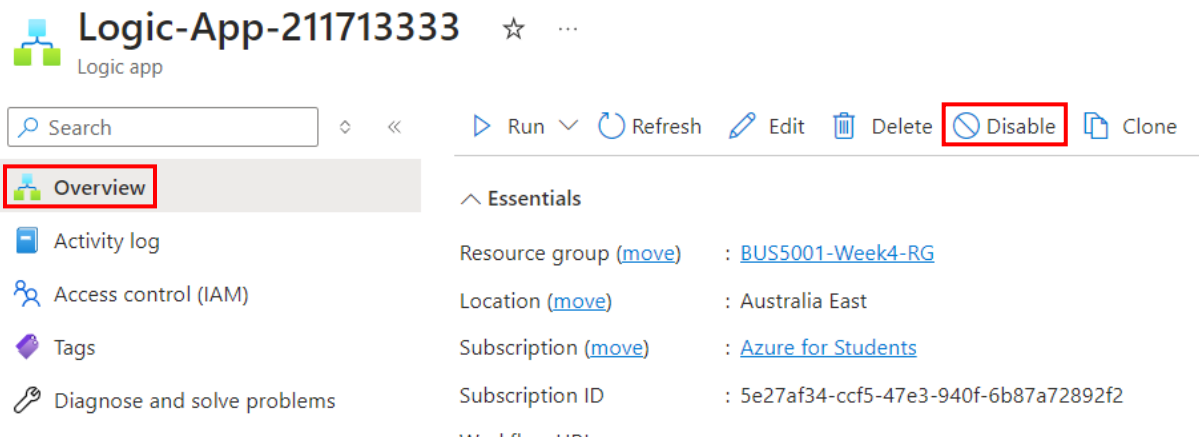
to me

Total sales processed: 343

La Trobe University | TEQSA PRV12132 - Australian University | CRICOS Provider 00115M

Clean up resources

IMPORTANT When you're done with this workshop, delete/disable the logic app resource and any related resources to prevent unnecessary cost.



Logic-App-211713333

Logic app

Resource group (move) : BUS5001-Week4-RG

Location (move) : Australia East

Subscription (move) : Azure for Students

Subscription ID : 5e27af34-ccf5-47e3-940f-6b87a72892f2