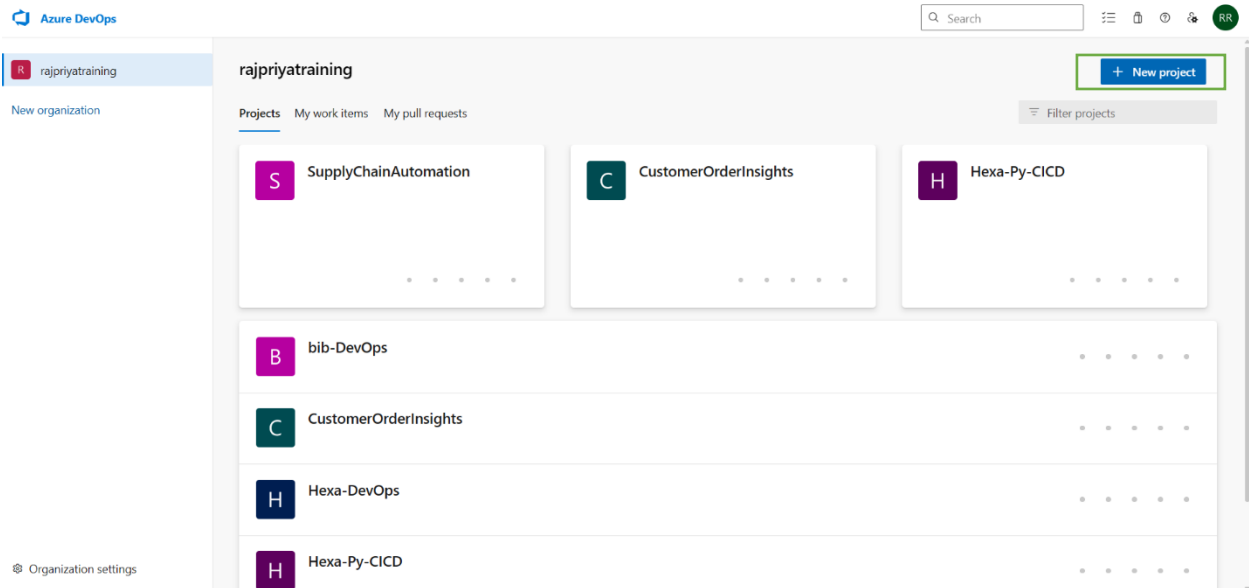


Pictorial Documentation to build a CI/CD pipeline

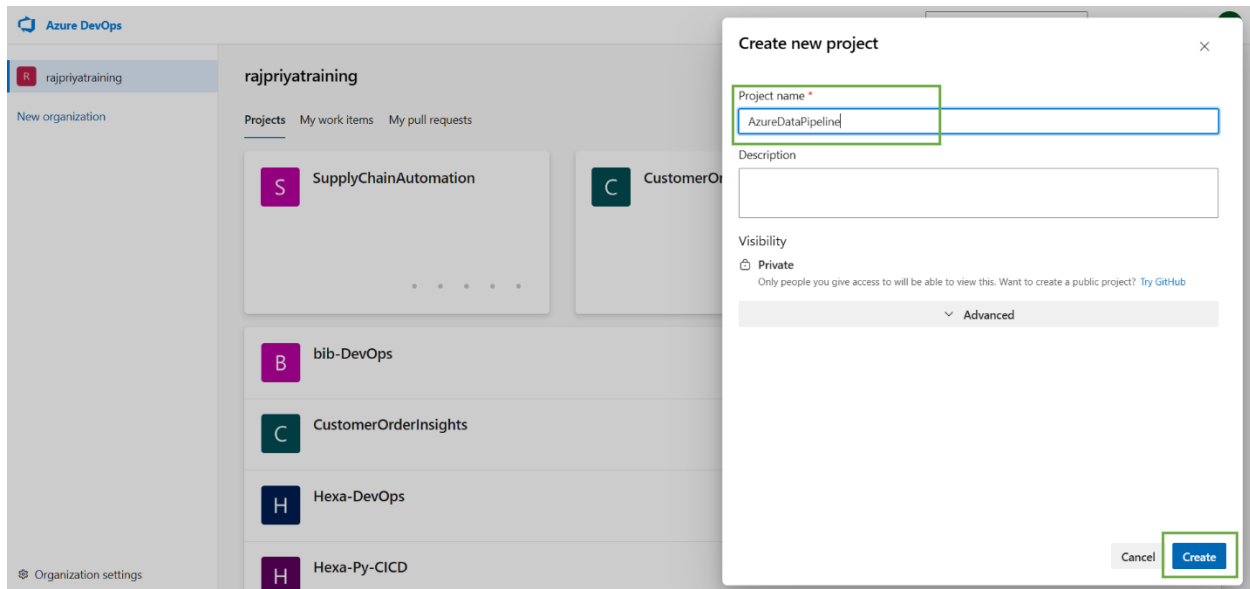
Step 1:

Create a New Project.



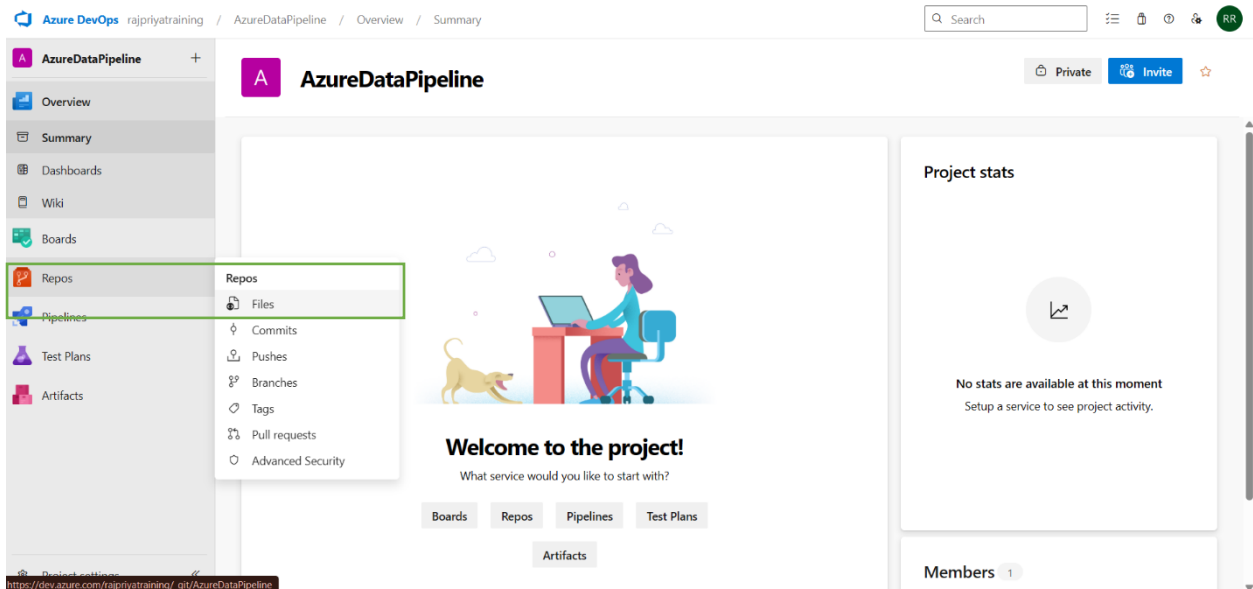
Step 2:

Create New Project -> AzureDataPipeline -> Create.



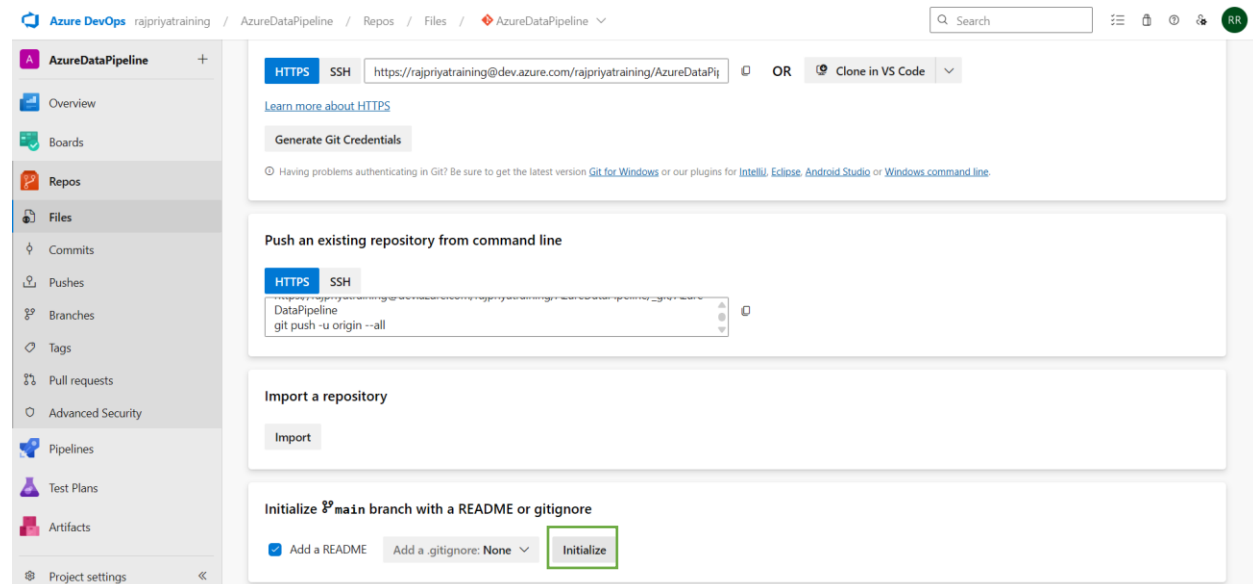
Step 3:

Go to Repos -> Files.



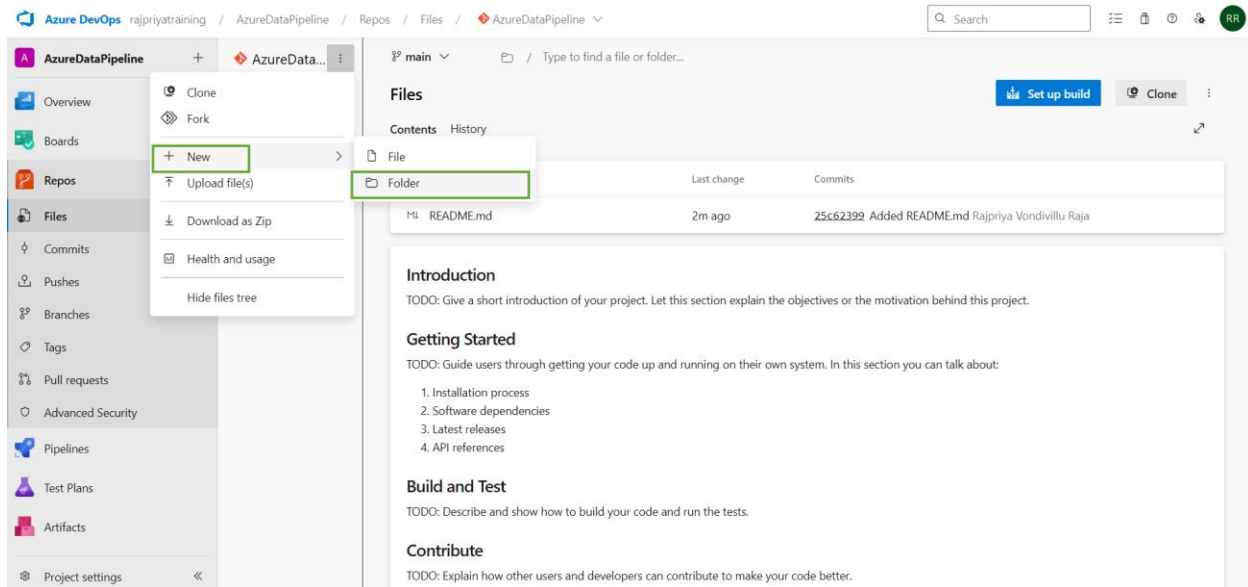
Step 4:

Click on Initialize.



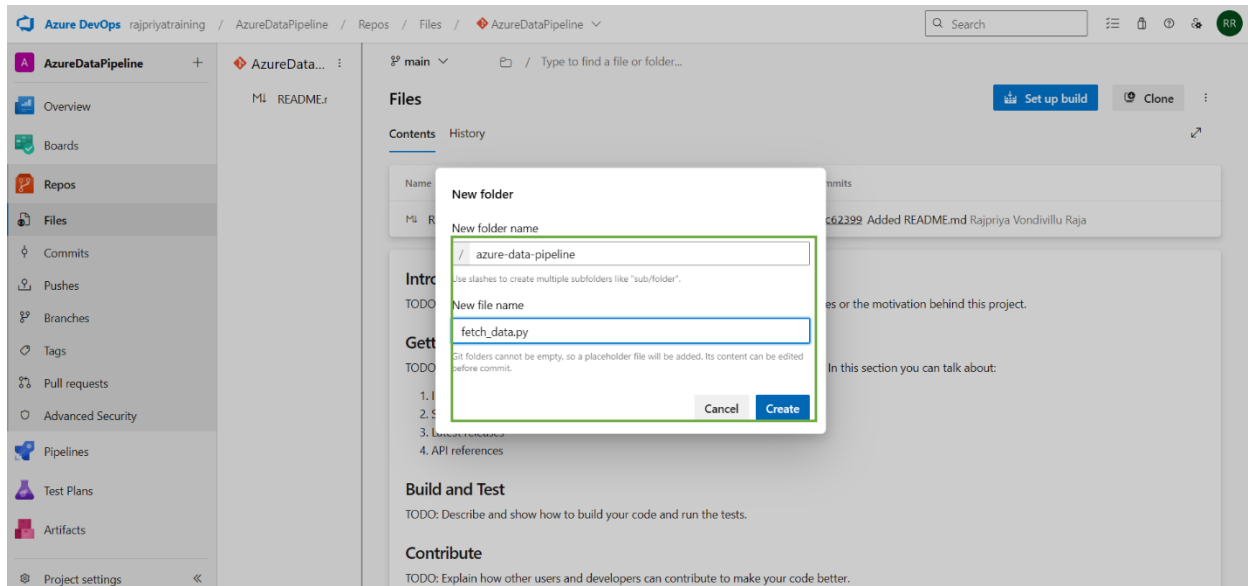
Step 5:

Click on the New -> Folder.



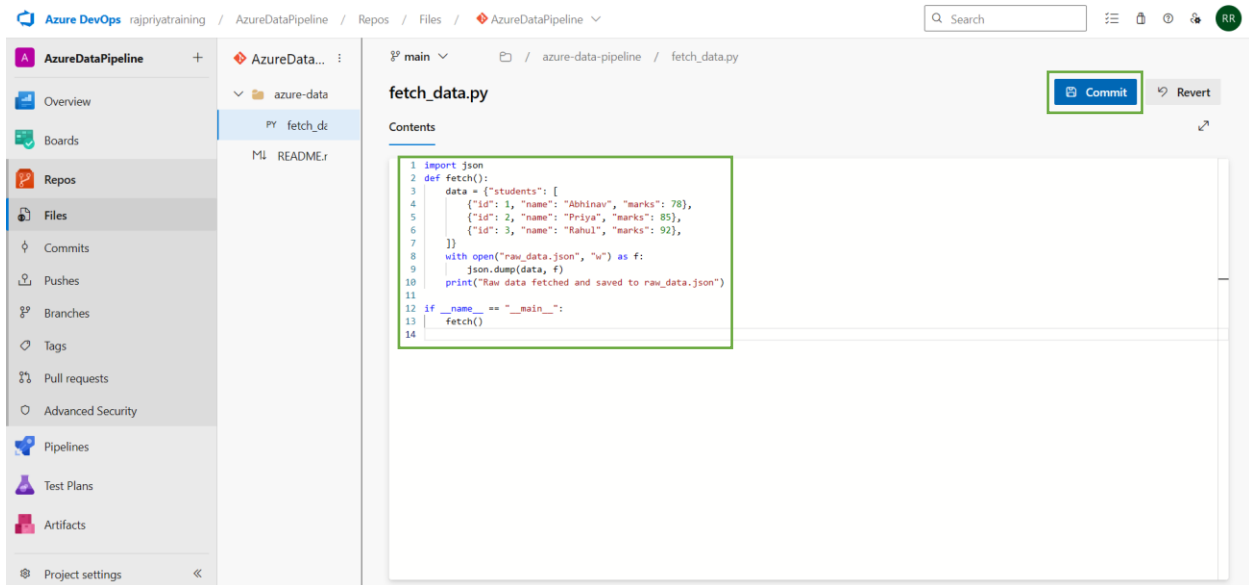
Step 6:

Write your FolderName and FileName -> Create.



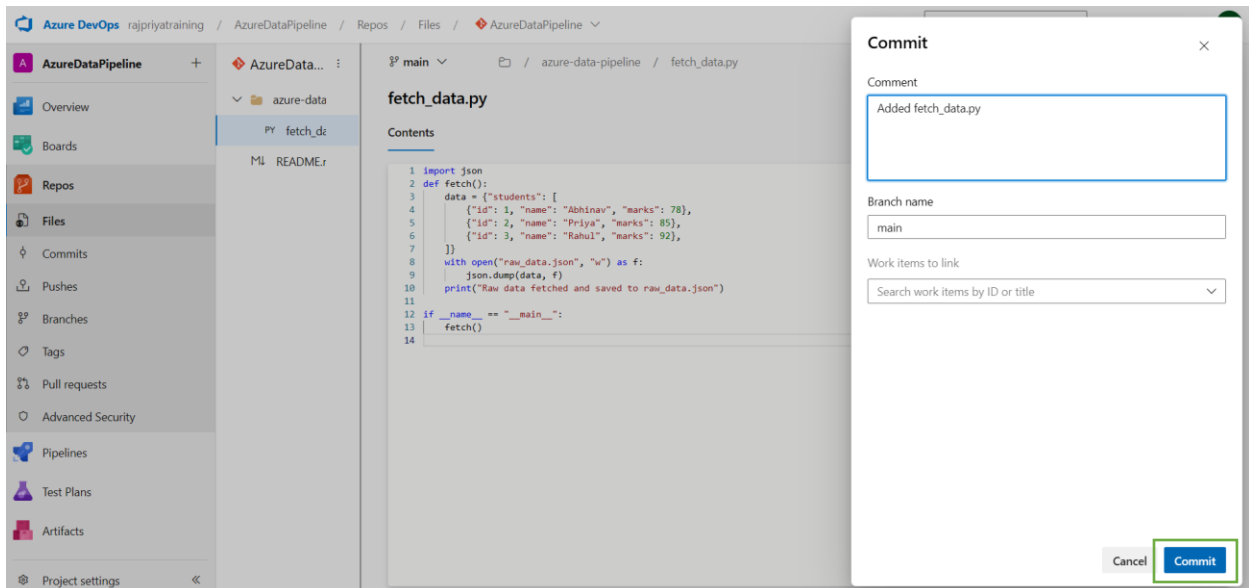
Step 7:

Create a “fetch_data.py” file, add your code -> Click Commit.



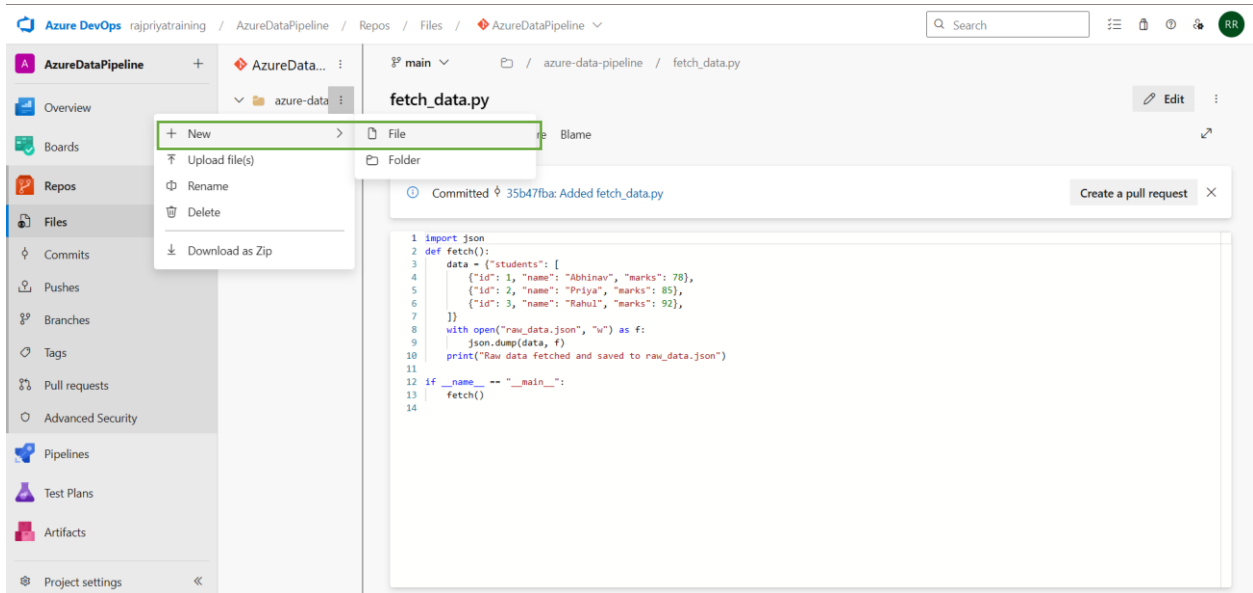
Step 8:

Again, Click on Commit.



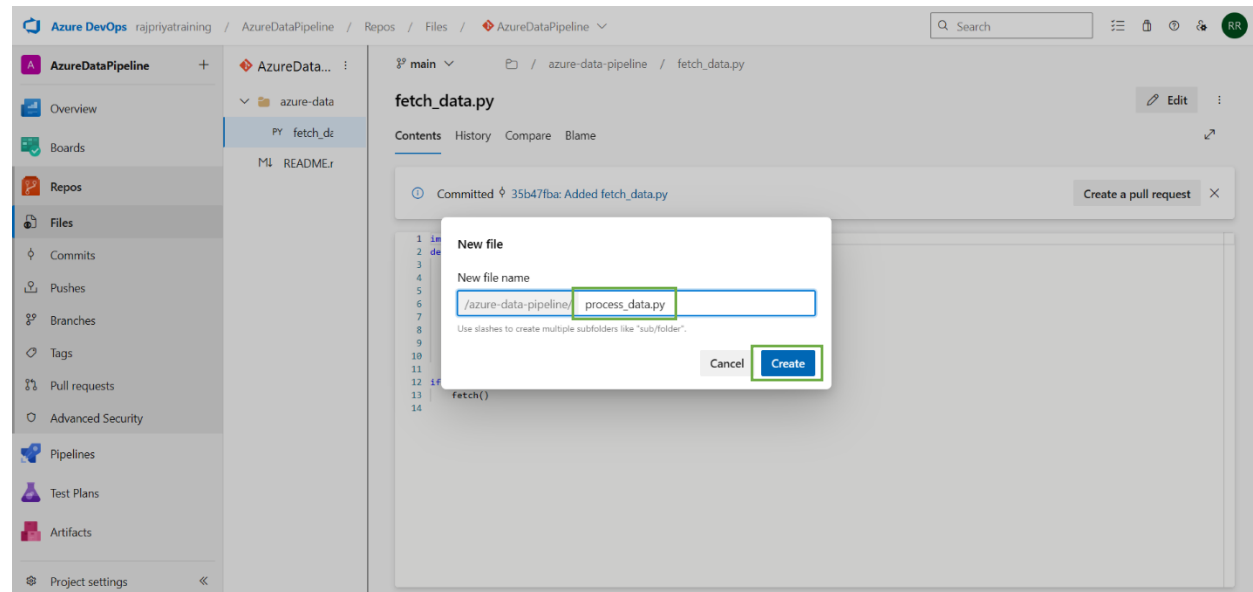
Step 9:

Go to your Folder -> New -> File.



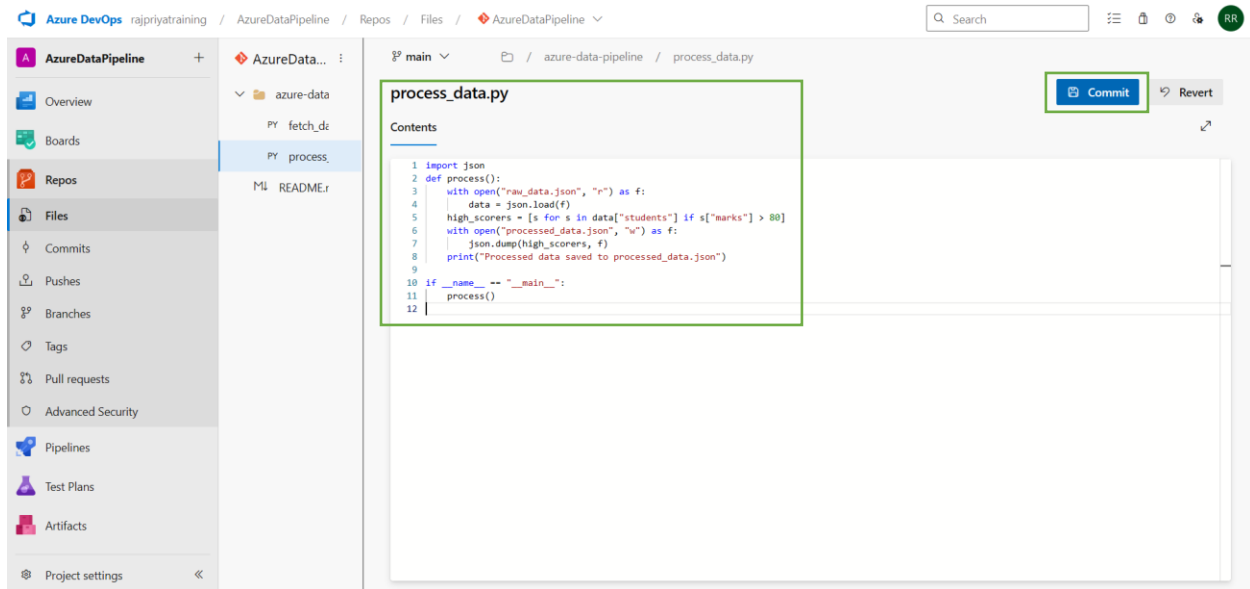
Step 10:

Write your file name and click Create.



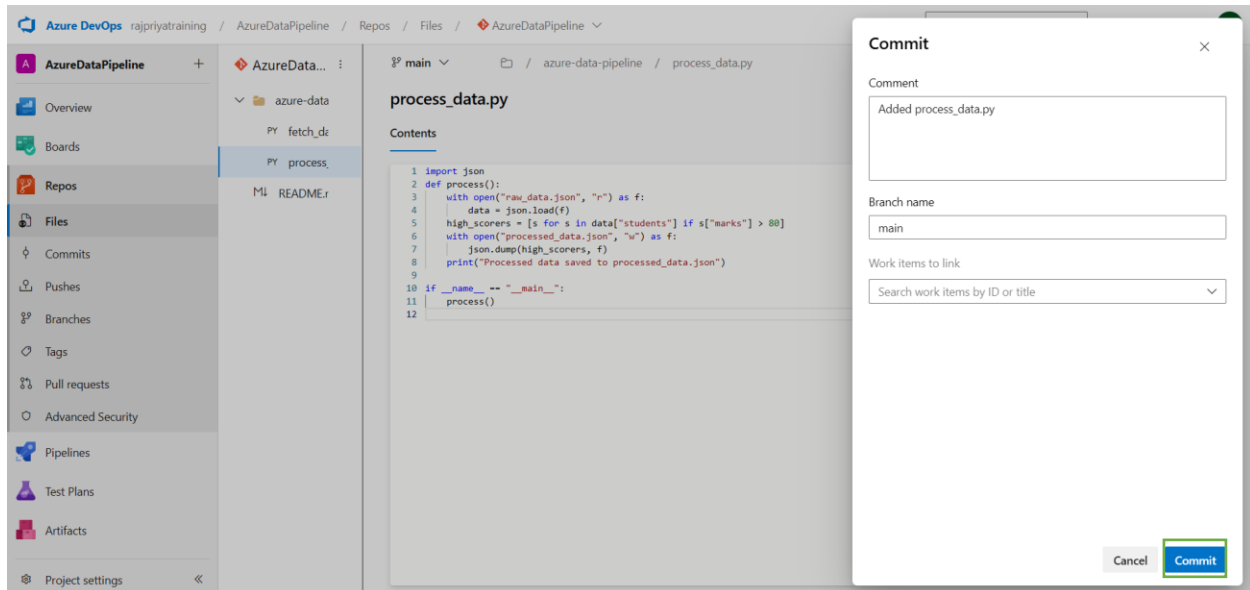
Step 11:

Write your code and click on Commit.



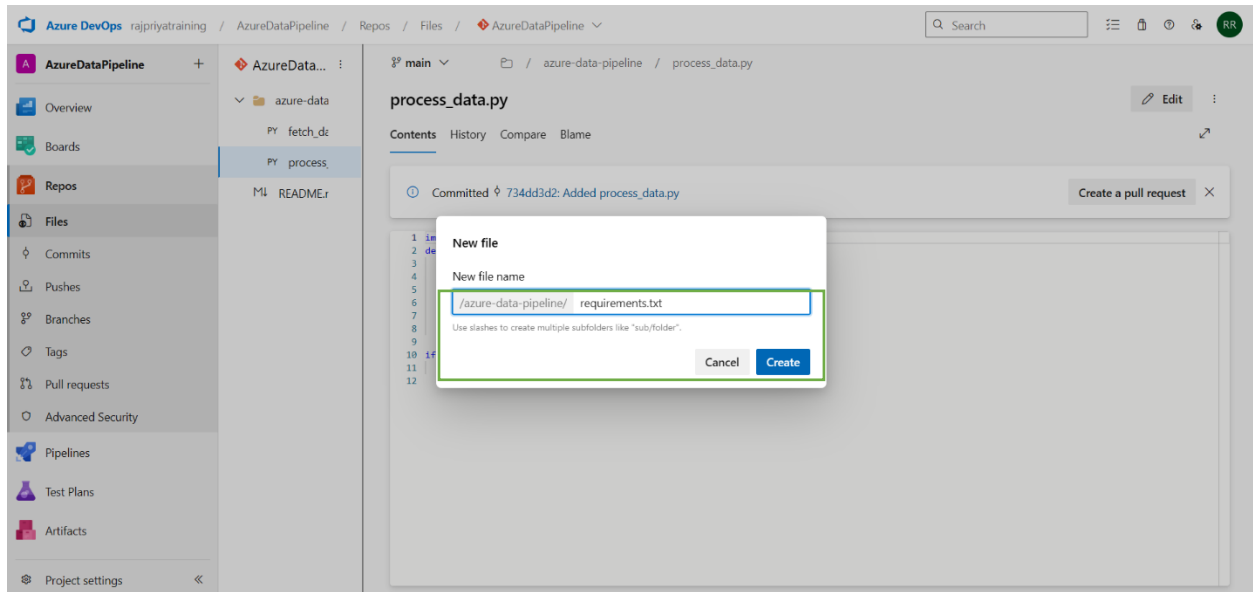
Step 12:

Again, Click Commit.



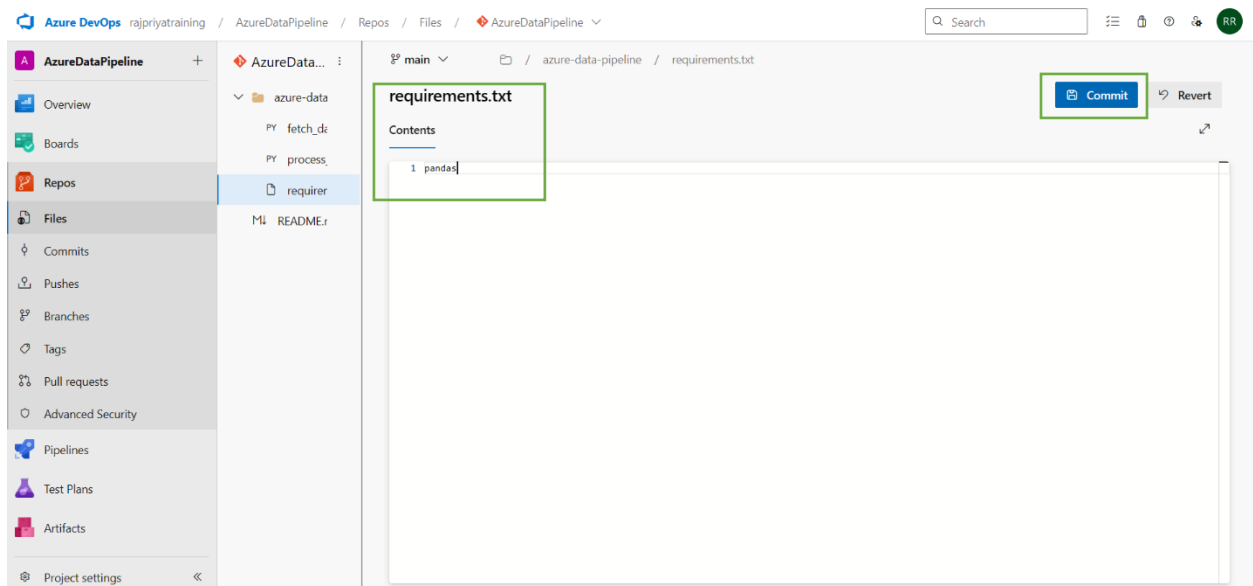
Step 13:

Create a “requirements.txt” file -> Create.



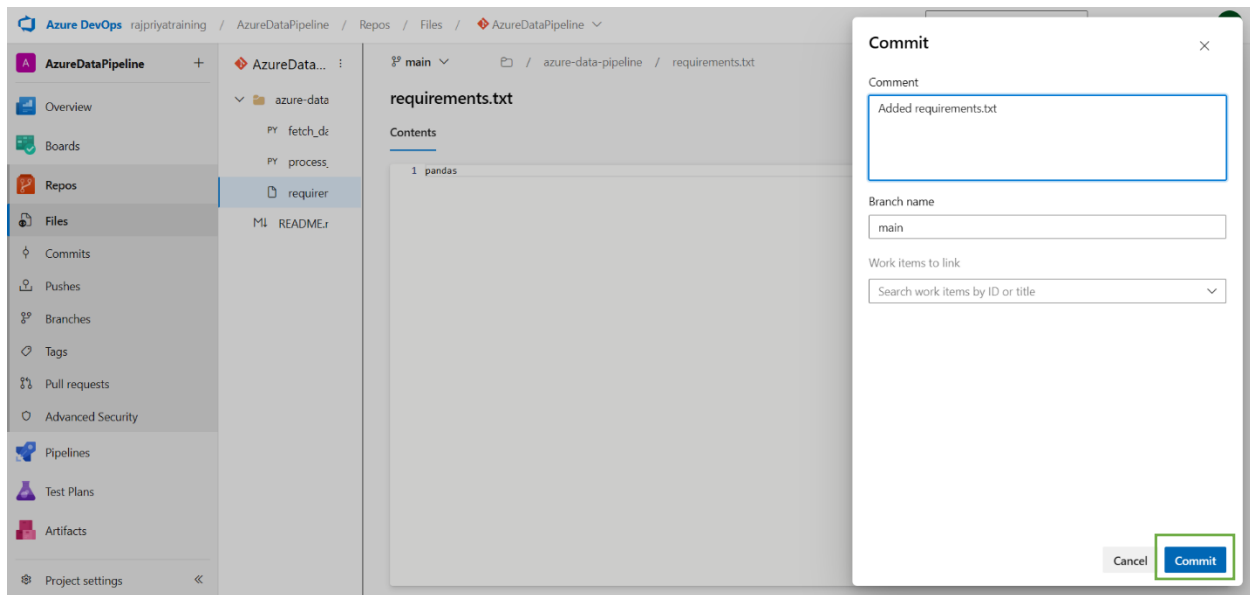
Step 14:

Click Commit.



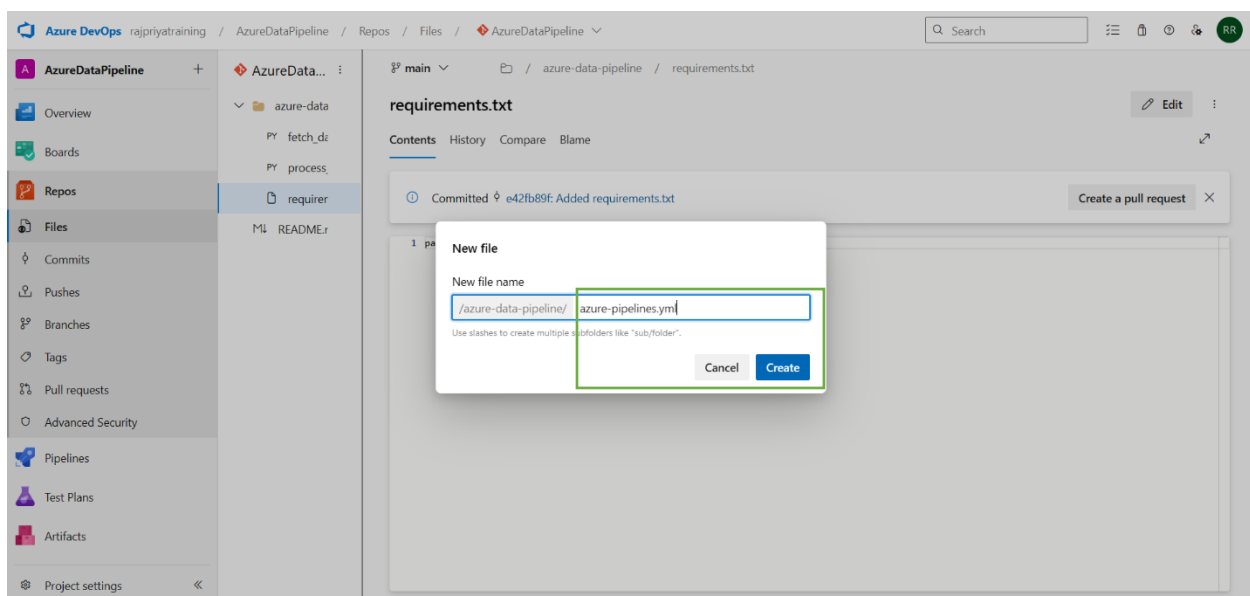
Step 15:

Again, Click Commit.



Step 16:

Create a “azure-pipelines.yml” file -> Create.



Step 17:

Write your code -> Commit.

The screenshot shows the Azure DevOps web interface. On the left is a sidebar with navigation options: Overview, Boards, Repos, Files, Commits, Pushes, Branches, Tags, Pull requests, Advanced Security, Pipelines, Test Plans, and Artifacts. The main area displays the 'azure-pipelines.yml' file for the 'azure-data' pipeline. The file content is as follows:

```
1 trigger:
2   - main
3   # Runs pipeline when code is pushed to 'main' branch
4 pool:
5   vmImage: 'ubuntu-latest'
6
7 steps:
8   # Step 1: Checkout code from repo
9   - task: Checkout@2
10
11 # Step 2: Set up Python
12 - task: UsePythonVersion@1
13   inputs:
14     versionSpec: '3.10'
15     addtoPath: true
16
17 # Step 3: Install dependencies
18 - script: |
19   python -m pip install --upgrade pip
20   pip install -r data_pipeline/requirements.txt
21   displayName: 'Install dependencies'
22
23 # Step 4: Fetch raw data
24 - script: |
25   cd data_pipeline
26   python fetch_data.py
27   displayName: 'Fetch raw data'
28
29 # Step 5: Process data
30 - script: |
31   cd data_pipeline
32   python process_data.py
33   displayName: 'Process data'
34
35 # Step 6: Publish output artifact
36 - task: PublishBuildArtifacts@1
37   inputs:
38     PathtoPublish: 'data_pipeline/processed_data.json'
39     ArtifactName: 'ProcessedData'
40     publishLocation: 'Container'
```

At the top right of the file editor, there is a 'Commit' button highlighted with a green box, and a 'Revert' button next to it.

Step 18:

Again, click on Commit.

The screenshot shows the 'Commit' dialog box in Azure DevOps. The dialog has a 'Comment' field with the text 'Added azure-pipelines.yml'. Below the comment field is a 'Branch name' field with 'main' entered. At the bottom, there is a 'Work items to link' section with a dropdown menu labeled 'Search work items by ID or title'. At the bottom right of the dialog, there are 'Cancel' and 'Commit' buttons, with the 'Commit' button highlighted by a green box.

Step 19:

Go to Pipelines -> Create Pipeline.

The screenshot shows the Azure DevOps interface for a project named 'AzureDataPipeline'. The left sidebar contains a navigation menu with options: Overview, Boards, Repos, Pipelines (highlighted with a green box), Pipelines (with a sub-menu icon), Environments, Library, Test Plans, and Artifacts. The main content area features an illustration of a robot and a person, followed by the heading 'Create your first Pipeline' and the text 'Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.' Below this is a blue 'Create Pipeline' button, which is also highlighted with a green box. At the bottom of the page, a URL is visible: https://dev.azure.com/rajpriyatraining/AzureDataPipeline/_apps/hub/ms-vs-ci-workflow/build-ci-hub?_a=build-definition-getting-started.

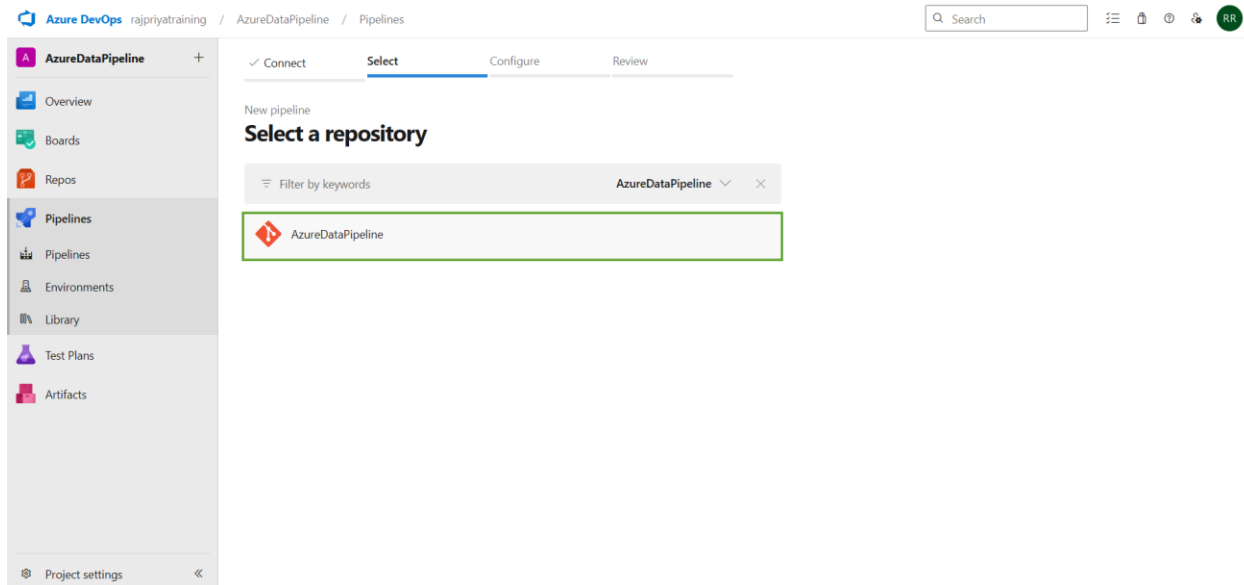
Step 20:

Select Azure Repos Git.

The screenshot shows the 'Where is your code?' selection screen in the Azure DevOps Pipelines interface. The left sidebar is the same as in Step 19. The main content area has a progress bar with four steps: Connect (active), Select, Configure, and Review. Below the progress bar, the heading 'Where is your code?' is displayed. Three options are listed: 'Azure Repos Git' (highlighted with a green box), 'GitHub', and 'Bitbucket Cloud'. Each option includes a brief description and a 'YAML' icon. A 'More options' button is located to the right of the GitHub option. The bottom of the sidebar shows a 'Project settings' link.

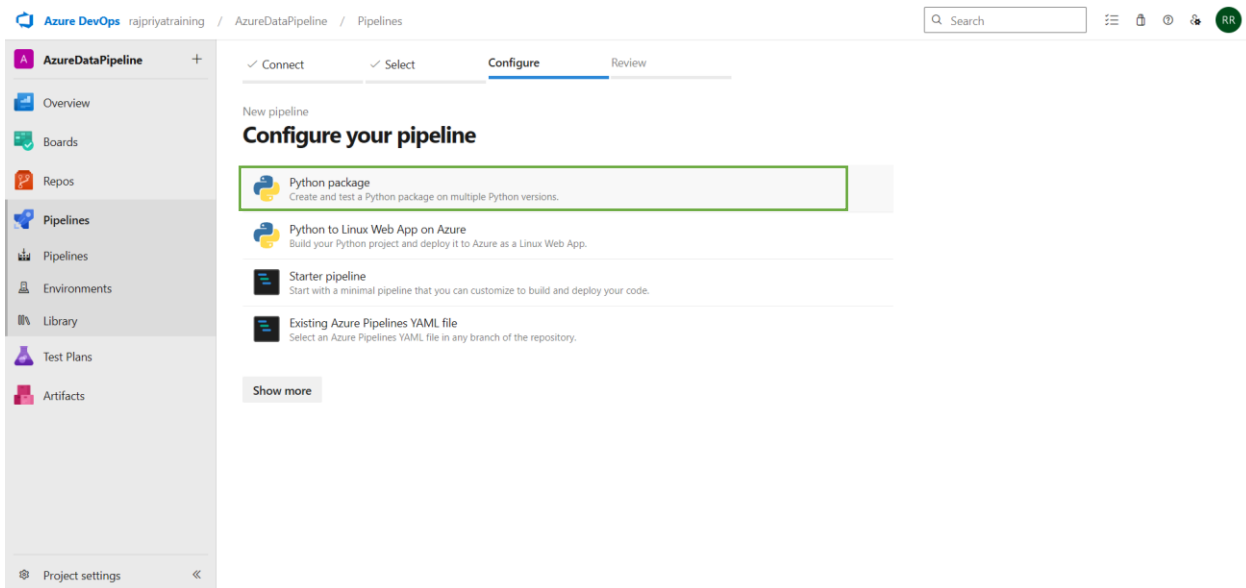
Step 21:

Select AzureDataPipeline.



Step 22:

Select Python Package.



Step 23:

Configure your yaml file and Click on Save and Run.

The screenshot shows the Azure DevOps interface for a new pipeline. The left sidebar contains navigation links: Overview, Boards, Repos, Pipelines (selected), Environments, Library, Test Plans, and Artifacts. The main area is titled 'Review your pipeline YAML' and shows the 'Review' tab of a new pipeline configuration. The pipeline is named 'AzureDataPipeline / azure-pipelines.yml'. The YAML content is displayed in a code editor, showing a trigger for pushes to the 'main' branch, followed by six steps: 1. Check out code from repo, 2. Set up Python, 3. Install dependencies, 4. Fetch raw data, 5. Process data, and 6. Publish output artifact. The 'Save and run' button is highlighted with a green box in the top right corner.

Step 24:

Again, Click on Save and Run.

This screenshot shows the same 'Review your pipeline YAML' interface as Step 23, but with the 'Save and run' dialog box open on the right. The dialog box contains the following information: 'Save and run' title, a message 'Saving will commit azure-pipelines.yml to the repository.', a 'Commit message' field with the text 'Set up CI with Azure Pipelines', an 'Optional extended description' field with the placeholder 'Add an optional description...', and two radio button options: 'Commit directly to the main branch' (selected) and 'Create a new branch for this commit'. The 'Save and run' button at the bottom right of the dialog is highlighted with a green box.

Step 25:

Now we can see the summary of the pipeline and it is scheduled to run with the configured agent.

The screenshot displays the Azure DevOps web interface. The top navigation bar shows the project path: **Azure DevOps** / rajpriyatraining / AzureDataPipeline / Pipelines / AzureDataPipeline / 20250825.1. A search bar and user profile icon are on the right. The left sidebar contains a navigation menu with options: Overview, Boards, Repos, Pipelines (selected), Environments, Library, Test Plans, and Artifacts. The main content area is titled **#20250825.1 • Set up CI with Azure Pipelines** and includes a **Cancel** button. Below the title, there are tabs for **Summary** and **Code Coverage**. The **Summary** tab is active, showing details for an individual CI run by user **Rajpriya Vondivillu Raja**. A **View 6 changes** button is present. The summary includes a table for **Repository and version** (AzureDataPipeline, main branch, commit cfc65748), **Time started and elapsed** (Just now), **Related** (0 work items, 0 artifacts), and **Tests and coverage** (Get started). Below this, a **Jobs** table shows a single job named **Job** with a status of **Queued**.

Azure DevOps rajpriyatraining / AzureDataPipeline / Pipelines / AzureDataPipeline / 20250825.1

AzureDataPipeline +

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

#20250825.1 • Set up CI with Azure Pipelines AzureDataPipeline Cancel

Summary Code Coverage

Individual CI by **Rajpriya Vondivillu Raja** View 6 changes

Repository and version
AzureDataPipeline
main cfc65748

Time started and elapsed
Just now

Related
0 work items
0 artifacts

Tests and coverage
Get started

Jobs

Name	Status	Duration
Job	Queued	