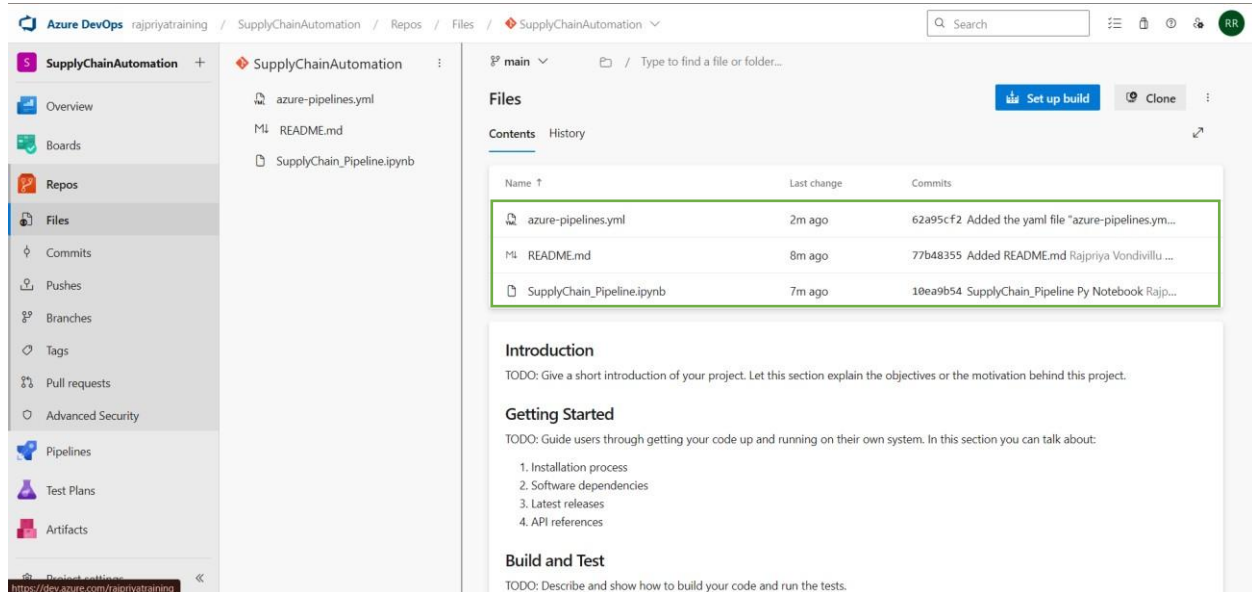


Pictorial Document for DevOps Pipeline Process – WEEK_5

Step 1:

Create New Project -> SupplyChainAutomation -> Click on to the Three dots on the top right corner and add your .ipynb notebook -> “SupplyChain_Pipeline.ipynb” and yaml file “azure-pipelines.yml” file.



The screenshot shows the Azure DevOps interface for a project named 'SupplyChainAutomation'. The 'Files' tab is selected, displaying a list of files: 'azure-pipelines.yml' (2m ago), 'README.md' (8m ago), and 'SupplyChain_Pipeline.ipynb' (7m ago). Below the file list, the 'Introduction' section is visible, followed by 'Getting Started' and 'Build and Test' sections.

Name	Last change	Commits
azure-pipelines.yml	2m ago	62a95cf2 Added the yaml file "azure-pipelines.yml..."
README.md	8m ago	77b48355 Added README.md Rajpriya Vondivilu...
SupplyChain_Pipeline.ipynb	7m ago	10ea9b54 SupplyChain_Pipeline Py Notebook Rajp...

Introduction
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

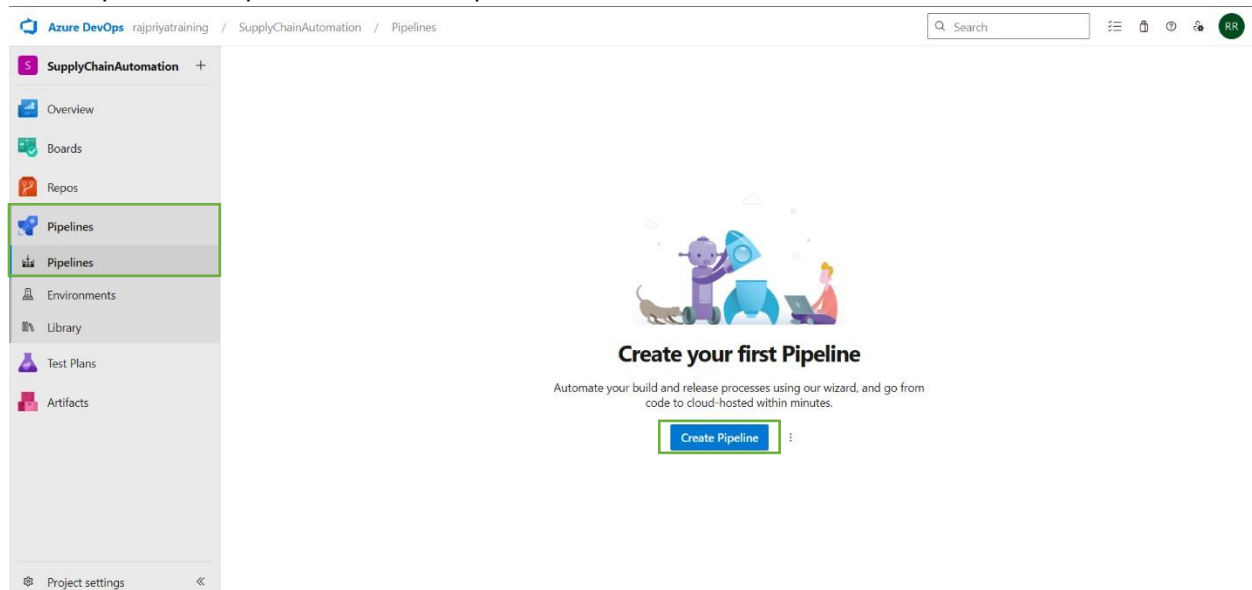
Getting Started
TODO: Guide users through getting your code up and running on their own system. In this section you can talk about:

1. Installation process
2. Software dependencies
3. Latest releases
4. API references

Build and Test
TODO: Describe and show how to build your code and run the tests.

Step 2:

Go to Pipelines -> Pipelines -> Create Pipeline.



The screenshot shows the Azure DevOps interface for the 'SupplyChainAutomation' project, specifically the 'Pipelines' tab. The 'Create Pipeline' button is highlighted. The main content area displays a 'Create your first Pipeline' wizard with an illustration of a person and a robot.

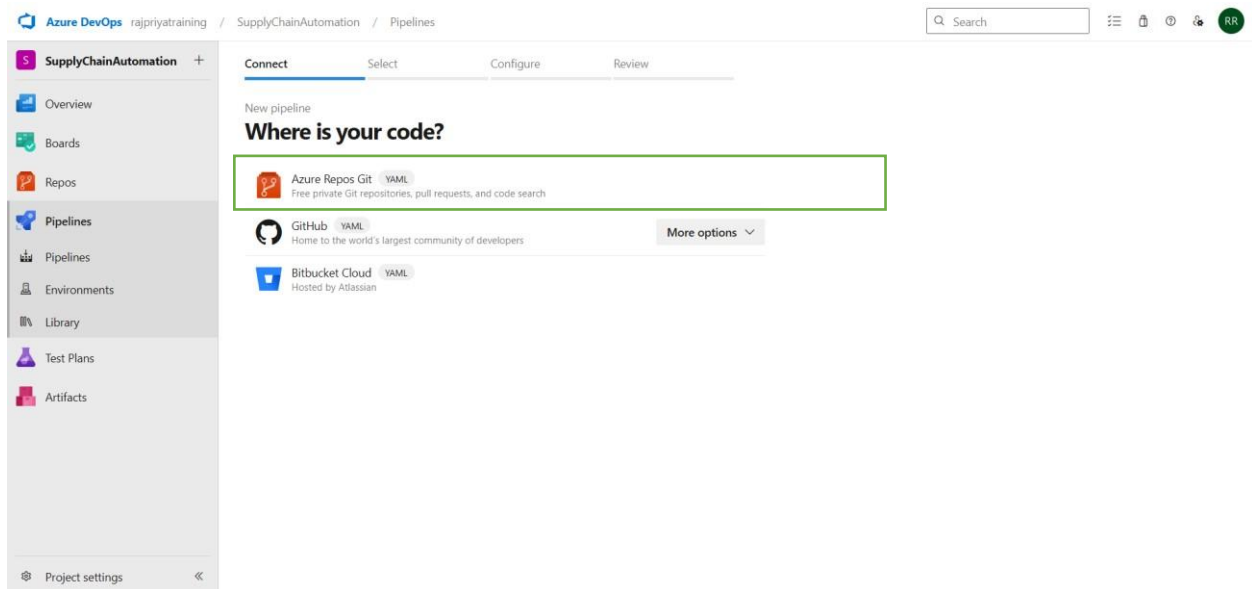
Create your first Pipeline

Automate your build and release processes using our wizard, and go from code to cloud-hosted within minutes.

[Create Pipeline](#)

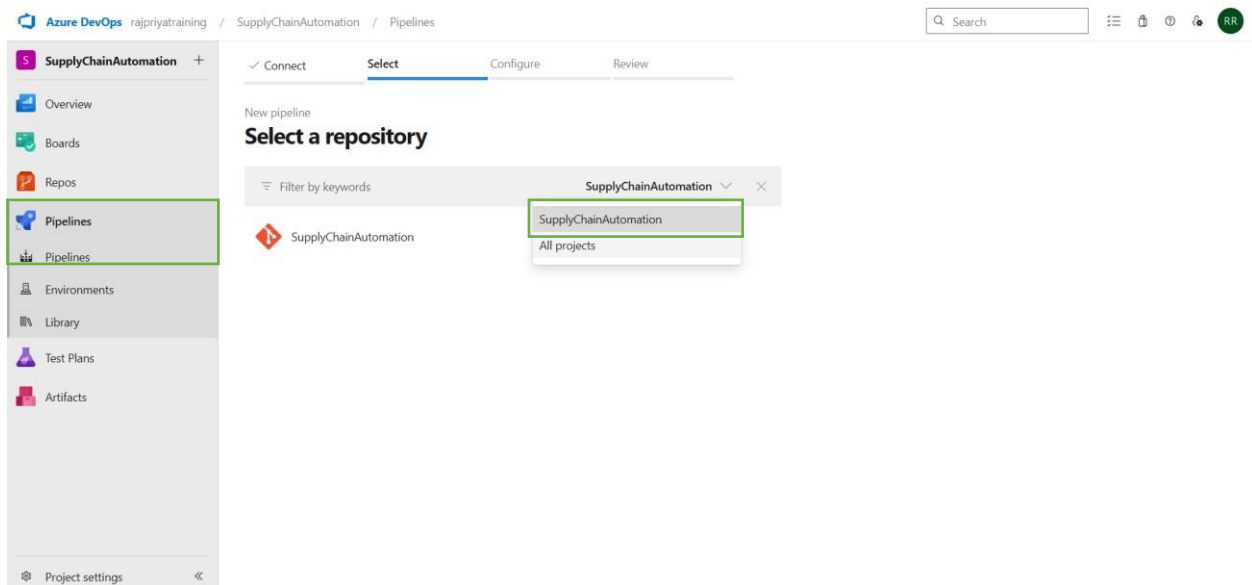
Step 3:

Select Azure Repos Git.



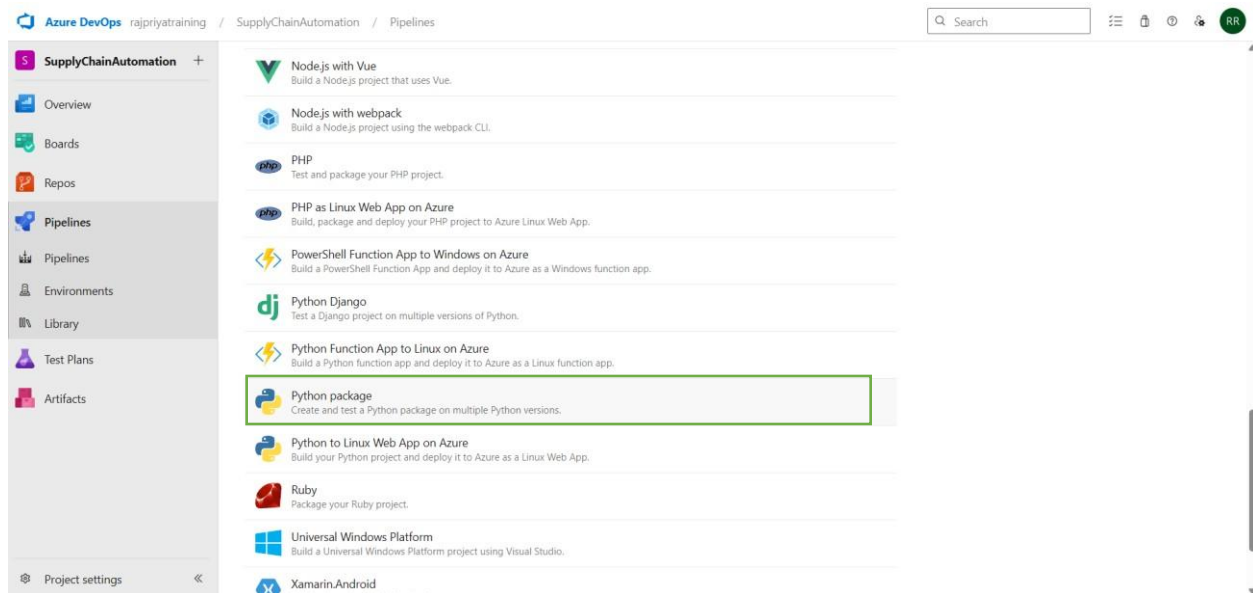
Step 4:

Select a repository -> SupplyChainAutomation.



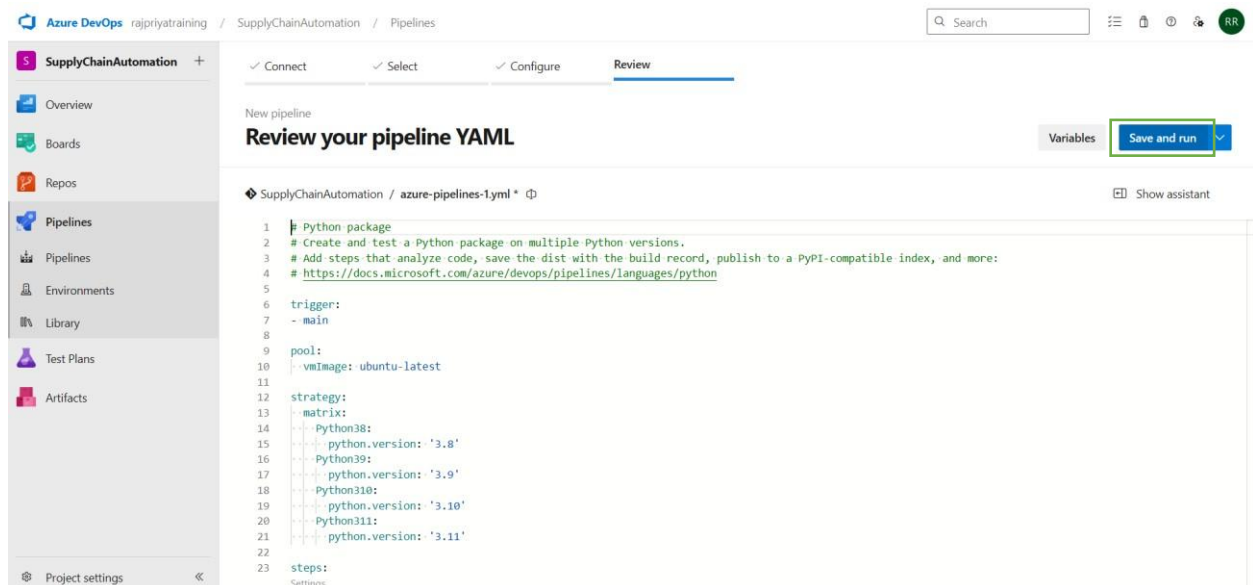
Step 5:

Select Python Package.



Step 6:

Click on Save and Run.



Step 7:

Again, click on Save and Run.

The screenshot shows the 'Review your pipeline YAML' dialog in Azure DevOps. The pipeline is named 'New pipeline' and is located in the 'SupplyChainAutomation' repository. The YAML content is as follows:

```
1 # Python package
2 # Create and test a Python package on multiple Python versions.
3 # Add steps that analyze code, save the dist with the build record, publish to a PyPI-comp
4 # https://docs.microsoft.com/azure/devops/pipelines/languages/python
5
6 trigger:
7   - main
8
9 pool:
10  vmImage: ubuntu-latest
11
12 strategy:
13   matrix:
14     Python38:
15       python.version: '3.8'
16     Python39:
17       python.version: '3.9'
18     Python310:
19       python.version: '3.10'
20     Python311:
21       python.version: '3.11'
22
23 steps:
24   Settings
```

The 'Save and run' modal is open, showing the commit message 'Set up CI with Azure Pipelines' and the option to 'Commit directly to the main branch' (selected). The 'Save and run' button is highlighted with a green border.

Step 8:

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

The screenshot shows the pipeline summary page for the pipeline named '#20250822.2 • Set up CI with Azure Pipelines'. The pipeline is manually run by 'Rajpriya Vondivillu Raja'. The summary shows the repository and version, time started and elapsed, related work items, and tests and coverage. The jobs table shows one job named 'Job' with a status of 'Queued'.

Name	Status	Duration
Job	Queued	