**WEEK 5 :** **CI/CD Automation with Azure DevOps**

**Pre-requisites:**

Before beginning, ensure:

* You have **SSH Public Key** or **Personal Access Token (PAT)** configured.
* Tools Installed:
  + **Git**
  + **Python 3.x**
  + **VS Code / any IDE**
* Your local project folder contains:
  + Python script for **ELT and alerting**
  + Azure DevOps **YAML pipeline file**

**Step 1: Create Python and YAML Files in Local Project**

1. Create a Python file (e.g. elt\_&\_alert.py)  
   Example:

# elt\_energy\_alert.py

import pandas as pd

def load\_data():

# Simulate device energy usage data

data = {

'device\_id': ['AC01', 'WM02', 'FR03'],

'date': ['2025-07-01'] \* 3,

'kwh\_used': [8.5, 12.4, 9.3]

}

return pd.DataFrame(data)

def transform\_data(df):

return df[df['kwh\_used'] > 10]

def alert\_high\_usage(df):

for \_, row in df.iterrows():

print(f"ALERT: {row['device\_id']} used {row['kwh\_used']} kWh on {row['date']}!")

def run\_elt():

df = load\_data()

high\_usage = transform\_data(df)

alert\_high\_usage(high\_usage)

if \_\_name\_\_ == "\_\_main\_\_":

run\_elt()

Create Azure Pipeline YAML (azure-pipelines.yml)

trigger:

- main

pool:

vmImage: 'ubuntu-latest'

steps:

- task: UsePythonVersion@0

inputs:

versionSpec: '3.x'

- script: |

pip install pandas

python Smart\_Home\_Energy\_Usage\_Tracker/

displayName: 'Successfully Run'

- script: |

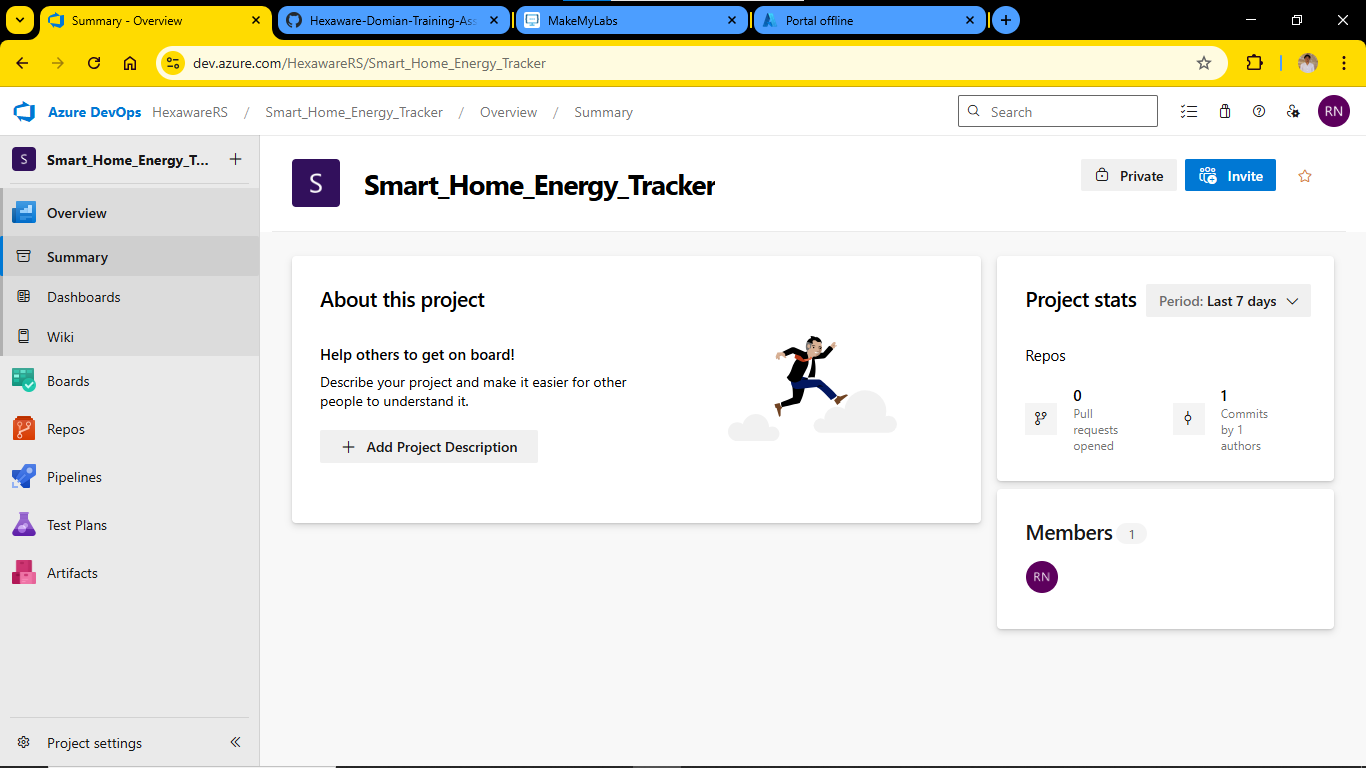
pip install pandas

python elt\_energy\_alert.py

displayName: 'Run ELT + Alert Script'

**Step 2: Create a New Azure DevOps Project**

1. Go to [Azure DevOps Portal](https://dev.azure.com).
2. Click **New Project** → Provide name and visibility → Click **Create**.
3. Navigate to **Repos** → Click **Clone** → Copy the **SSH** URL.



**Step 3: Push Local Project to Azure Repo via SSH**

Open **Command Prompt / Git Bash**, then run the following:

# Go to the directory where your local project exists

cd path\to\your\project-folder

# Initialize git repository

git init

# Add files to git

git add .

# Commit the files

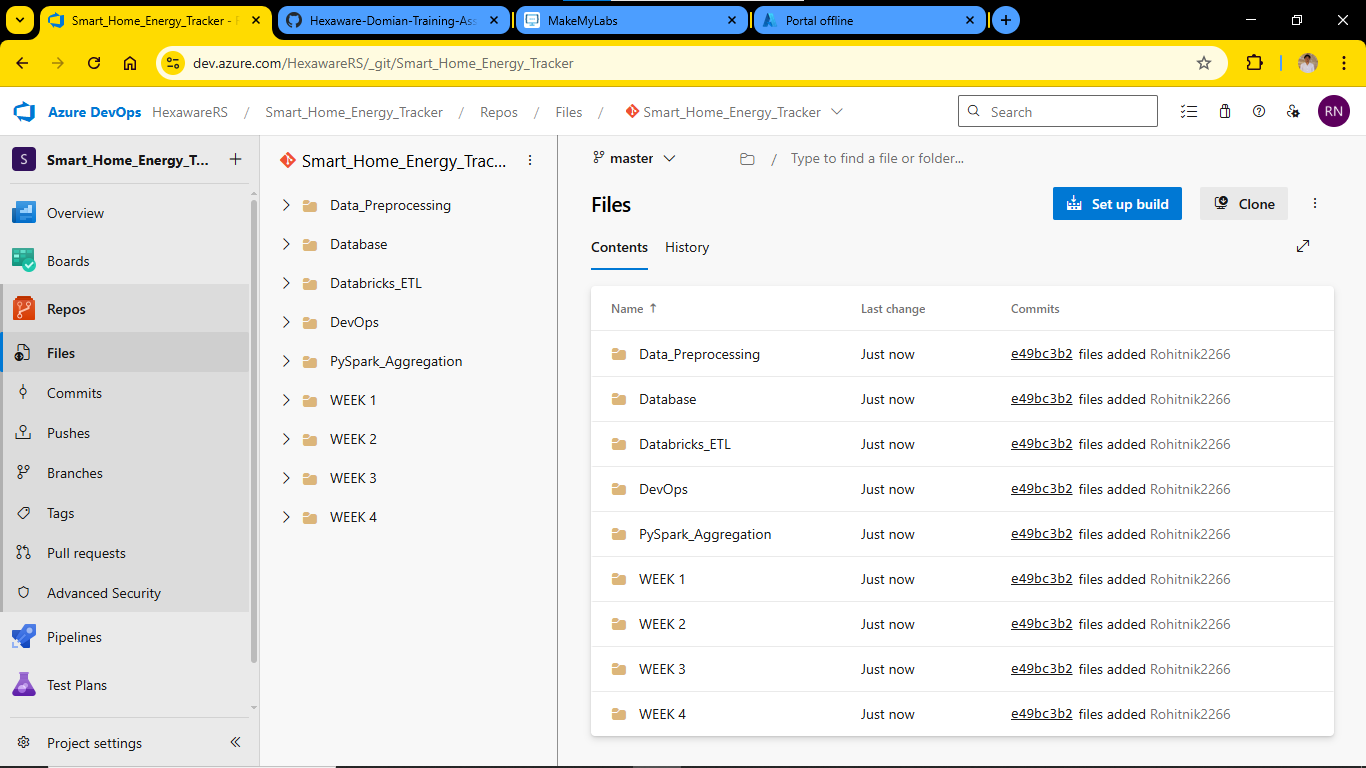
git commit -m "Initial commit"

# Add Azure DevOps repo as remote (replace with your SSH link)

git remote add origin git@ssh.dev.azure.com:v3/YourOrg/YourProject/YourRepo

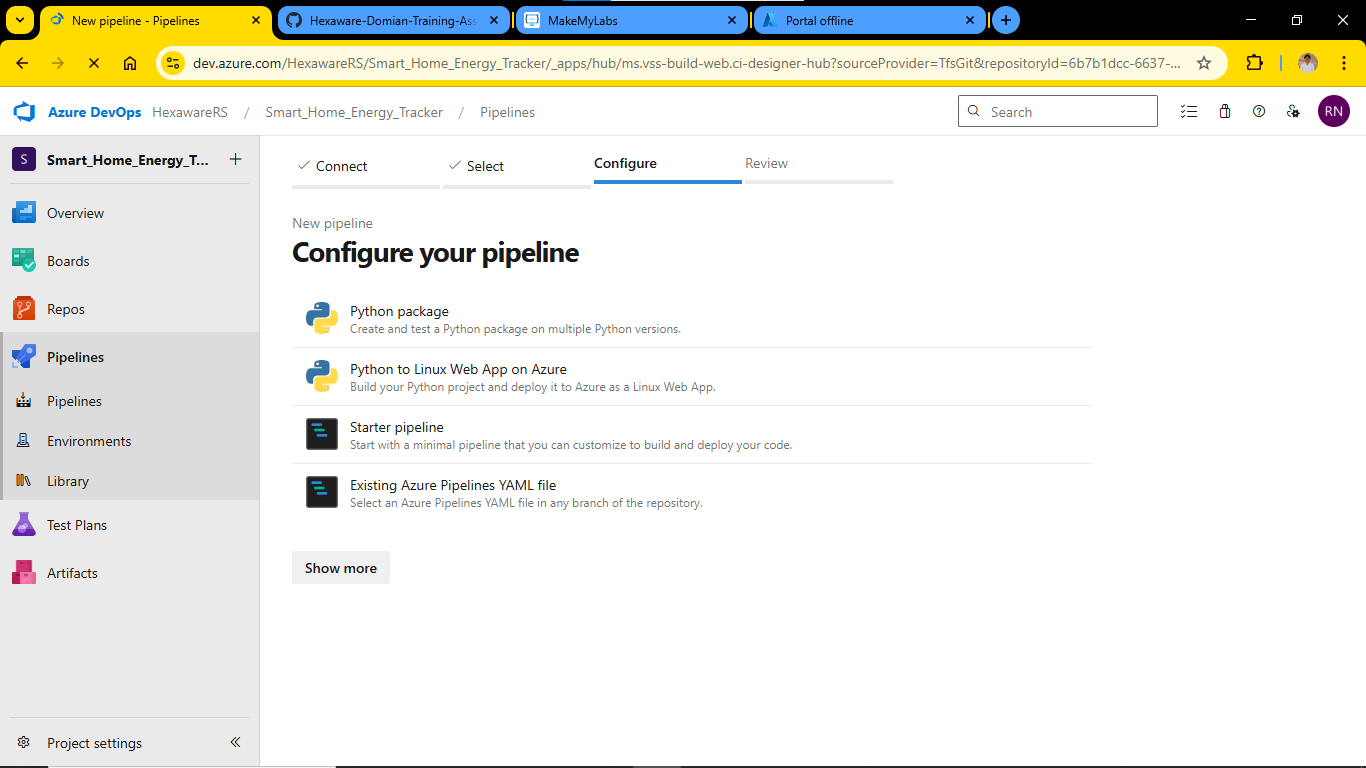
# Push the code to Azure repo

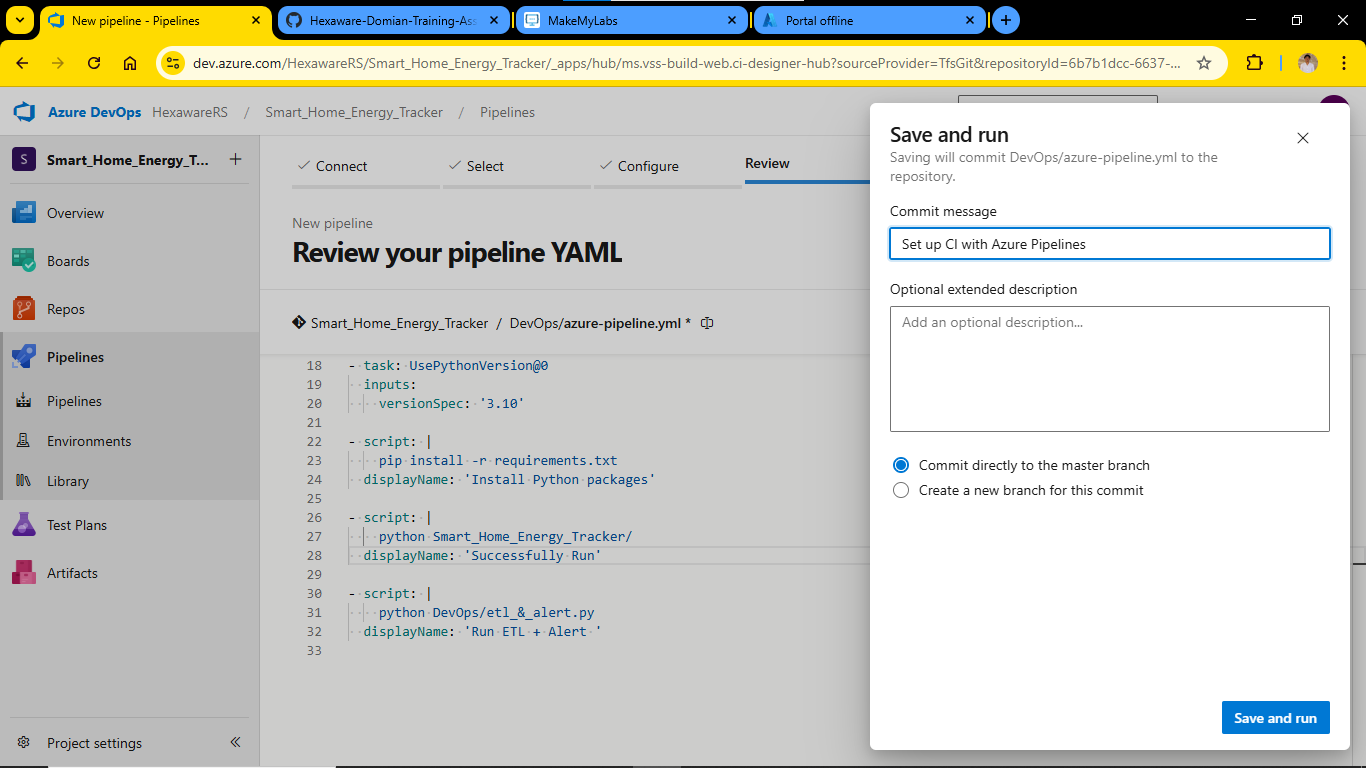
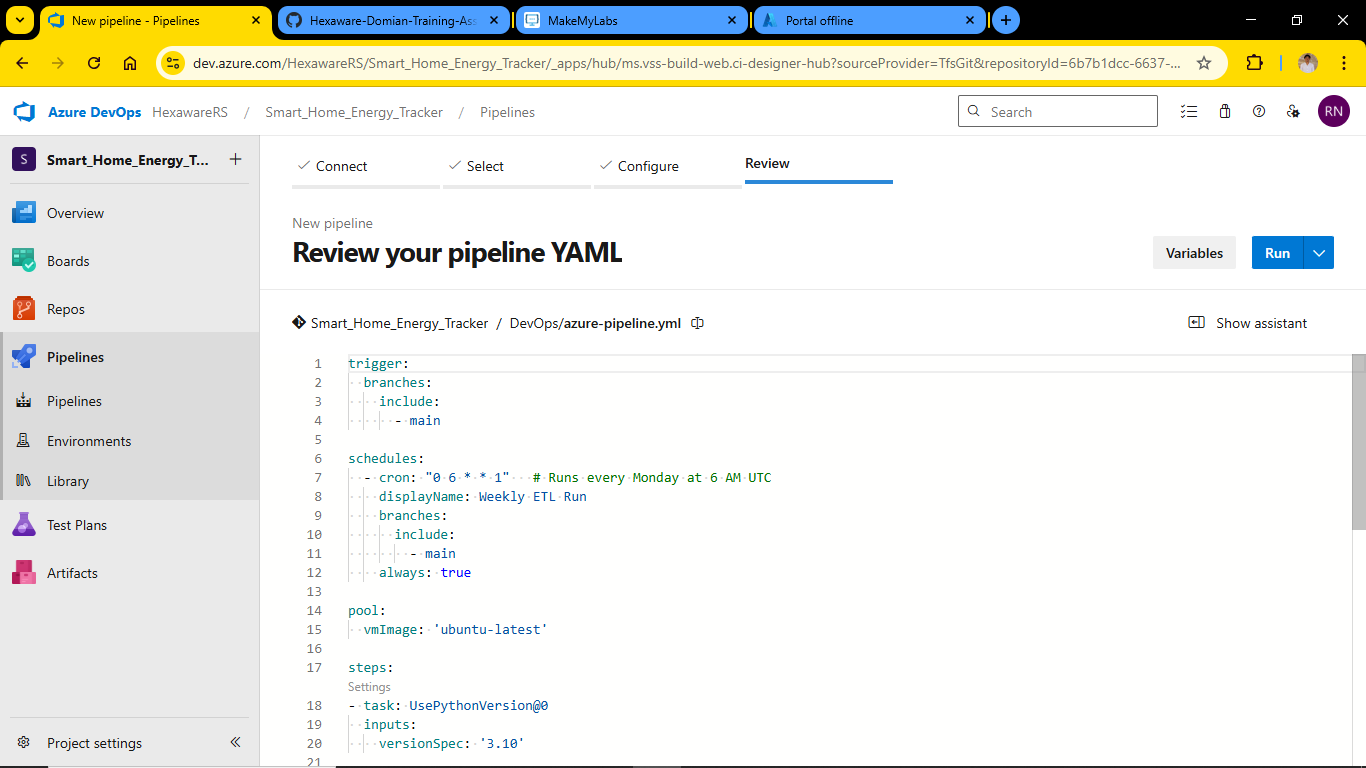
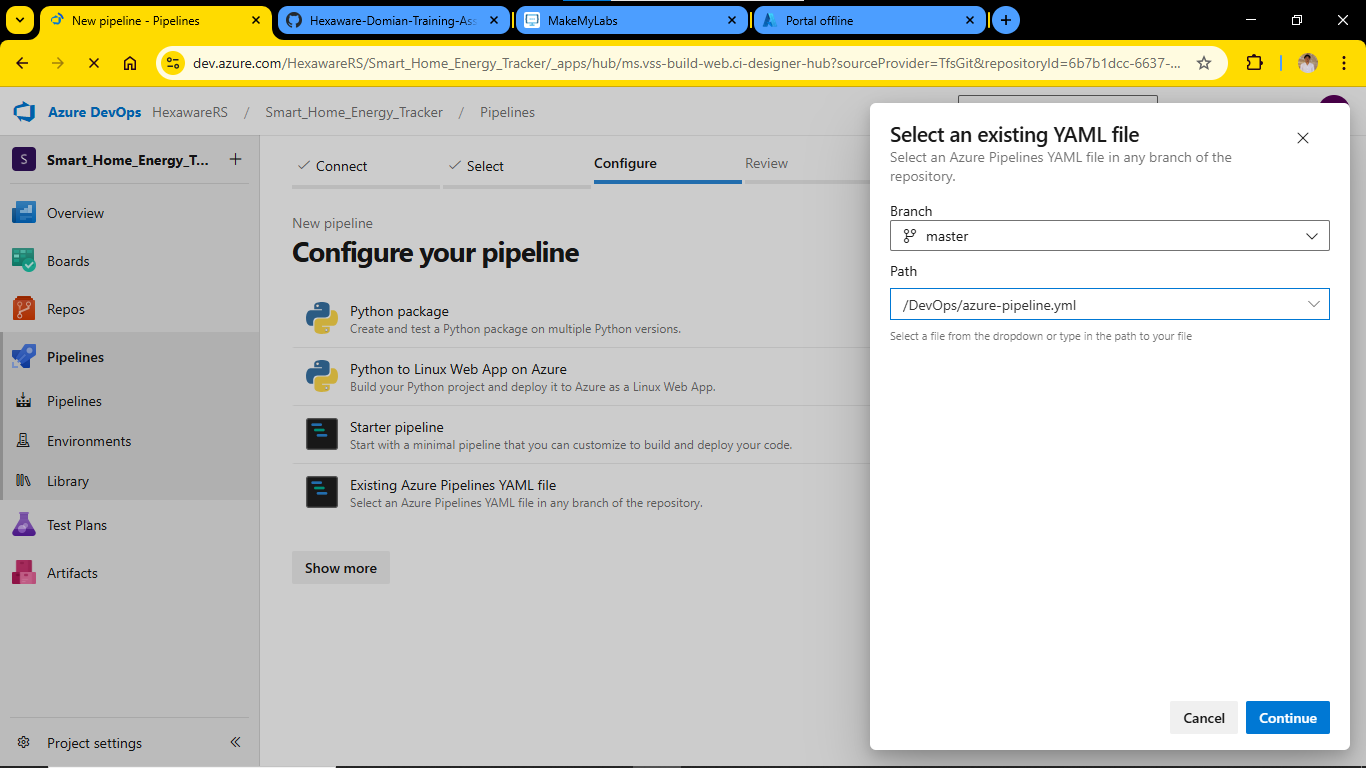
git push -u origin main



**Step 4: Configure and Run the Azure Pipeline**

1. Go back to your Azure DevOps project.
2. Navigate to **Pipelines** → Click **Create Pipeline**.
3. Choose:
   * **Code in Azure Repos Git**
   * Select your repository
   * Choose **"Existing Azure Pipelines YAML file"**
4. Select:
   * **Branch:** main
   * **Path:** /DevOps/azure-pipelines.yml
5. Click **Continue**, then **Run** the pipeline.





**✅ Final Output**

* The pipeline will run automatically.
* It will set up the environment, run your all file and alert.py, and display output in logs.
* If everything is set correctly, you'll see **“Successfully Run” and “Run ETL + Alert”** message.

