

# Week 5: CI/CD Automation with Azure DevOps

## Objective

1. To automate monthly personal expense analysis using Azure DevOps.
2. Load expense data from CSV.
3. Calculate total monthly spend and category-wise breakdown.
4. Trigger an alert if spending exceeds ₹10,000.
5. Automate the entire flow using Azure DevOps pipeline.

## Pre-requisites

Before you begin, ensure you have:

- **Azure DevOps Account** and access to create projects
- **SSH Key** or **Personal Access Token (PAT)** configured
- Tools Installed:
  - Git
  - Python 3.x
  - VS Code or any IDE

## Your Local Project Folder Must Contain:

- `week1/expenses.csv` → The input data
- `week2/alert.py` → Python script
- `azure-pipelines.yml` → Azure DevOps pipeline file

# Step-by-Step Execution

## Step 1: Create Python Script & YAML File Locally

alert.py

```
import pandas as pd
import os

df = pd.read_csv('week1/expenses.csv')
df['amount'] = df['amount'].replace('[\$,]', '',
regex=True).astype(float)
df['date'] = pd.to_datetime(df['date'])
df['month'] = df['date'].dt.to_period('M')

monthly_summary = df.groupby(['month',
'category'])['amount'].sum().unstack().fillna(0)

os.makedirs('output', exist_ok=True)
monthly_summary.to_csv('output/monthly_summary.csv')

total = df['amount'].sum()
if total > 10000:
    print(f"⚠ Alert: Total monthly spending is ₹{total:.2f}, which
exceeds ₹10,000.")
else:
    print(f"✅ Total monthly spending is ₹{total:.2f}.")
```

azure-pipelines.yml

```
trigger:
  schedule:
    - cron: "0 0 1 * *" # Runs on the 1st of every month
      displayName: Monthly Expense Report
      branches:
        include:
          - main

pool:
```

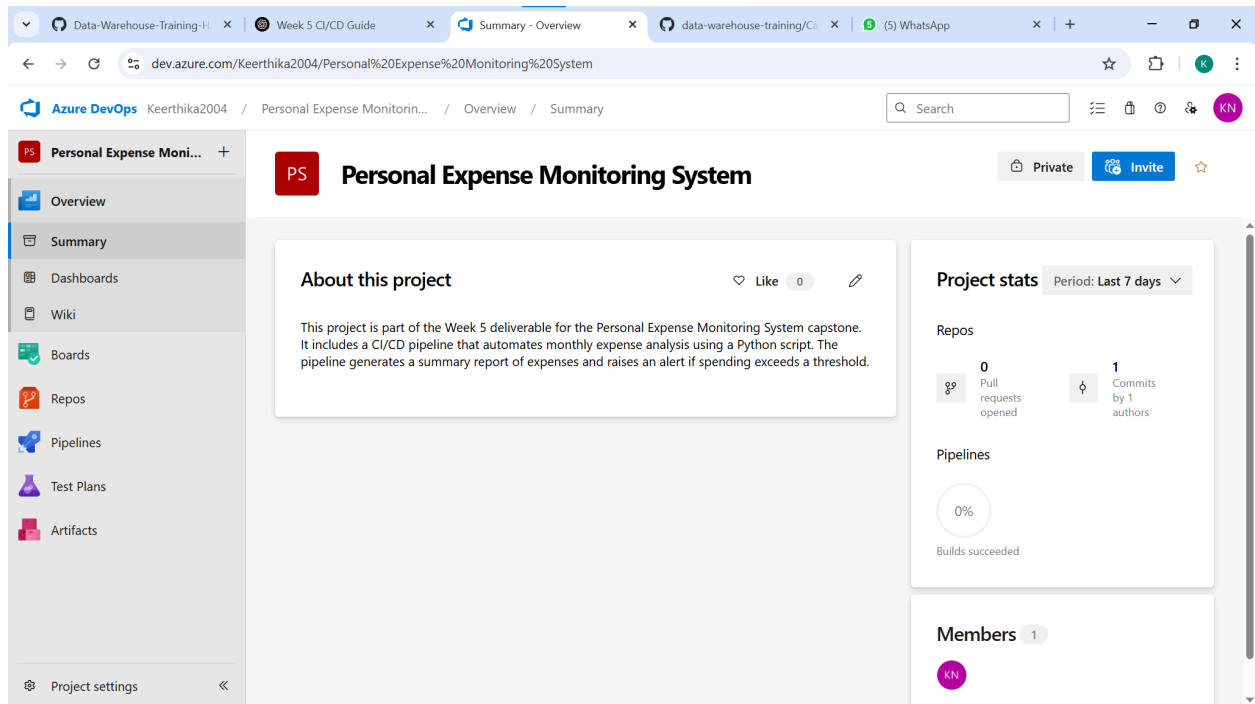
```
vmImage: 'ubuntu-latest'
```

```
steps:
```

- task: UsePythonVersion@0  
 inputs:  
 versionSpec: '3.x'
  
- script: |  
 pip install pandas  
 python week2/alert.py  
 displayName: 'Run Expense Alert Script'
  
- task: PublishBuildArtifacts@1  
 inputs:  
 PathToPublish: 'output/monthly\_summary.csv'  
 ArtifactName: 'MonthlyReport'  
 publishLocation: 'Container'

## Step 2: Create Azure DevOps Project

1. Go to <https://dev.azure.com>
2. Click **New Project**
3. Name: **Personal Expense Monitoring System**
4. Set visibility to **Private** or **Public**
5. Click **Create**



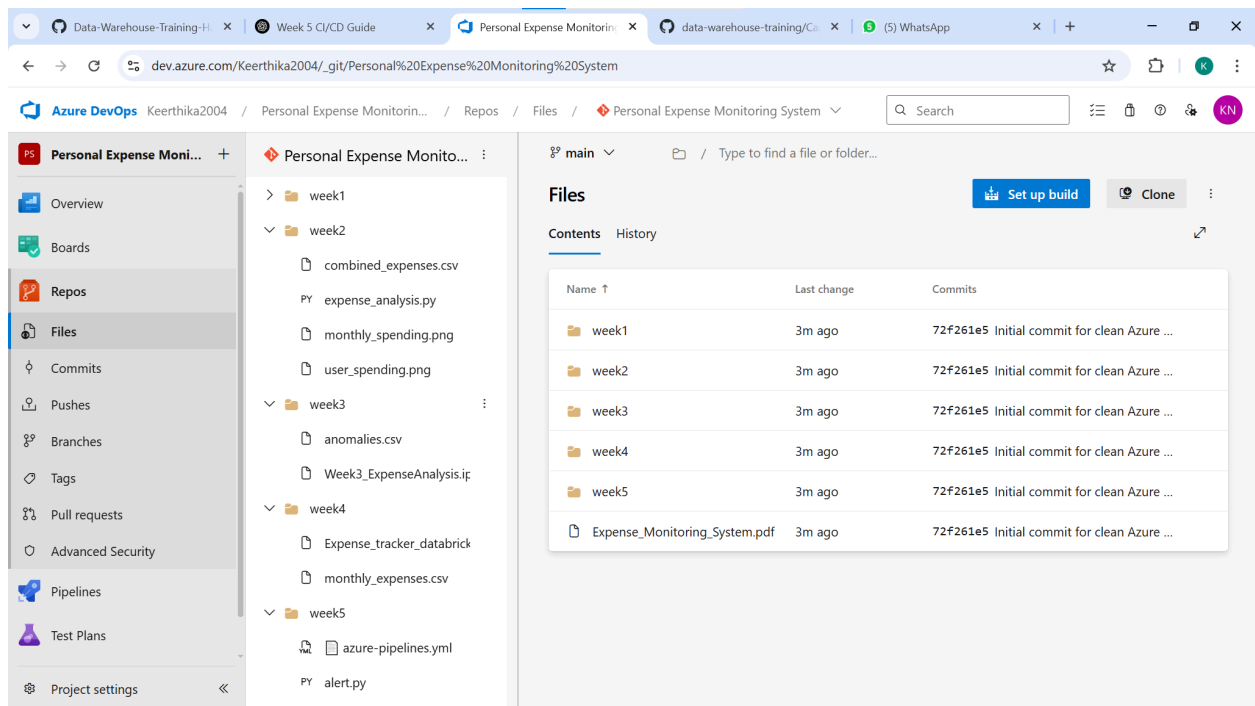
### Step 3: Push Local Code to Azure Repo (SSH)

```
cd "C:\path\to\your\project\Personal Expense Monitoring System"
```

```
git init
git add .
git commit -m "Initial commit for Week 5 CI/CD"
git branch -M main
git remote add origin
git@ssh.dev.azure.com:v3/YOUR_ORG_NAME/YOUR_PROJECT/YOUR_PROJECT
git push -u origin main
```

Replace **YOUR\_ORG\_NAME** and **YOUR\_PROJECT** with your actual values.

```
Windows PowerShell
PS C:\Users\Keerthika> cd "C:\Users\Keerthika\Desktop\Personal Expense Tracking System\data-warehouse-training\Capstone Project\Project 1 - Personal Expense Monitoring System"
>> # Remove any old git setup
>> Remove-Item -Recurse -Force .git
>> # Remove/Initialize git clean
>> git init
>> git add .
>> git commit -m "Initial commit for clean Azure DevOps push"
>> git branch -M main
>>
Initialized empty Git repository in C:\Users\Keerthika\Desktop\Personal Expense Tracking System\data-warehouse-training\Capstone Project\Project 1 - Personal Expense Monitoring System\.git\
[master (root-commit) 72f261e] Initial commit for clean Azure DevOps push
16 files changed, 1254 insertions(+)
create mode 100644 Expense_Monitoring_System.pdf
create mode 100644 week1/categories.csv
create mode 100644 week1/expense_tracker.sql
create mode 100644 week1/expenses.csv
create mode 100644 week1/mongodb.json
create mode 100644 week1/users.csv
create mode 100644 week2/combined_expenses.csv
create mode 100644 week2/expense_analysis.py
create mode 100644 week2/monthly_spending.png
create mode 100644 week2/user_spending.png
create mode 100644 week3/Week3_ExpenseAnalysis.ipynb
create mode 100644 week3/anomalies.csv
create mode 100644 week4/Expense_tracker_databricks.ipynb
create mode 100644 week4/monthly_expenses.csv
create mode 100644 week5/alert.py
create mode 100644 "week5\360\237\223\204 azure-pipelines.yml"
PS C:\Users\Keerthika\Desktop\Personal Expense Tracking System\data-warehouse-training\Capstone Project\Project 1 - Personal Expense Monitoring System> git remote add origin git@ssh.dev.azure.com:v3/Keerthika2004/PersonalExpenseX20MonitoringX20System
PS C:\Users\Keerthika\Desktop\Personal Expense Tracking System\data-warehouse-training\Capstone Project\Project 1 - Personal Expense Monitoring System> git push -u origin main
Enumerating objects: 23, done.
Counting objects: 100% (23/23), done.
Delta compression using up to 4 threads
Compressing objects: 100% (23/23), done.
Writing objects: 100% (23/23), 108.97 KiB | 368.00 KiB/s, done.
Total 23 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
remote: Analyzing objects... (23/23) (390 ms)
remote: Validating commits... (1/1) done (1 ms)
remote: Storing packfile... done (95 ms)
remote: Storing index... done (44 ms)
remote: Updating refs... done (201 ms)
To ssh.dev.azure.com:v3/Keerthika2004/PersonalExpenseX20MonitoringX20System/PersonalExpenseX20MonitoringX20System
 * [new branch]      main -> main
branch 'main' set up to track 'origin/main'.
PS C:\Users\Keerthika\Desktop\Personal Expense Tracking System\data-warehouse-training\Capstone Project\Project 1 - Personal Expense Monitoring System>
```



## Step 4: Configure and Run the Pipeline

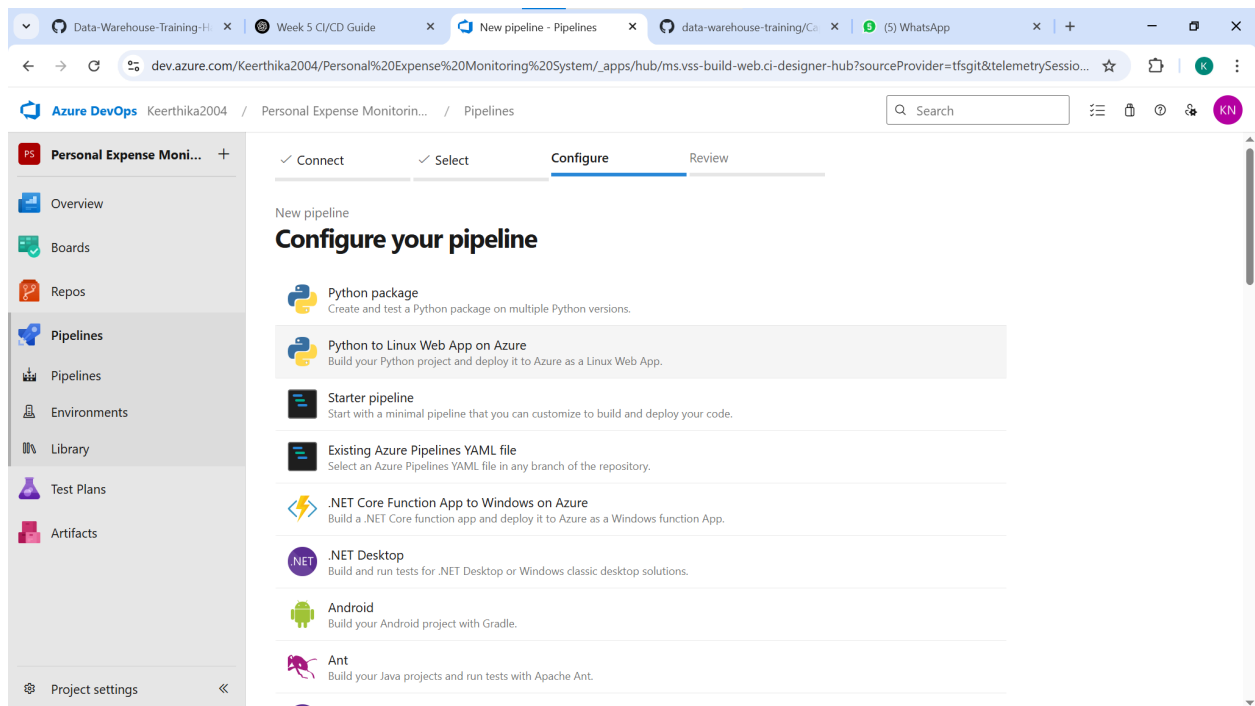
1. Go to **Pipelines** → **New Pipeline**
2. Choose:

- Source: **Azure Repos Git**
- Repository: Your project repo
- YAML file: *Existing Azure Pipelines YAML*

### 3. Specify:

- Branch: **main**
- Path: **/azure-pipelines.yml**

### 4. Click **Continue** → **Run**



dev.azure.com/Keerthika2004/Personal%20Expense%20Monitoring%20System/\_apps/hub/ms.vss-build-web.ci-designer-hub?sourceProvider=tfsgit&telemetrySessio...

Azure DevOps Keerthika2004 / Personal Expense Monitorin... / Pipelines

Personal Expense Moni... +

Overview  
Boards  
Repos  
Pipelines  
Environments  
Library  
Test Plans  
Artifacts  
Project settings

Connect Select **Configure** Review

### Configure your pipeline

New pipeline

- Python package  
Create and test a Python package on multiple Python versions.
- Python to Linux Web App on Azure  
Build your Python project and deploy it to Azure as a Linux Web App.
- Starter pipeline  
Start with a minimal pipeline that you can customize to build and deploy your code.
- Existing Azure Pipelines YAML file  
Select an Azure Pipelines YAML file in any branch of the repository.
- .NET Core Function App to Windows on Azure  
Build a .NET Core function app and deploy it to Azure as a Windows function App.
- .NET Desktop  
Build and run tests for .NET Desktop or Windows classic desktop solutions.
- Android  
Build your Android project with Gradle.
- Ant  
Build your Java projects and run tests with Apache Ant.

**Select an existing YAML file**

Select an Azure Pipelines YAML file in any branch of the repository.

Branch  
main

Path  
/week5/azure-pipelines.yml

Select a file from the dropdown or type in the path to your file

[Personal Expense Monitoring System](#)

Cancel Continue

dev.azure.com/Keerthika2004/Personal%20Expense%20Monitoring%20System/\_apps/hub/ms.vss-build-web.ci-designer-hub?sourceProvider=tfsgit&telemetrySessio...

Azure DevOps Keerthika2004 / Personal Expense Monitorin... / Pipelines

Personal Expense Moni... +

Overview  
Boards  
Repos  
Pipelines  
Environments  
Library  
Test Plans  
Artifacts  
Project settings

Connect Select **Configure** **Review**

### Review your pipeline YAML

Variables Run

Personal Expense Monitoring System / week5/azure-pipelines.yml

Show assistant

```
1 trigger:
2   schedule:
3     - cron: "0 0 1 * * *" # runs 1st of every month
4     displayName: Monthly Run
5     branches:
6       include:
7         - main
8
9 pool:
10  vmImage: 'ubuntu-latest'
11
12 steps:
13   - task: UsePythonVersion@0
14     inputs:
15       versionSpec: '3.x'
16
17   - script: |
18     pip install pandas
19     python week2/alert.py
20     displayName: 'Run Expense Script'
```

The screenshot shows the Azure DevOps Pipelines interface. The top navigation bar includes the Azure DevOps logo and the project name 'Keerthika2004'. The left sidebar contains a list of project items: Overview, Boards, Repos, Pipelines (selected), Environments, Library, Test Plans, and Artifacts. The main content area displays the details of a failed pipeline run. The title is '#1 • Initial commit for clean Azure DevOps push'. Below the title, there is a message: 'This run will be cleaned up after 1 month based on your project settings.' The 'Summary' tab is selected, showing the build was manually run by 'Keerthika Nagarajan'. The repository is 'Personal Expense Monitoring System' and the version is 'main' with commit '72f261e5'. The build started 'Just now' and took less than 1 second. There are 0 work items and 0 artifacts. The 'Errors' section shows one error: 'Unexpected value 'schedule'' on Line 2, Col: 3 of the file 'azure-pipelines.yml'. A 'View documentation for troubleshooting failed runs' link is provided at the bottom of the error section.

## Final Output

Once the pipeline runs:

- Python environment is set up
- Pandas is installed
- `alert.py` is executed
- `output/monthly_summary.csv` is generated
- If monthly spending > ₹10,000, alert is shown in logs

Example Output:

⚠ Alert: Total monthly spending is ₹12,300.50, which exceeds ₹10,000.