

Inventory Management System

Pre-requisites:

- ✓ Git installed and configured with SSH or Personal Access Token (PAT)
 - ✓ Python installed (preferably 3.x)
 - ✓ Azure DevOps account with a new project created
 - ✓ Your **local project folder** contains:
 - check_reorder.py
 - final_inventory_dashboard.csv
 - azure-pipelines.yml
-

Step 1: Create Python and YAML Files

1. Python Script (check_reorder.py)

```
import pandas as pd

# Load the inventory CSV file
df = pd.read_csv("final_inventory_dashboard.csv")

# Filter only products that need reorder
reorder_df = df[df["reorder_flag"] == "REORDER"]

# Save the result to a new CSV file
reorder_df.to_csv("reorder_report.csv", index=False)

print("reorder_report.csv generated successfully.")
```

2. Azure DevOps Pipeline File (azure-pipelines.yml)

trigger:

schedule:

```
- cron: "0 7 * * *" # Runs daily at 7 AM UTC
```

```
displayName: Daily 7AM run
```

branches:

include:

```
- main
```

```
always: true

pool:
  vmImage: 'ubuntu-latest'

steps:
  - task: UsePythonVersion@0
    inputs:
      versionSpec: '3.10'
      addToPath: true

  - script: |
      python -m pip install --upgrade pip
      pip install pandas
    displayName: 'Install Dependencies'

  - script: |
      python check_reorder.py
    displayName: 'Run Reorder Check Script'

  - task: PublishBuildArtifacts@1
    inputs:
      PathToPublish: 'reorder_report.csv'
      ArtifactName: 'ReorderReport'
      publishLocation: 'Container'
    displayName: 'Publish Reorder Report Artifact'
```

Step 2: Create a New Azure DevOps Project

1. Go to <https://dev.azure.com/>
2. Click "New Project"

3. Give it a name like `Inventory_Management_System`
4. Set Visibility: Private
5. Click **Create**

The screenshot shows the Azure DevOps interface for a project named "Inventory_Management_System". The left sidebar contains links for Overview, Summary, Dashboards, Wiki, Boards, Repos, Pipelines, Test Plans, Artifacts, and Project settings. The main content area displays the project's name in large bold letters. Below it is the "About this project" section with a brief description: "Create a system to manage inventory levels, track stock movement, and identify products that need to be reordered or are overstocked." To the right of this is the "Project stats" section, which includes a "Period: Last 7 days" dropdown. Under "Repos", it shows 0 pull requests opened. Under "Pipelines", it shows 16 commits by 2 authors.

Step 3: Push Local Project to Azure Repo via SSH

In PowerShell:

```
cd
```

```
"C:\Users\anith\OneDrive\Desktop\Inventory_Management_System\Inventory_Management_System-2"
```

```
git init
```

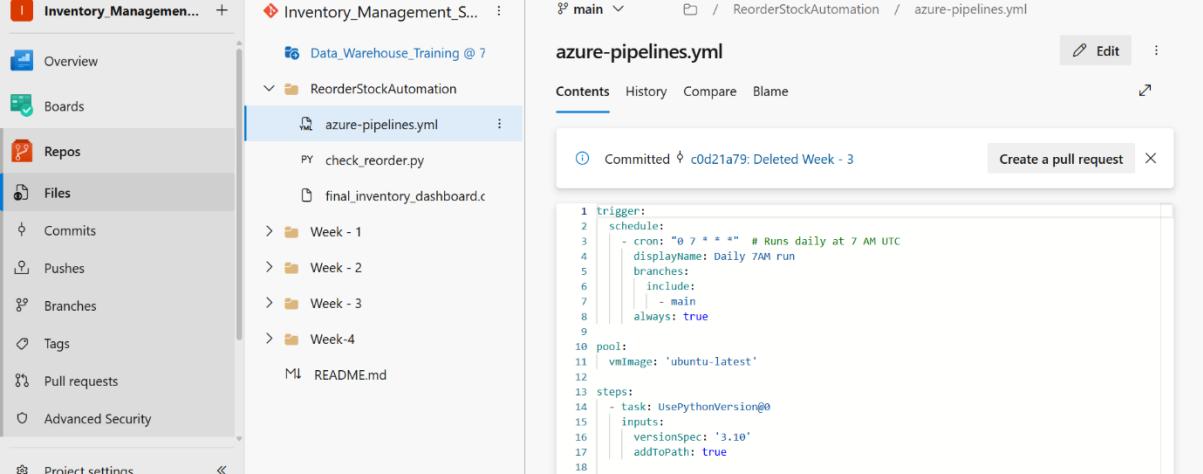
```
git remote add origin
```

```
git@ssh.dev.azure.com:v3/rithi31/Inventory_Management_System/Inventory_Management_System
```

```
git add .
```

```
git commit -m "Initial commit"
```

```
git push -u origin main
```



The screenshot shows the Azure DevOps interface for a project named 'Inventory_Management_Sys...'. The left sidebar is open, showing options like Overview, Boards, Repos, Files (which is selected), Commits, Pushes, Branches, Tags, Pull requests, and Advanced Security. The main area displays the contents of the 'ReorderStockAutomation' folder. Inside, there is a file named 'azure-pipelines.yml'. The code editor shows the following YAML configuration:

```
trigger:
  schedule:
    - cron: "0 7 * * *" # Runs daily at 7 AM UTC
      displayName: Daily 7AM run
  branches:
    include:
      - main
    always: true

pool:
  vmImage: 'ubuntu-latest'

steps:
  - task: UsePythonVersion@0
    inputs:
      versionspec: '3.10'
      addToPath: true
```

Step 4: Configure and Run Azure Pipeline

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

The screenshot shows the 'Where is your code?' section of the Azure DevOps Pipelines interface. On the left, there is a sidebar with project navigation links: Overview, Boards, Repos, Pipelines (selected), Environments, Library, Test Plans, Artifacts, and Project settings. The main area has tabs: Connect, Select, Configure, and Review. The 'Connect' tab is active. Below it, the text 'New pipeline' is displayed. A heading 'Where is your code?' is followed by a list of code sources:

- Azure Repos Git (YAML) - Free private Git repositories, pull requests, and code search
- Bitbucket Cloud (YAML) - Hosted by Atlassian
- GitHub (YAML) - Home to the world's largest community of developers
- GitHub Enterprise Server (YAML) - The self-hosted version of GitHub Enterprise

The screenshot shows the 'Review your pipeline YAML' screen. The sidebar and tabs are identical to the previous screenshot. The main area displays the 'azure-pipelines.yml' file content:

```
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 number
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'
```

A modal window titled 'Save and run' is open on the right side. It contains fields for 'Commit message' (Set up CI with Azure Pipelines) and 'Optional extended description' (Add an optional description...). It also includes two radio buttons for committing: 'Commit directly to the main branch' (selected) and 'Create a new branch for this commit'. A progress indicator at the bottom right says 'Creating pipeline...'.

Step 5: Final Output

- The pipeline will:
 - Install dependencies
 - Run the reorder script
 - Save `reorder_report.csv`

← → × dev.azure.com/rithi31/Inventory_Management_System/_build/results?buildId=7&view=results

Azure DevOps rithi31 / Inventory_Management_Sys... / Pipelines / Inventory_Management_Sys... / 20250701.1 Search ⋮

Inventory_Management... +

Overview

Boards

Repos

Pipelines

Pipelines

Environments

Library

Test Plans

Artifacts

Project settings

#20250701.1 • Set up CI with Azure Pipelines

Inventory_Management_System (4)

Cancel ⋮

Summary Code Coverage

Individual CI by Rithika R

Repository and version

Inventory_Management_System
main f7c53c95

Time started and elapsed

Just now

Related

0 work items

0 artifacts

View 14 changes

Tests and coverage

Get started

Jobs

Name	Status	Duration
Job	Queued	