Data Engineering Batch 1 **Date: 28/02/2024**

Name: Pradip Bochare

Azure DevOps Coding Assessment Question 1

Create Azure Devops Environment and configuring Azure Devops Git Repository, configure on your local git to implement this upload few test files on same.

❖ Azure DevOps Repos

Azure DevOps Repos is a version control service that allows teams to collaborate on code development securely and efficiently. It provides Git repositories or Team Foundation Version Control (TFVC) for source control management, offering features tailored for DevOps practices. Here's a breakdown of key aspects and features of Azure DevOps Repos:

1. **Git Repositories**:
   - Azure DevOps Repos supports Git as its distributed version control system (DVCS).
   - Git repositories provide a decentralized workflow, enabling developers to work offline, commit changes locally, and collaborate effectively.
   - Teams can create multiple Git repositories within a project to organize codebases based on modules, components, or projects.

2. **TFVC (Team Foundation Version Control)**:
   - TFVC is a centralized version control system that's also supported in Azure DevOps Repos.
   - TFVC offers features like exclusive file checkout, atomic commits, and granular access control.
   - While Git is the preferred choice for many modern development workflows, TFVC remains an option for teams accustomed to centralized version control.

3. **Branching and Merging**:
    - Azure DevOps Repos allows teams to create branches to isolate changes, work on features, or experiment without affecting the main codebase.
    - Developers can merge branches back into the main branch (e.g., master or main) using various merge strategies, such as squash, rebase, or merge commit.
    - Branch policies can be enforced to ensure code quality, such as requiring pull request reviews, passing automated tests, and meeting coding standards before merging.

4. **Code Reviews**:
    - Pull Requests (PRs) facilitate code reviews in Azure DevOps Repos.
    - Developers can create PRs to request feedback, discuss changes, and ensure code quality before merging.
    - Reviewers can provide comments, suggest changes, and approve PRs based on predefined criteria.

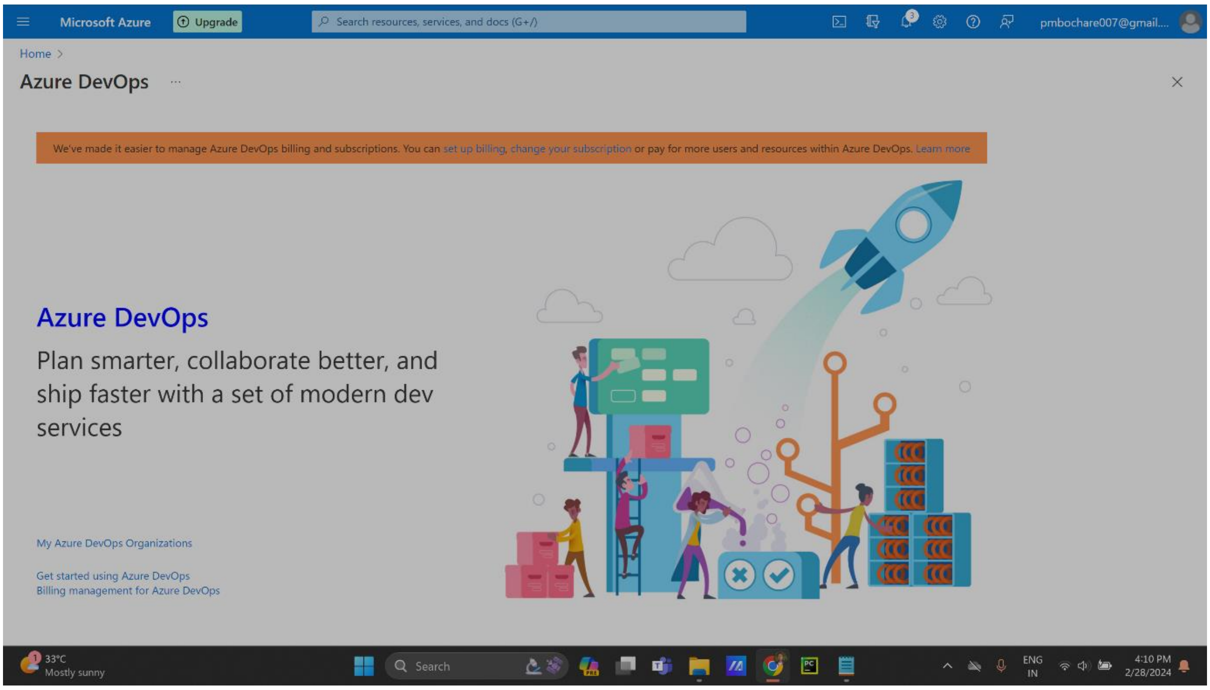
5. **Integration with Azure Pipelines**:
    - Azure DevOps Repos seamlessly integrates with Azure Pipelines, the CI/CD service in Azure DevOps.
    - Developers can trigger automated builds and deployments directly from code changes in repositories.
    - CI/CD pipelines can be defined using YAML or classic UI-based editors to automate the build, test, and deployment processes.

6. **Access Control and Security**:
    - Azure DevOps Repos provides fine-grained access control to repositories, branches, and code.
    - Role-based access control (RBAC) allows administrators to define permissions for users and groups based on their roles and responsibilities.
    - Security features such as branch policies, branch permissions, and code scanning help ensure compliance and protect code assets from vulnerabilities.

7. **Code Search and Code Navigation**:
    - Azure DevOps Repos offers powerful code search capabilities, allowing developers to quickly find code snippets, files, or commits within repositories.
    - Code navigation features like code browsing, file history, and blame annotations help developers understand code changes and trace their origins.

Overall, Azure DevOps Repos provides a robust and feature-rich platform for version control and collaboration, enabling teams to build, test, and deploy software with agility and confidence as part of their DevOps practices.

❖ Creating Azure Devops Environment


Microsoft Azure
Upgrade
Search resources, services, and docs (G+/)
pmbochare007@gmail....
Home >
Azure DevOps

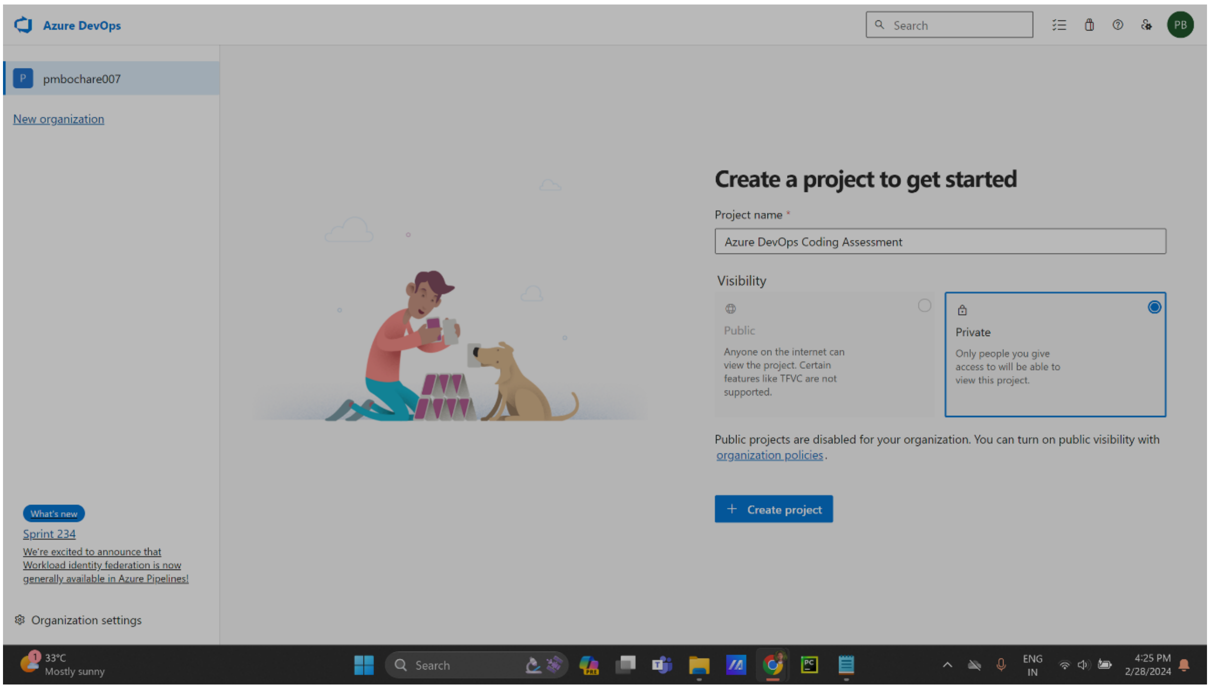
We've made it easier to manage Azure DevOps billing and subscriptions. You can set up billing, change your subscription or pay for more users and resources within Azure DevOps. Learn more

Azure DevOps

Plan smarter, collaborate better, and ship faster with a set of modern dev services

My Azure DevOps Organizations
Get started using Azure DevOps
Billing management for Azure DevOps
33°C
Mostly sunny
Search
ENG
IN
4:10 PM
2/28/2024

Creating Project in Azure DevOps


Azure DevOps
Search
PB
pmbochare007
New organization
What's new
Sprint 234
We're excited to announce that Workload identity federation is now generally available in Azure Pipelines!
Organization settings

Create a project to get started

Project name *

Azure DevOps Coding Assessment

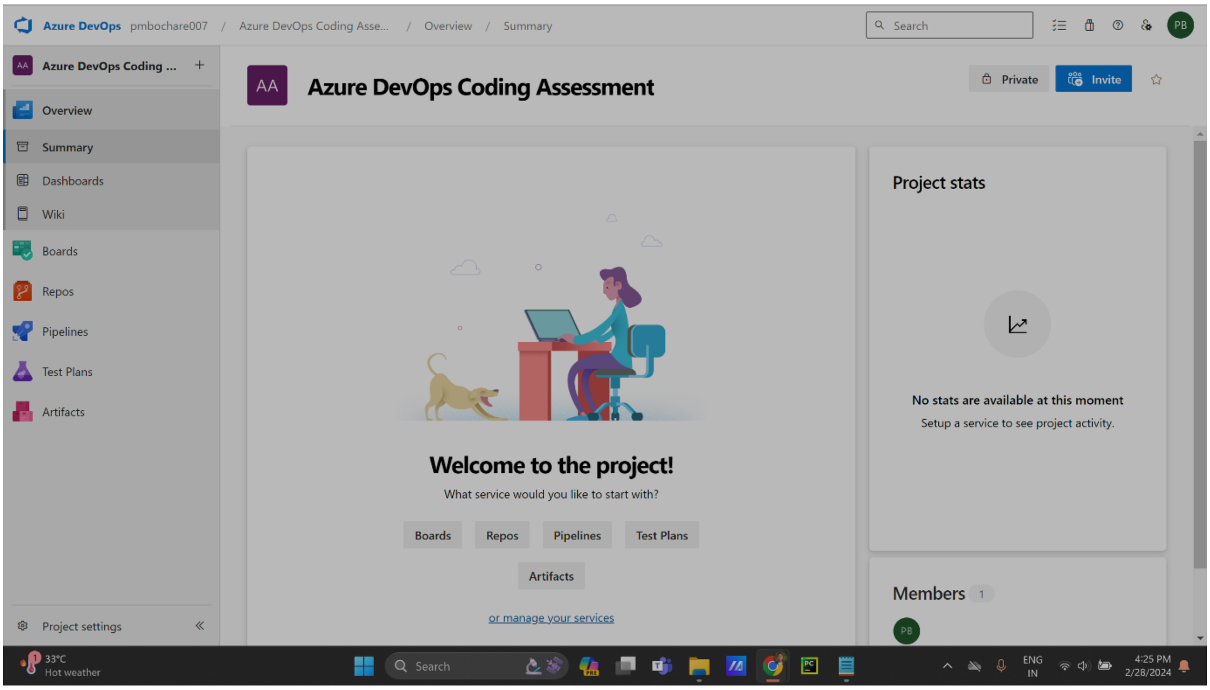
Visibility

○ Public
Anyone on the internet can view the project. Certain features like TFVC are not supported.

● Private
Only people you give access to will be able to view this project.

Public projects are disabled for your organization. You can turn on public visibility with organization policies.

Home page of Azure Devops Project



Azure DevOps Coding Assessment

Welcome to the project!

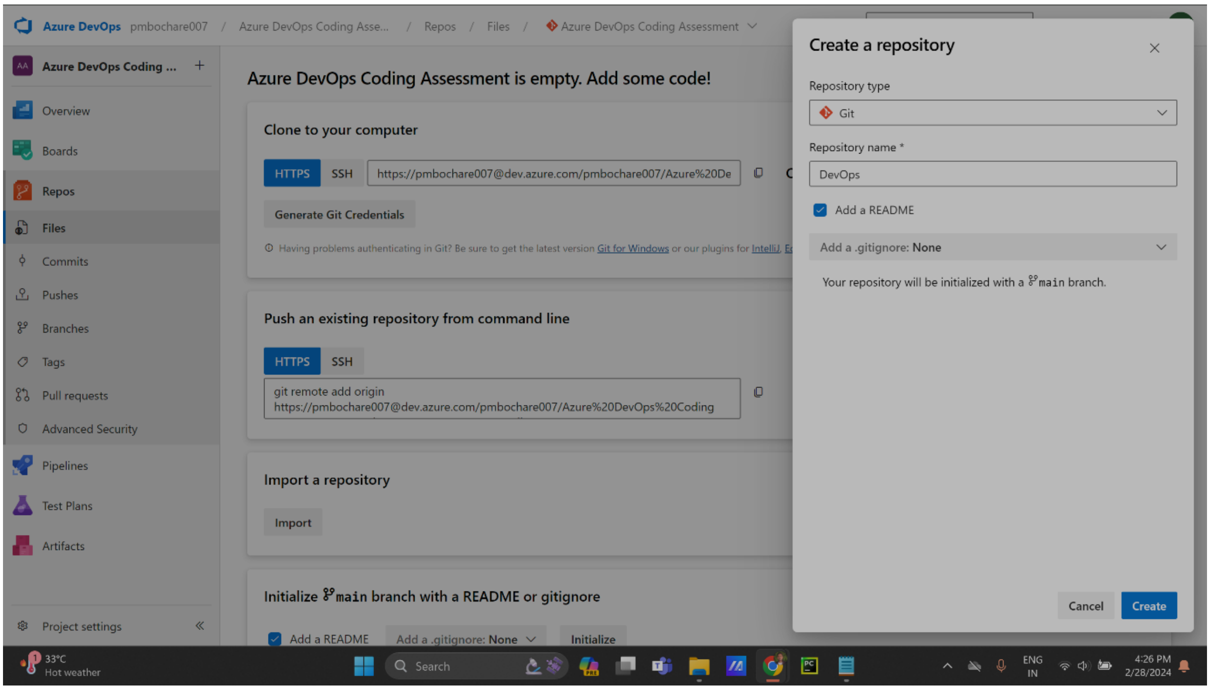
What service would you like to start with?

Project stats

No stats are available at this moment

Setup a service to see project activity.

Making Git Repository in Azure DevOps



Azure DevOps Coding Assessment is empty. Add some code!

Clone to your computer

https://pmbochare007@dev.azure.com/pmbochare007/Azure%20De

Having problems authenticating in Git? Be sure to get the latest version Git for Windows or our plugins for IntelliJ, E

Push an existing repository from command line

```
git remote add origin
https://pmbochare007@dev.azure.com/pmbochare007/Azure%20DevOps%20Coding
```

Import a repository

Initialize main branch with a README or gitignore

☑ Add a README   Add a .gitignore: None   Initialize

Create a repository

Repository type: Git

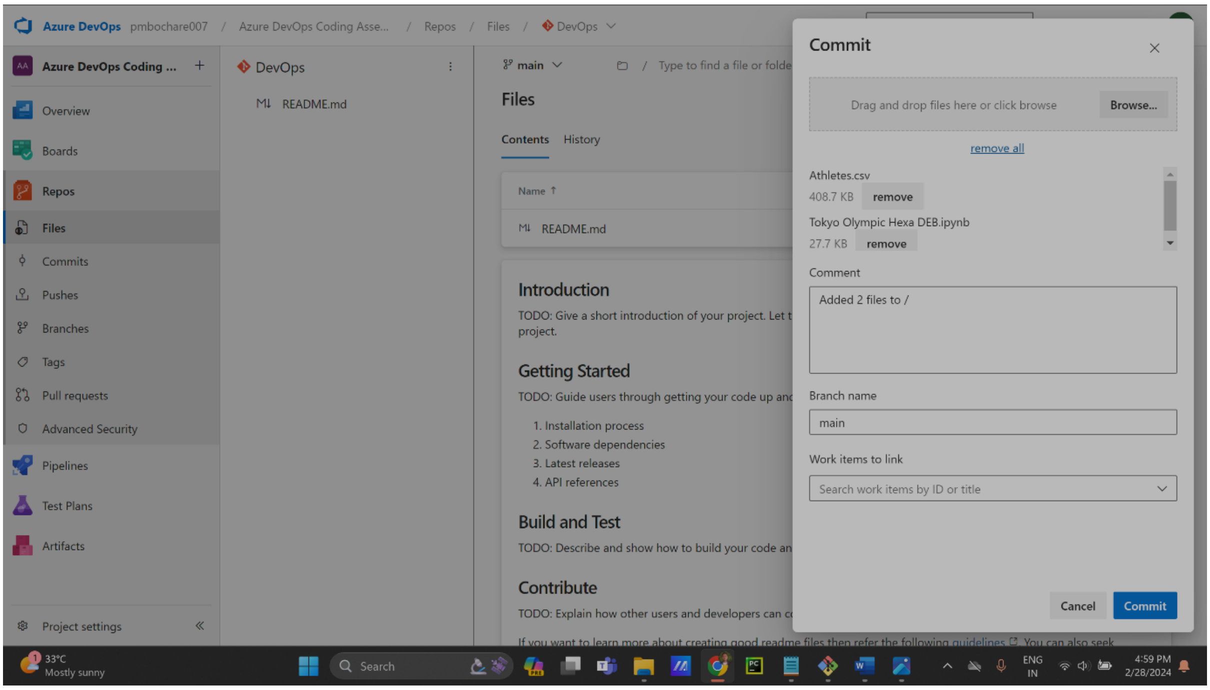
Repository name *: DevOps

☑ Add a README

Add a .gitignore: None

Your repository will be initialized with a main branch.

Adding files in Azure DevOps Repository



Introduction

TODO: Give a short introduction of your project. Let t project.

Getting Started

TODO: Guide users through getting your code up and

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

Build and Test

TODO: Describe and show how to build your code an

Contribute

TODO: Explain how other users and developers can c

If you want to learn more about creating good readme files then refer the following guidelines. You can also seek

Commit

Drag and drop files here or click browse　Browse...

remove all

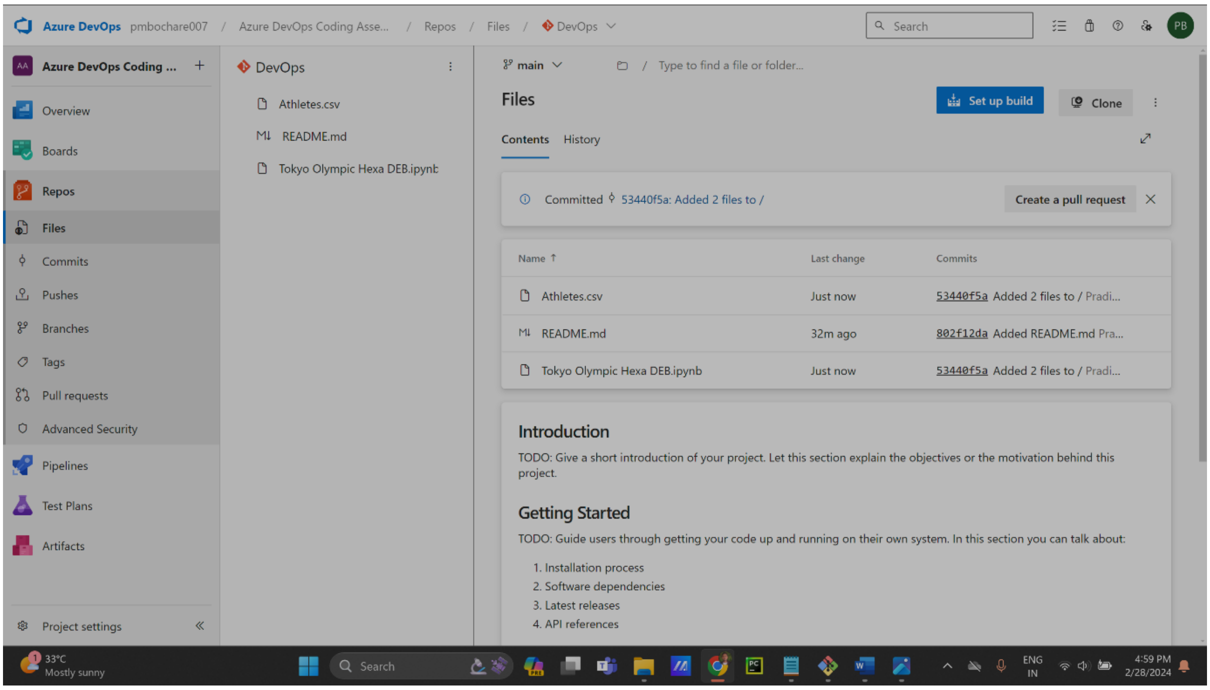
Athletes.csv 408.7 KB	remove
Tokyo Olympic Hexa DEB.ipynb 27.7 KB	remove

Comment: Added 2 files to /

Branch name: main

Work items to link: Search work items by ID or title

Here we can see files added in repository



Committed 53440f5a: Added 2 files to /

Name ↑	Last change	Commits
Athletes.csv	Just now	53440f5a Added 2 files to / Pradi...
README.md	32m ago	802f12da Added README.md Pra...
Tokyo Olympic Hexa DEB.ipynb	Just now	53440f5a Added 2 files to / Pradi...

Introduction

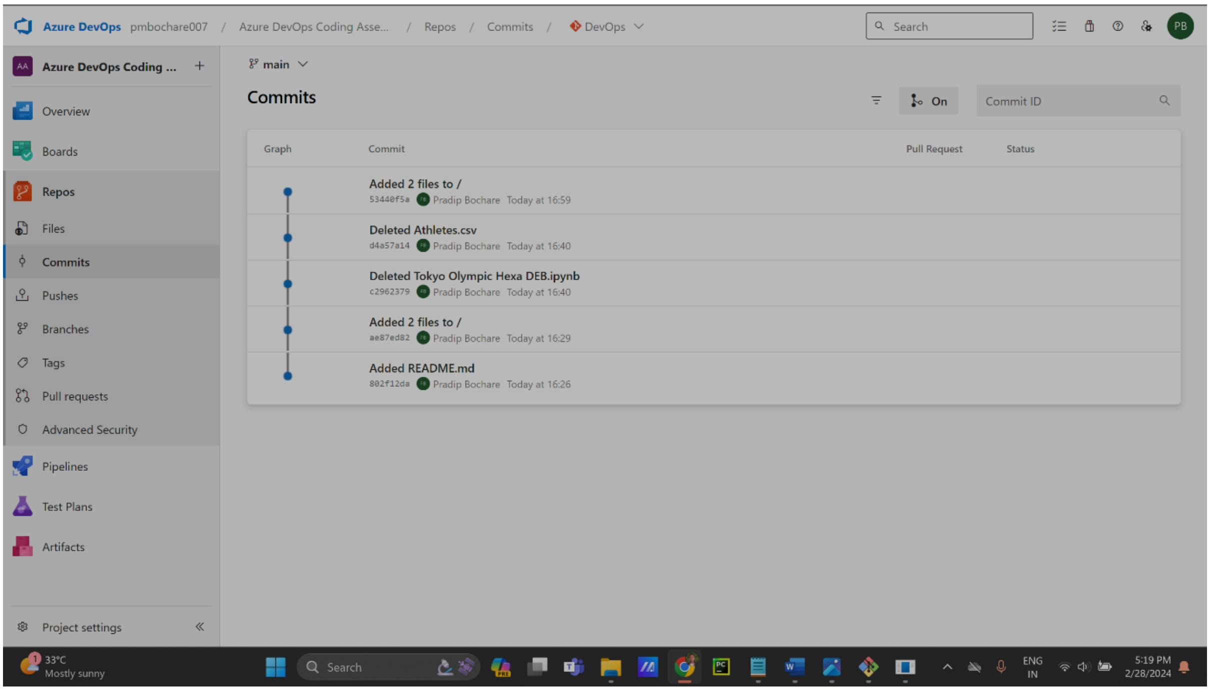
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

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

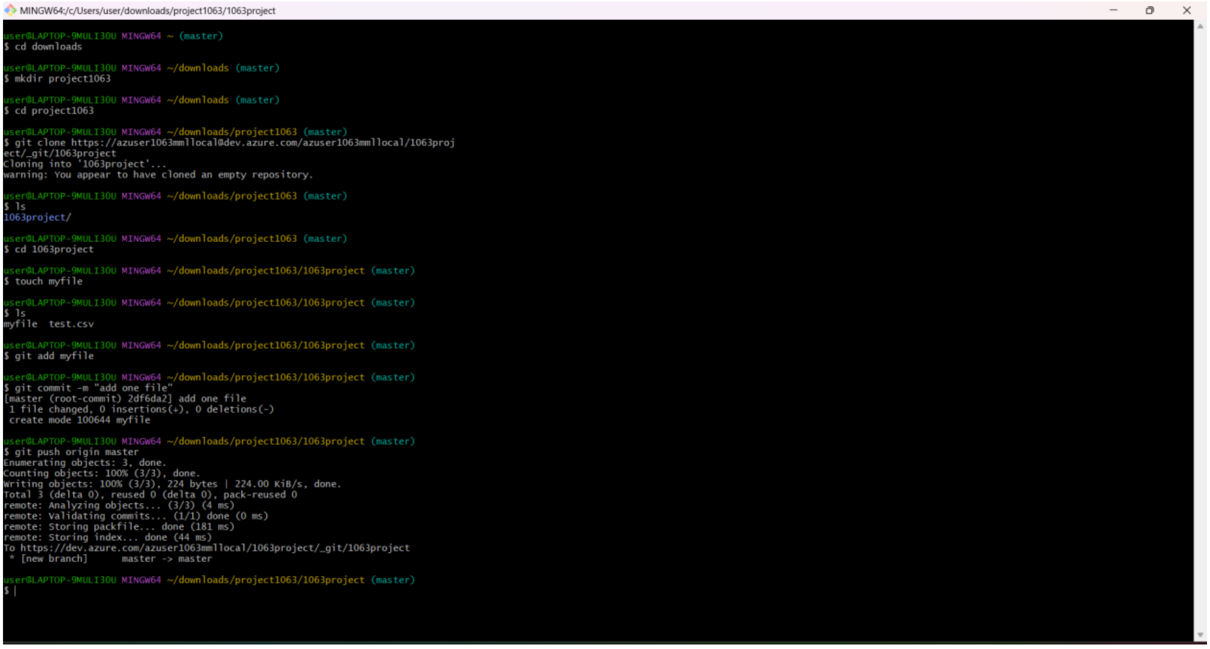
Commits done in repository



Commits

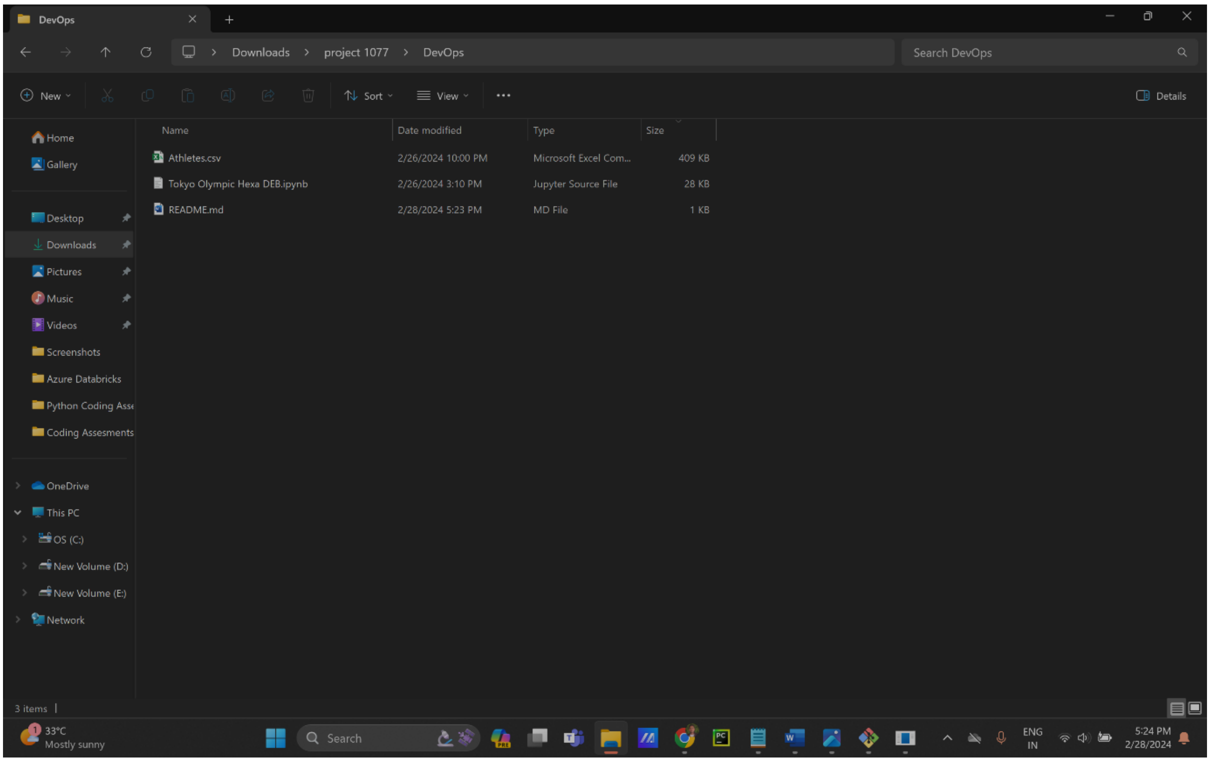
Graph	Commit	Pull Request	Status
●	Added 2 files to / 53440f5a Pradip Bochare Today at 16:59		
●	Deleted Athletes.csv d4a57a14 Pradip Bochare Today at 16:40		
●	Deleted Tokyo Olympic Hexa DEB.ipynb c2962379 Pradip Bochare Today at 16:40		
●	Added 2 files to / ae87ed82 Pradip Bochare Today at 16:29		
●	Added README.md 802f12da Pradip Bochare Today at 16:26		

❖ Opening Git Bash and Performing Commands in it



```
user@LAPTOP-9MULI30U MINGW64 ~ (master)
$ cd downloads

user@LAPTOP-9MULI30U MINGW64 ~/downloads (master)
$ mkdir project1063

user@LAPTOP-9MULI30U MINGW64 ~/downloads (master)
$ cd project1063

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063 (master)
$ git clone https://azuser1063mmllocal@dev.azure.com/azuser1063mmllocal/1063proj
ect/_git/1063project
Cloning into '1063project'...
warning: You appear to have cloned an empty repository.

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063 (master)
$ ls
1063project/

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063 (master)
$ cd 1063project

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ touch myfile

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ ls
myfile  test.csv

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ git add myfile

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ git commit -m "add one file"
[master (root-commit) 2df6da2] add one file
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 myfile

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Writing objects: 100% (3/3), 224 bytes | 224.00 KiB/s, done.
Total 3 (delta 0), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (3/3) (4 ms)
remote: Validating commits... (1/1) done (0 ms)
remote: Storing packfile... done (181 ms)
remote: Storing index... done (44 ms)
To https://dev.azure.com/azuser1063mmllocal/1063project/_git/1063project
 * [new branch]      master -> master

user@LAPTOP-9MULI30U MINGW64 ~/downloads/project1063/1063project (master)
$ |
```

Here we can see repository downloaded to local



Name	Date modified	Type	Size
Athletes.csv	2/26/2024 10:00 PM	Microsoft Excel Com...	409 KB
Tokyo Olympic Hexa DEB.ipynb	2/26/2024 3:10 PM	Jupyter Source File	28 KB
README.md	2/28/2024 5:23 PM	MD File	1 KB

❖ Key Points of Azure Repos

1. Git and TFVC Support: Azure DevOps Repos supports both Git and Team Foundation Version Control (TFVC), offering flexibility for version control.
2. Branching and Merging: Allows for efficient branching strategies and merging capabilities to manage code changes effectively.
3. Pull Requests: Facilitates code reviews, discussions, and approvals through pull requests, ensuring code quality and collaboration.
4. Integration with Azure Pipelines: Seamlessly integrates with Azure Pipelines for automating builds, tests, and deployments based on code changes.
5. Access Control and Security: Provides fine-grained access control and security features to protect code assets and ensure compliance.
6. Code Search and Navigation: Offers powerful code search and navigation capabilities for quickly finding and understanding code within repositories.

7. Collaboration Features: Enables collaboration among team members with features like inline comments, code reviews, and notifications.
8. Built-in Code Insights: Provides built-in code metrics, code coverage, and code scanning capabilities to improve code quality and identify issues early.
9. Integration with Azure DevOps Boards: Integrates with Azure DevOps Boards for linking work items, tracking progress, and managing development workflows.
10. Extensible and Customizable: Allows for customization and extension through APIs, extensions, and integration with third-party tools and services.