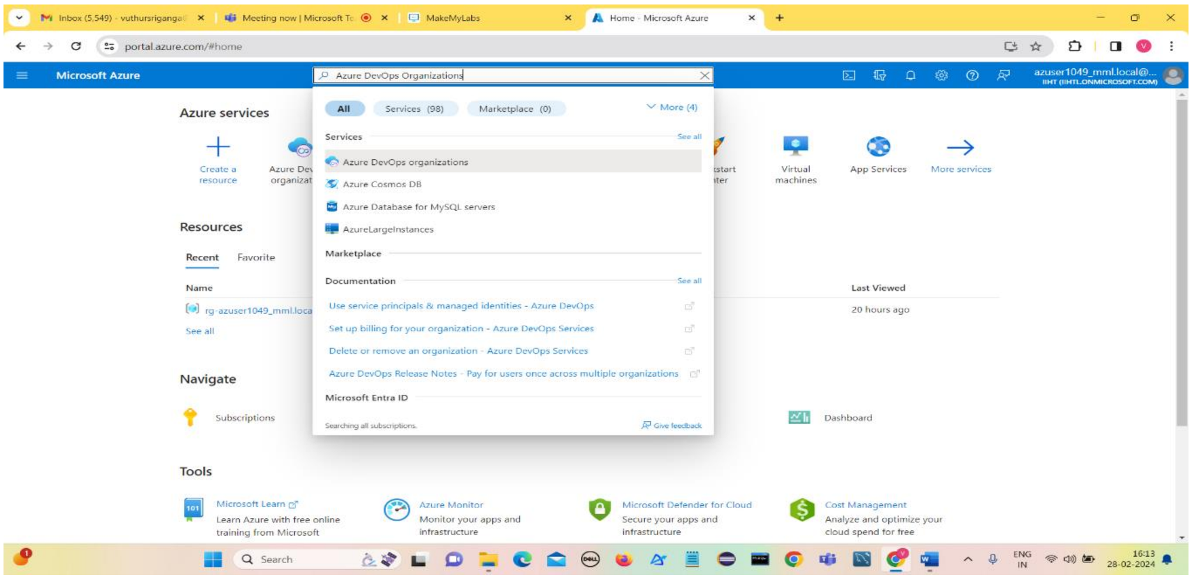
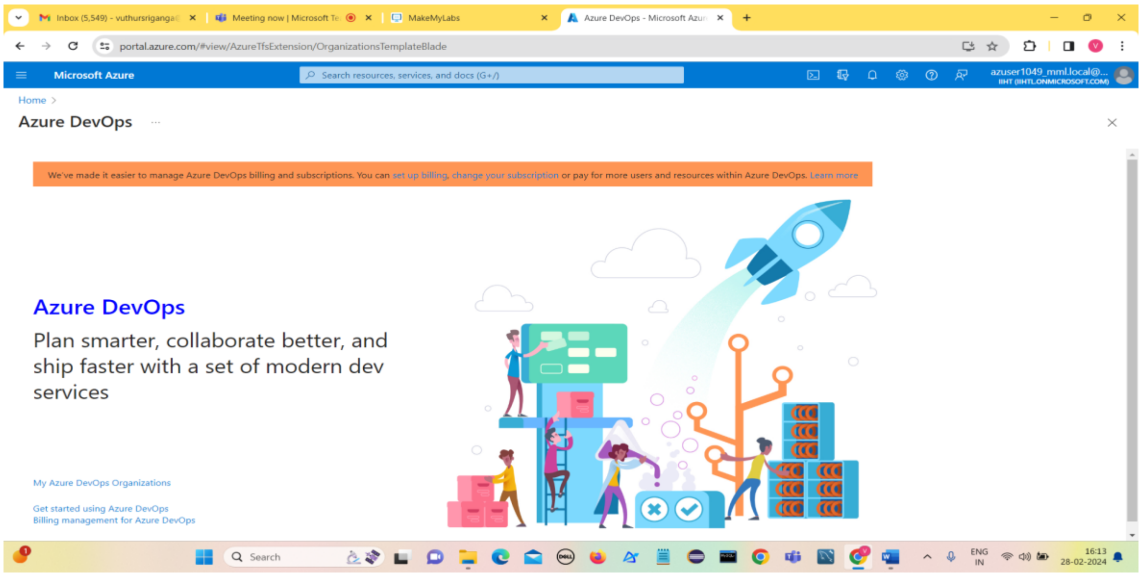
Azure DevOps and Synapse Coding Assessment

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.

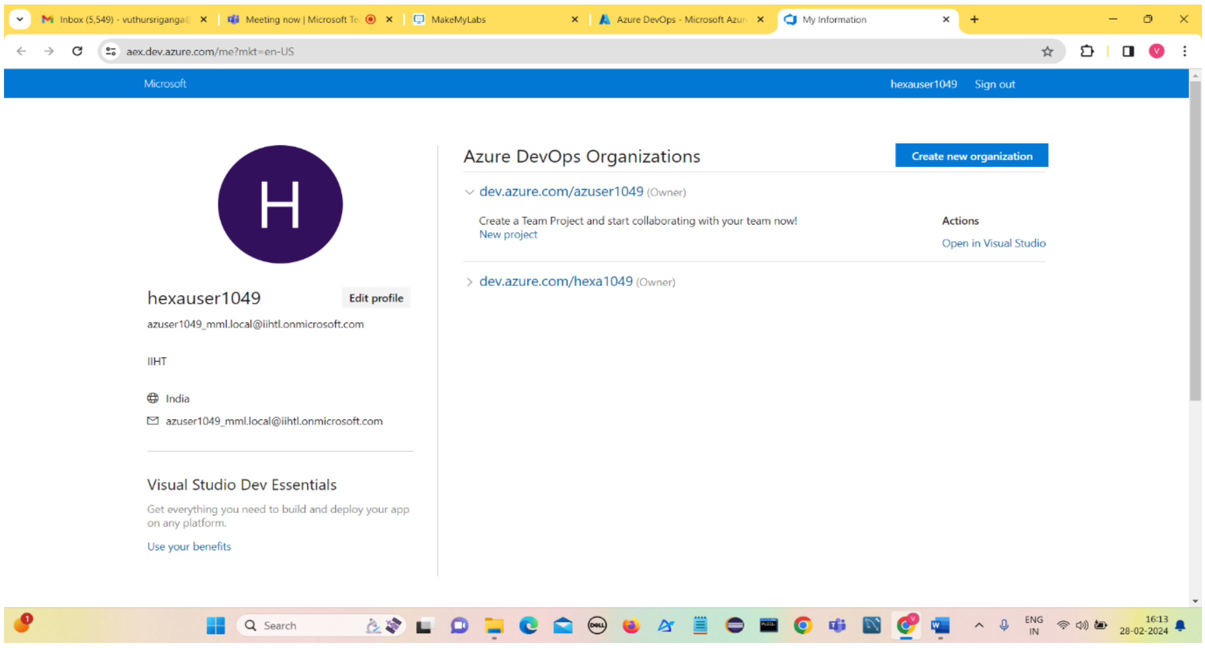
Steps to be followed to set up an Azure DevOps Organization.

Open Azure Portal and Search for Azure DevOps Organization.


portal.azure.com/#home
Microsoft Azure
Azure DevOps Organizations
All
Services (98)
Marketplace (0)
More (4)
Services
Azure DevOps organizations
Azure Cosmos DB
Azure Database for MySQL servers
AzureLargeInstances
Marketplace
Documentation
Use service principals & managed identities - Azure DevOps
Set up billing for your organization - Azure DevOps Services
Delete or remove an organization - Azure DevOps Services
Azure DevOps Release Notes - Pay for users once across multiple organizations
Microsoft Entra ID
Searching all subscriptions.
Give feedback
Azure services
Create a resource
Virtual machines
App Services
More services
Resources
Recent
Favorite
Navigate
Subscriptions
Dashboard
Tools
Microsoft Learn
Azure Monitor
Microsoft Defender for Cloud
Cost Management


portal.azure.com/#view/AzureTfsExtension/OrganizationsTemplateBlade
Microsoft Azure
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

Click on My Azure DevOps Organizations


Inbox (5,549) - vuthursriganga
Meeting now | Microsoft Te
MakeMyLabs
Azure DevOps - Microsoft Azure
My Information
aex.dev.azure.com/me?mkt=en-US
Microsoft
hexauser1049
Sign out

Azure DevOps Organizations

Create new organization

dev.azure.com/azuser1049 (Owner)

Create a Team Project and start collaborating with your team now!
New project

Actions
Open in Visual Studio

dev.azure.com/hexa1049 (Owner)

hexauser1049 Edit profile

azuser1049_mml.local@iihtl.onmicrosoft.com

IIHT

India

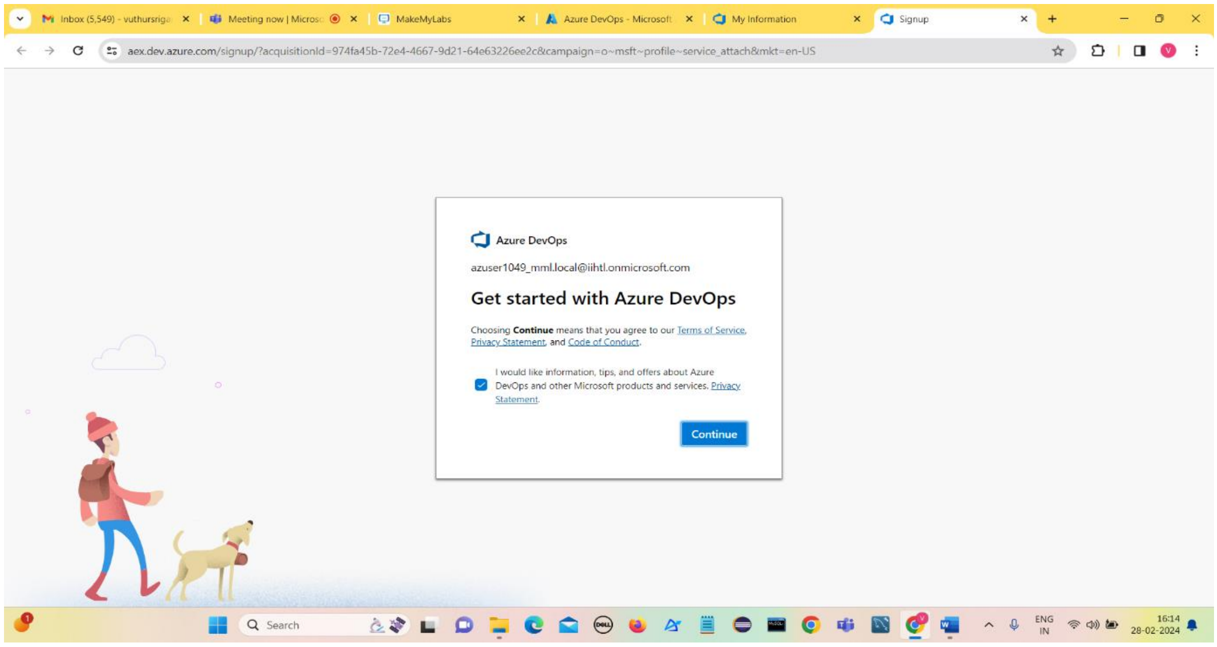
azuser1049_mml.local@iihtl.onmicrosoft.com

Visual Studio Dev Essentials

Get everything you need to build and deploy your app on any platform.

Use your benefits

Click on Create new organization.


Inbox (5,549) - vuthursriga
Meeting now | Microso
MakeMyLabs
Azure DevOps - Microsoft
My Information
Signup
aex.dev.azure.com/signup/?acquisitionId=974fa45b-72e4-4667-9d21-64e63226ee2c&campaign=o~msft~profile~service_attach&mkt=en-US

Azure DevOps

azuser1049_mml.local@iihtl.onmicrosoft.com

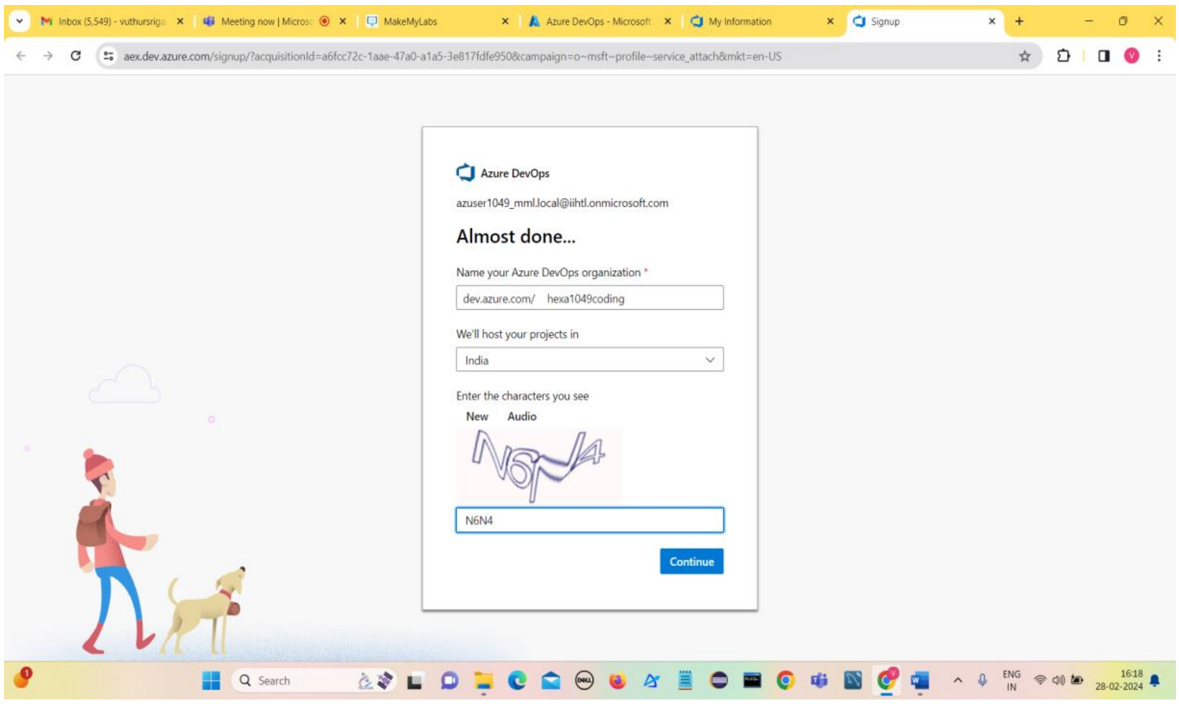
Get started with Azure DevOps

Choosing **Continue** means that you agree to our Terms of Service, Privacy Statement, and Code of Conduct.

[✓] I would like information, tips, and offers about Azure DevOps and other Microsoft products and services. Privacy Statement.

Continue

Click on Continue


aex.dev.azure.com/signup/?acquisitionId=a6fcc72c-1aae-47a0-a1a5-3e817fdfe950&campaign=o~msft~profile~service_attach&mkt=en-US

Azure DevOps

azuser1049_mml.local@iihtl.onmicrosoft.com

Almost done...

Name your Azure DevOps organization *

dev.azure.com/ hexa1049coding

We'll host your projects in

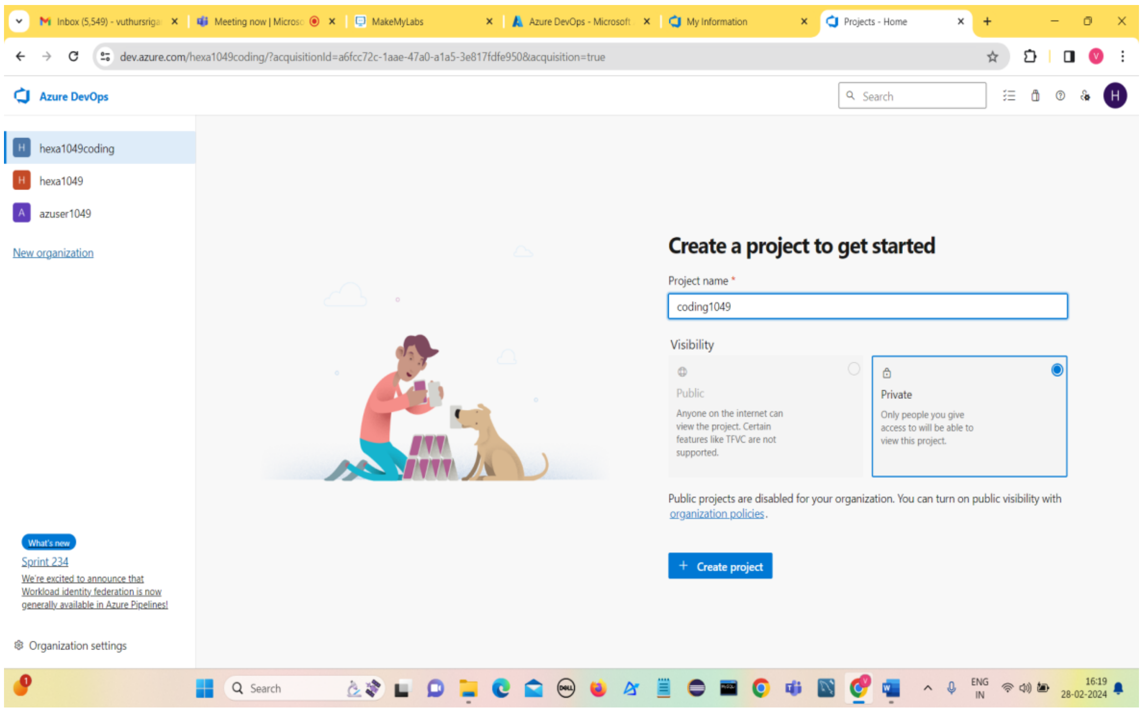
India

Enter the characters you see

New Audio

N6N4

Give the name of Azure DevOps organization and the hosting country , Click on Continue.


dev.azure.com/hexa1049coding/?acquisitionId=a6fcc72c-1aae-47a0-a1a5-3e817fdfe950&acquisition=true

Sprint 234

We're excited to announce that Workload identity federation is now generally available in Azure Pipelines!

Create a project to get started

Project name *

coding1049

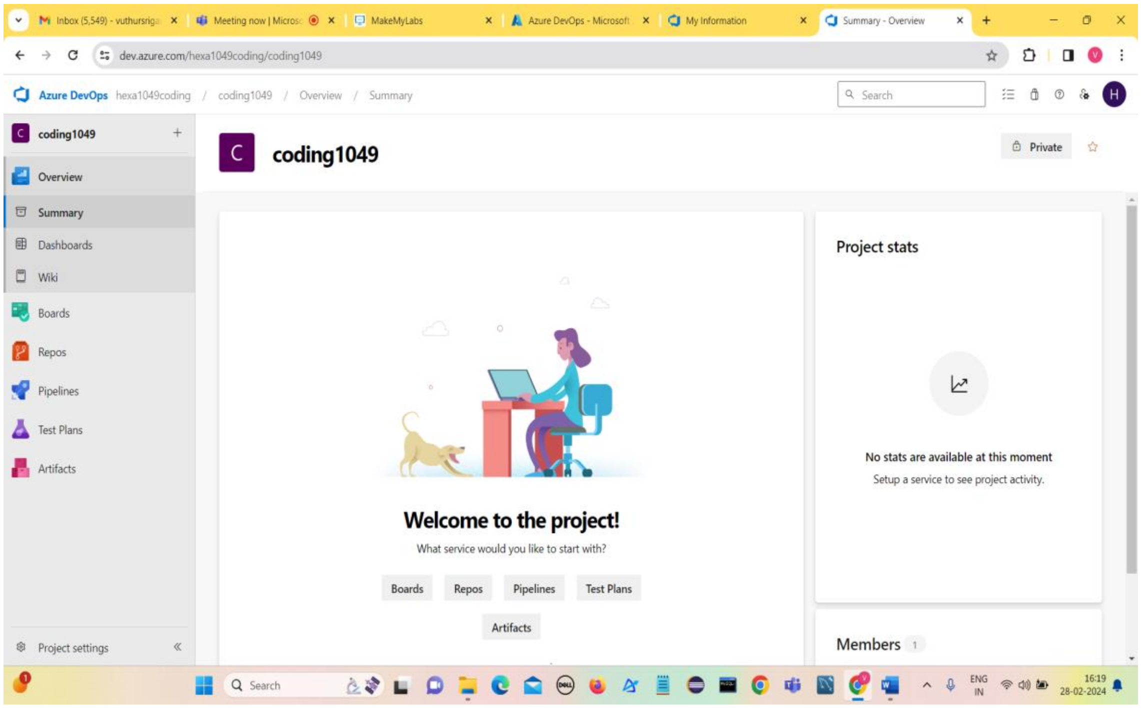
Visibility

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

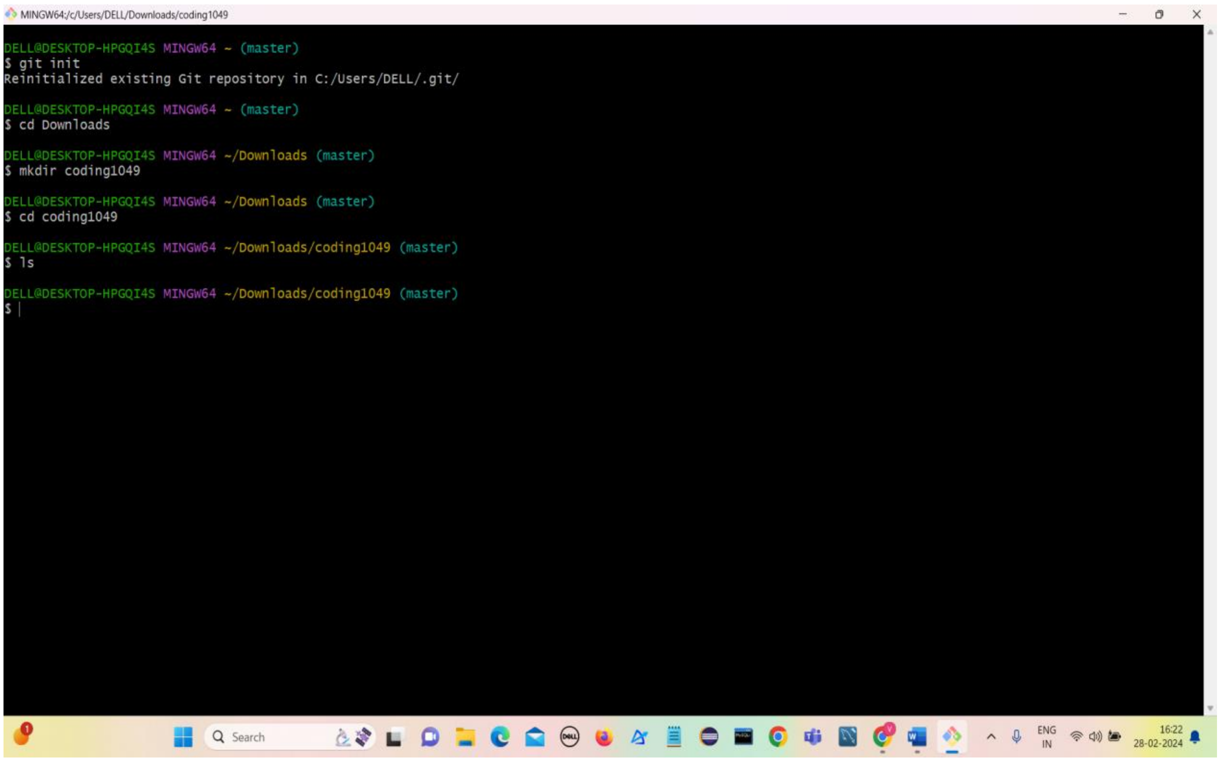
[✓ Private]
Only people you give access to will be able to view this project.

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

Give the name of project and click on Create project.



Go to Git Bash terminal and create a folder (coding1049) in Downloads using mkdir command

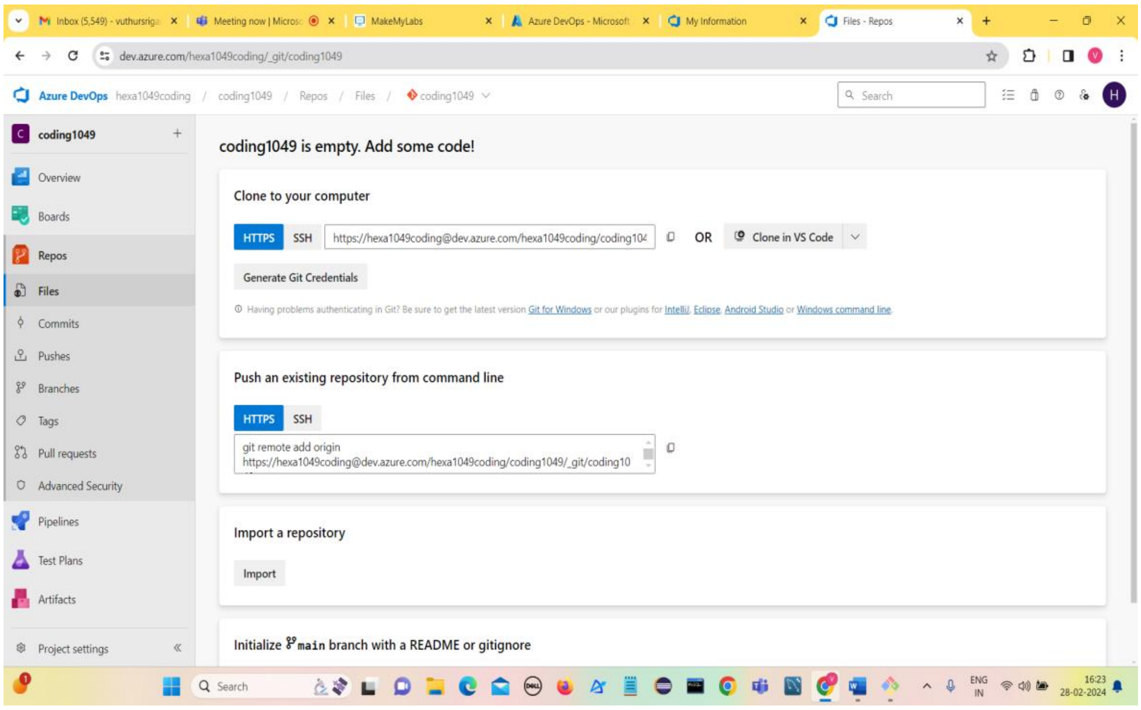


```
DELL@DESKTOP-HPGQI45 MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/DELL/.git/

DELL@DESKTOP-HPGQI45 MINGW64 ~ (master)
$ cd Downloads

DELL@DESKTOP-HPGQI45 MINGW64 ~/Downloads (master)
$ mkdir coding1049

DELL@DESKTOP-HPGQI45 MINGW64 ~/Downloads (master)
$ cd coding1049

DELL@DESKTOP-HPGQI45 MINGW64 ~/Downloads/coding1049 (master)
$ ls

DELL@DESKTOP-HPGQI45 MINGW64 ~/Downloads/coding1049 (master)
$ |
```

We observe that there are no files in the folder it is empty.

Cloning the Repository to local:

Copy url from repos in the DevOps project.



coding1049 is empty. Add some code!

Clone to your computer

HTTPS SSH https://hexa1049coding@dev.azure.com/hexa1049coding/coding104 OR Clone in VS Code

Generate Git Credentials

Having problems authenticating in Git? Be sure to get the latest version Git for Windows or our plugins for IntelliJ, Eclipse, Android Studio or Windows command line.

Push an existing repository from command line

HTTPS SSH

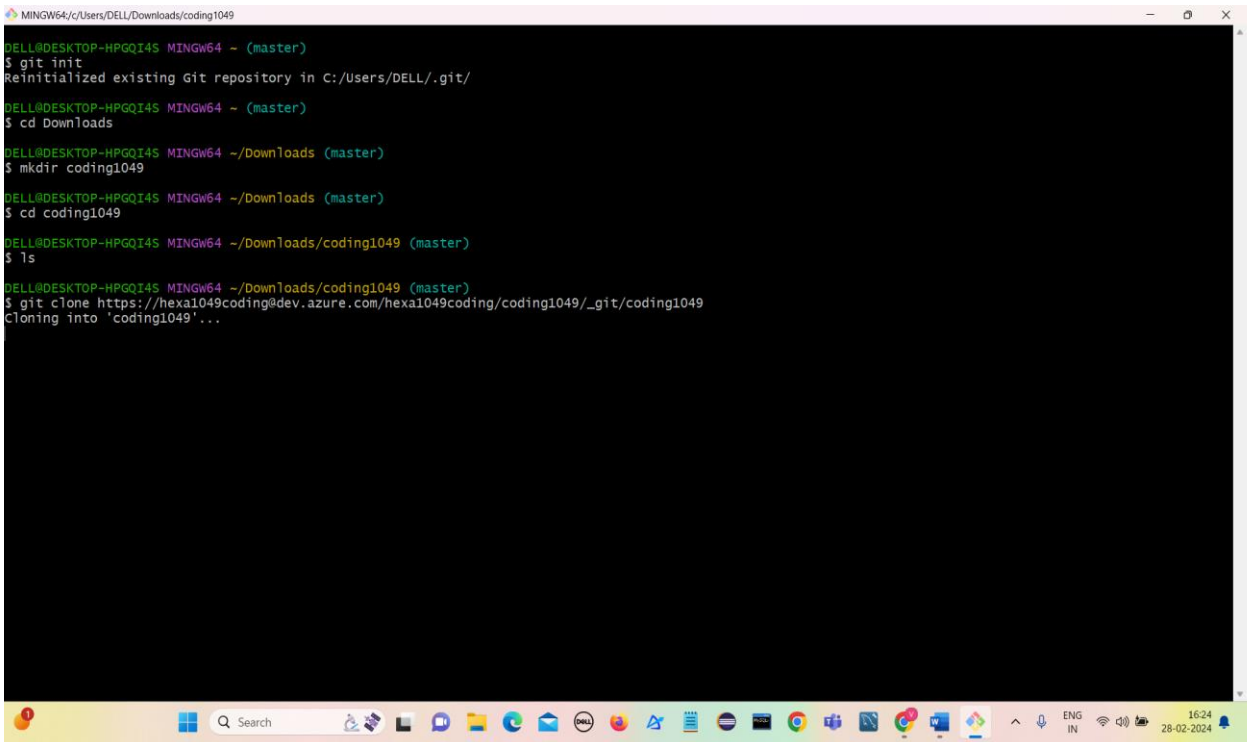
```
git remote add origin
https://hexa1049coding@dev.azure.com/hexa1049coding/coding1049/_git/coding10
```

Import a repository

Import

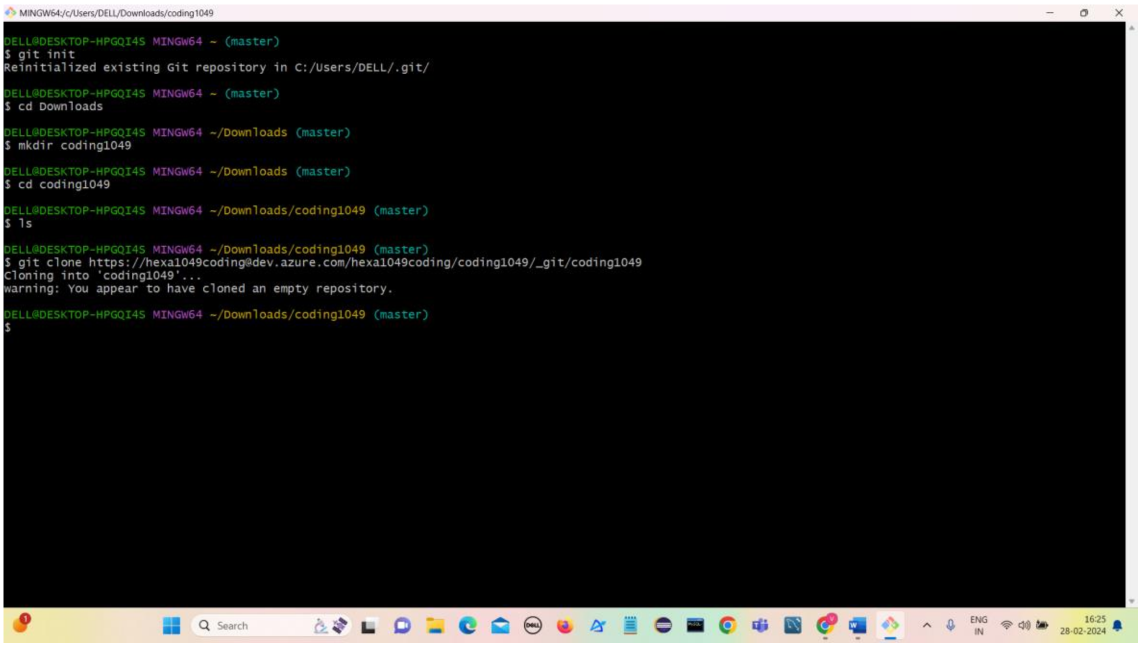
Initialize main branch with a README or gitignore

Use command **git clone url**



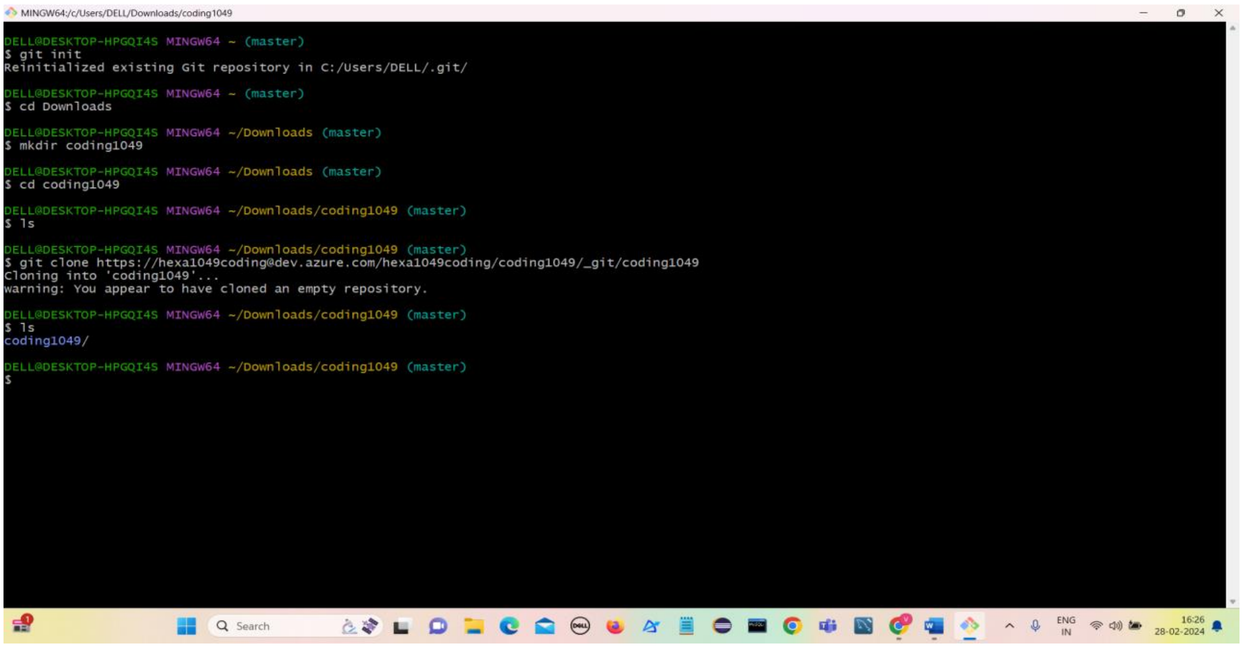
```
DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/DELL/.git/

DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ cd Downloads

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ mkdir coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ cd coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ ls

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ git clone https://hexa1049coding@dev.azure.com/hexa1049coding/coding1049/_git/coding1049
Cloning into 'coding1049'...
```

It shows we have cloned and empty repository.



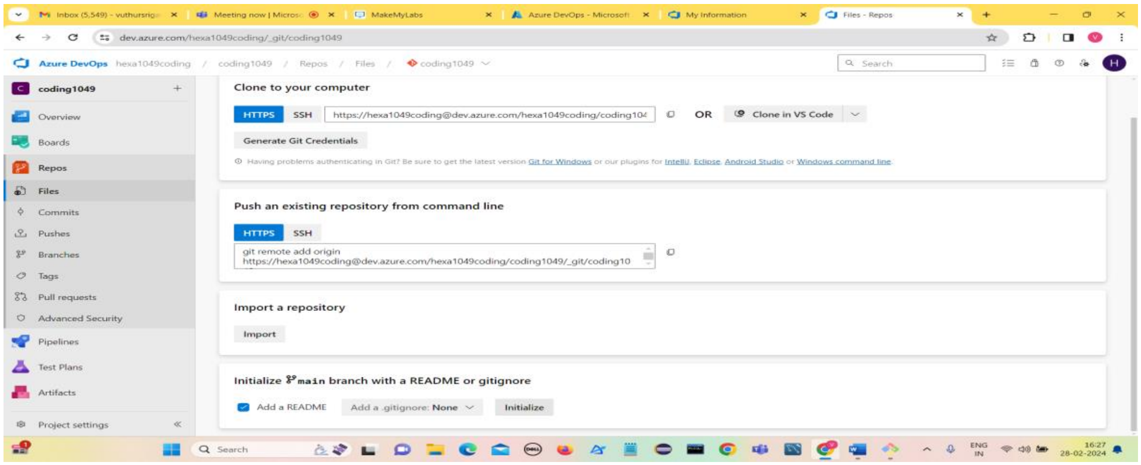
```
MINGW64:/c/Users/DELL/Downloads/coding1049
DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/DELL/.git/

DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ cd Downloads

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ mkdir coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ cd coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ ls

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ git clone https://hexa1049coding@dev.azure.com/hexa1049coding/coding1049/_git/coding1049
Cloning into 'coding1049'...
warning: You appear to have cloned an empty repository.

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$
```

The folder got created in the coding1049 after cloning the repository.



```
MINGW64:/c/Users/DELL/Downloads/coding1049
DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ git init
Reinitialized existing Git repository in C:/Users/DELL/.git/

DELL@DESKTOP-HPGQI4S MINGW64 ~ (master)
$ cd Downloads

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ mkdir coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads (master)
$ cd coding1049

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ ls

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ git clone https://hexa1049coding@dev.azure.com/hexa1049coding/coding1049/_git/coding1049
Cloning into 'coding1049'...
warning: You appear to have cloned an empty repository.

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$ ls
coding1049/

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049 (master)
$
```

Click on Initialize to add a readME file into the project.


Clone to your computer

HTTPS SSH https://hexa1049coding@dev.azure.com/hexa1049coding/coding104 OR Clone in VS Code

Generate Git Credentials

Having problems authenticating in Git? Be sure to get the latest version Git for Windows or our plugins for IntelliJ, Eclipse, Android Studio or Windows command line.

Push an existing repository from command line

HTTPS SSH

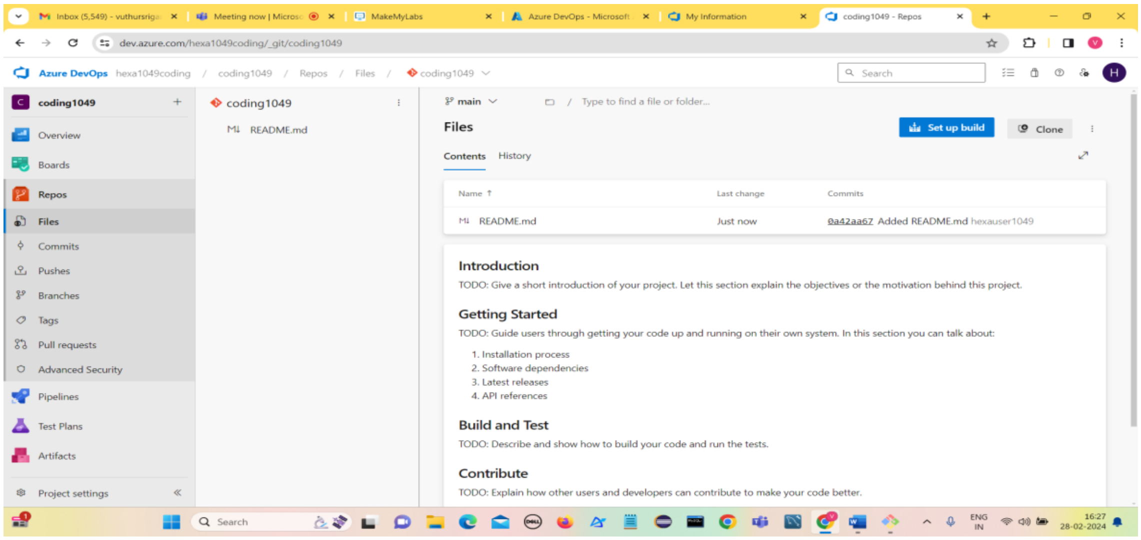
```
git remote add origin
https://hexa1049coding@dev.azure.com/hexa1049coding/coding1049/_git/coding10
```

Import a repository

Import

Initialize main branch with a README or gitignore

Add a README Add a .gitignore: None Initialize


Files

Name	Last change	Commits
README.md	Just now	0a42aa67 Added README.md hexauser1049

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

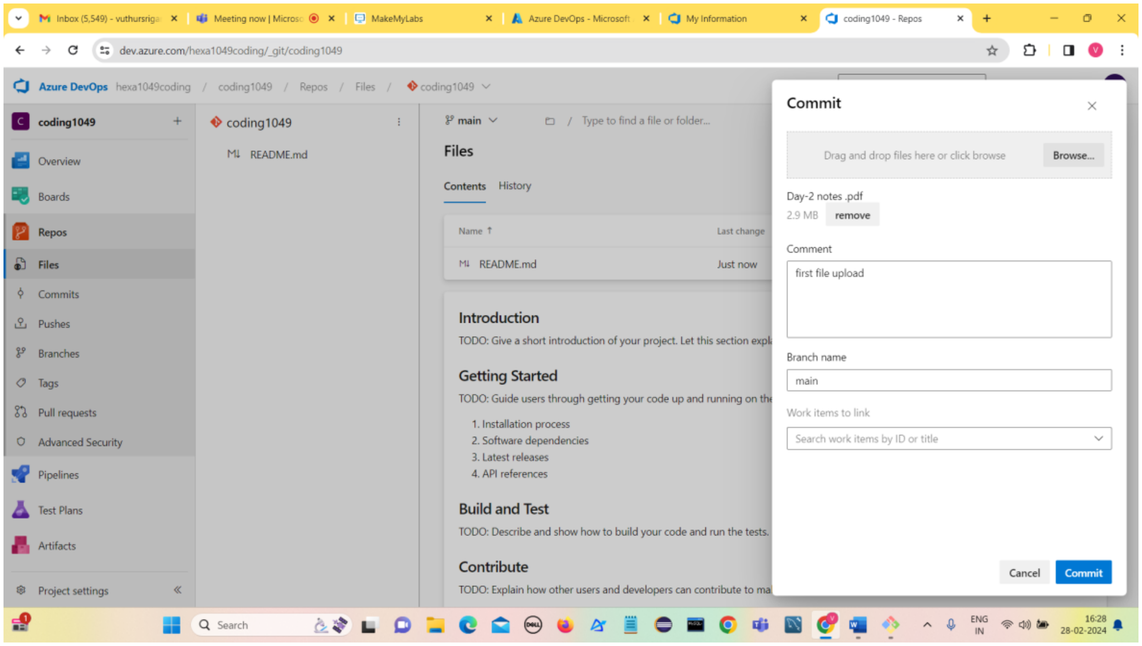
Build and Test

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

Contribute

TODO: Explain how other users and developers can contribute to make your code better.

Uploading a file Day-2 notes.pdf into the project.


Commit

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

Day-2 notes .pdf
2.9 MB remove

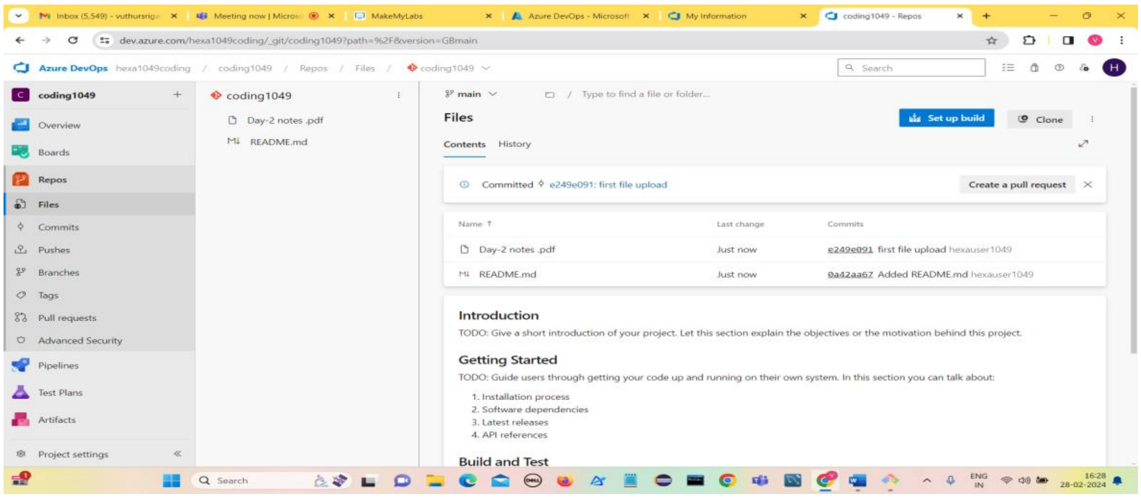
Comment: first file upload

Branch name: main

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

Cancel Commit

Click on Commit



Committed e249e091: first file upload

Name ↑	Last change	Commits
Day-2 notes .pdf	Just now	e249e091 first file upload hexauser1049
README.md	Just now	0a42aa67 Added README.md hexauser1049

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:

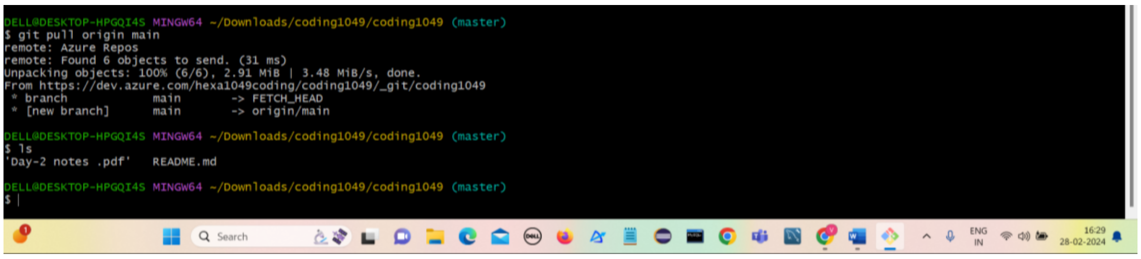
1. Installation process
2. Software dependencies
3. Latest releases
4. API references

Build and Test

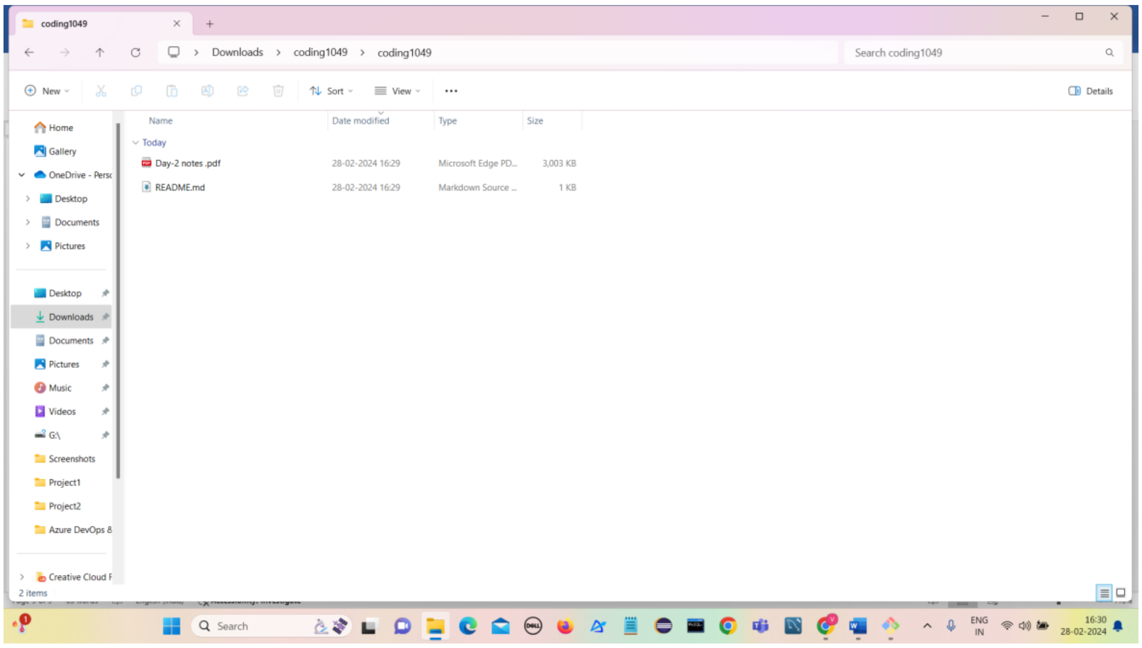
We observe that file got added.

Pull the uploaded file from project to local

Use the command git pull origin main



```
DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git pull origin main
remote: Azure Repos
remote: Found 6 objects to send. (31 ms)
Unpacking objects: 100% (6/6), 2.91 MiB | 3.48 MiB/s, done.
From https://dev.azure.com/hexa1049coding/coding1049/_git/coding1049
 * branch            main       -> FETCH_HEAD
 * [new branch]      main       -> origin/main

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ ls
'Day-2 notes .pdf'    README.md

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ |
```



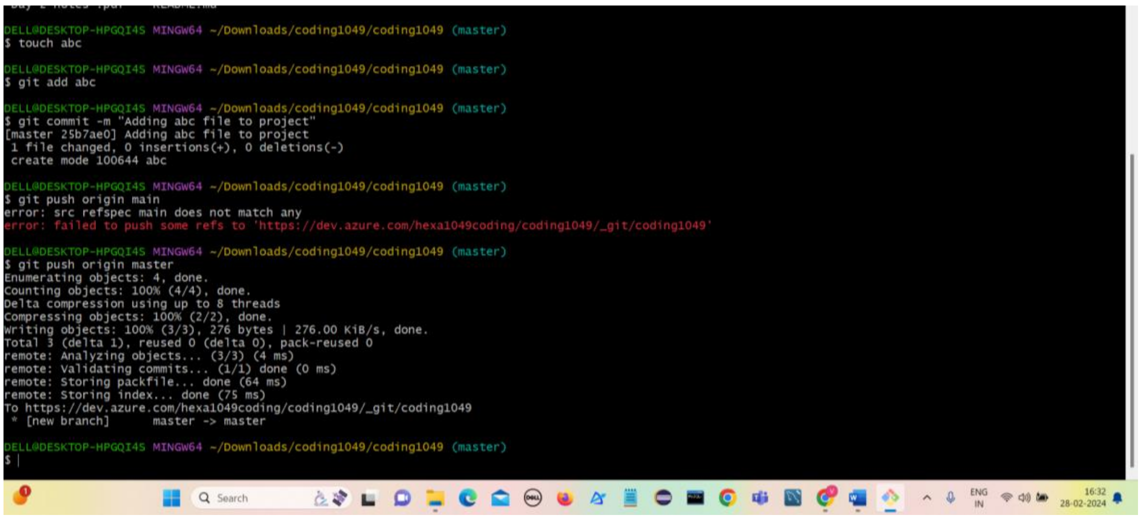
Name	Date modified	Type	Size
Today			
Day-2 notes .pdf	28-02-2024 16:29	Microsoft Edge PD...	3,003 KB
README.md	28-02-2024 16:29	Markdown Source ...	1 KB

We observe that file got pulled into local from the repository.

Add files in project using git

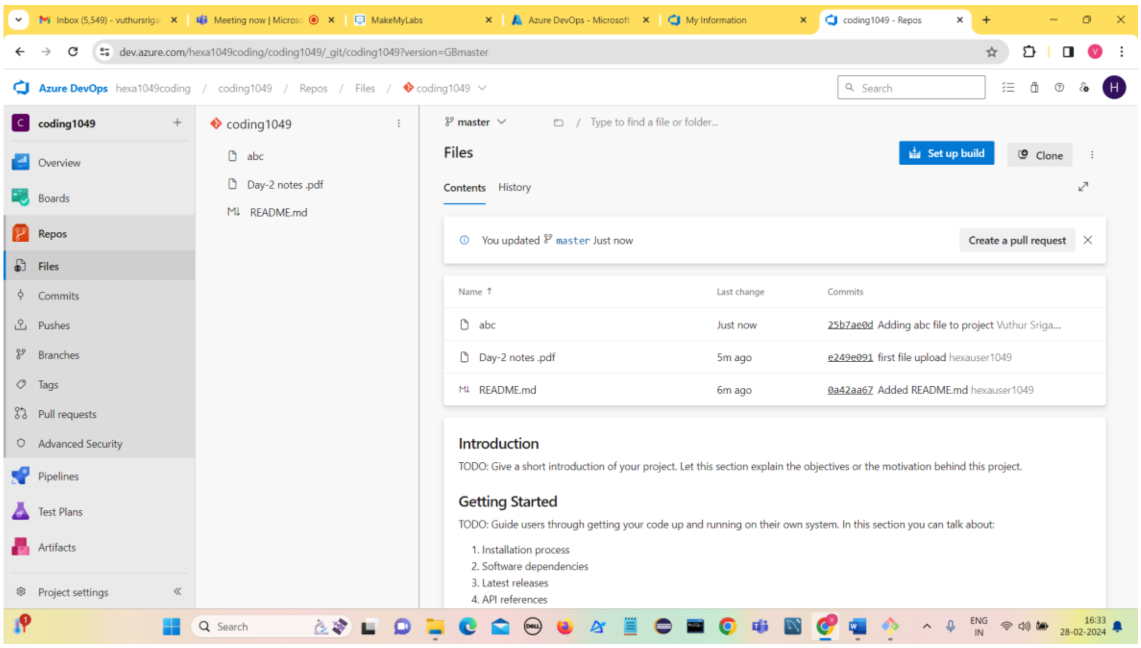
Commands used are:

1) touch abc
2) git add abc
3) git commit -m “Adding abc file to project”
4) git push origin master



```
DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ touch abc

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git add abc

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git commit -m "Adding abc file to project"
[master 25b7ae0] Adding abc file to project
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 abc

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git push origin main
error: src refspec main does not match any
error: failed to push some refs to 'https://dev.azure.com/hexa1049coding/coding1049/_git/coding1049'

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git push origin master
Enumerating objects: 4, done.
Counting objects: 100% (4/4), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (3/3), 276 bytes | 276.00 KiB/s, done.
Total 3 (delta 1), 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 (64 ms)
remote: Storing index... done (75 ms)
To https://dev.azure.com/hexa1049coding/coding1049/_git/coding1049
 * [new branch]      master -> master

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ |
```

abc file got added to project



dev.azure.com/hexa1049coding/coding1049/_git/coding1049?version=GBmaster

Azure DevOps hexa1049coding / coding1049 / Repos / Files / coding1049

master / Type to find a file or folder...

Files

Contents History

You updated master Just now — Create a pull request

Name ↑	Last change	Commits
abc	Just now	25b7ae0d Adding abc file to project Vuthur Sriga...
Day-2 notes .pdf	5m ago	e249e091 first file upload hexauser1049
README.md	6m ago	0a42aa67 Added README.md hexauser1049

Introduction

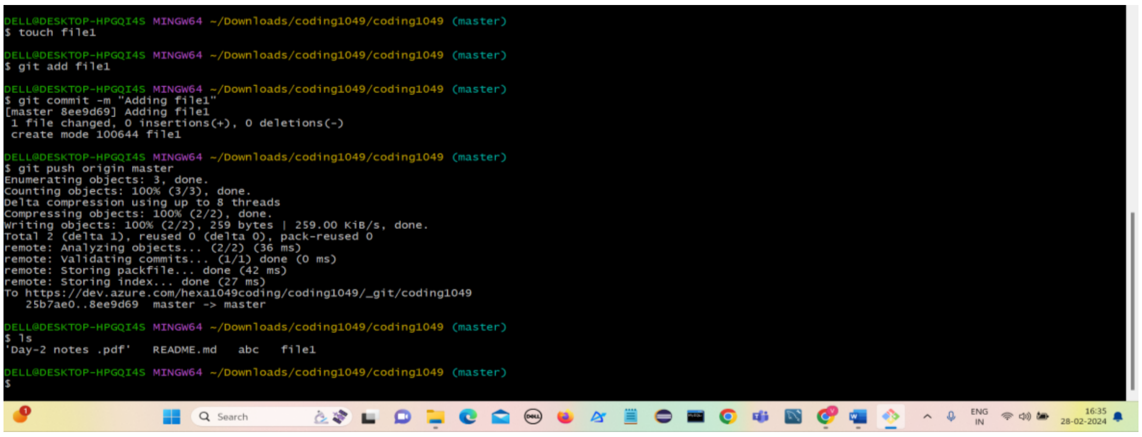
TODO: Give a short introduction of your project. Let this section explain the objectives or the motivation behind this project.

Getting Started

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

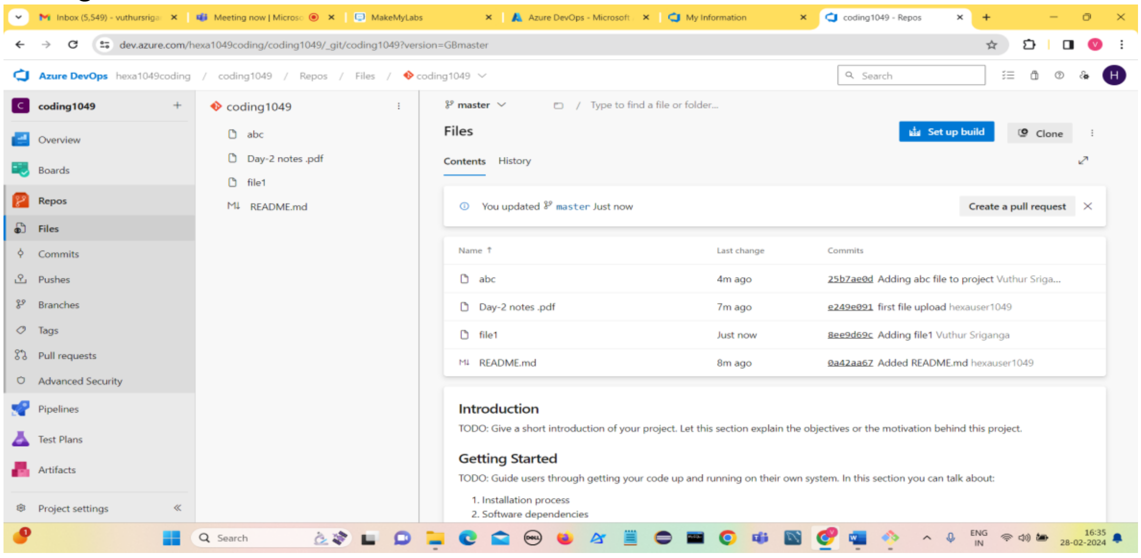
1. Installation process
2. Software dependencies
3. Latest releases
4. API references

Adding one more file file1



```
DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ touch file1

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git add file1

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git commit -m "Adding file1"
[master 8ee9d69] Adding file1
 1 file changed, 0 insertions(+), 0 deletions(-)
 create mode 100644 file1

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ git push origin master
Enumerating objects: 3, done.
Counting objects: 100% (3/3), done.
Delta compression using up to 8 threads
Compressing objects: 100% (2/2), done.
Writing objects: 100% (2/2), 259 bytes | 259.00 KiB/s, done.
Total 2 (delta 1), reused 0 (delta 0), pack-reused 0
remote: Analyzing objects... (2/2) (36 ms)
remote: Validating commits... (1/1) done (0 ms)
remote: Storing packfile... done (42 ms)
remote: Storing index... done (27 ms)
To https://dev.azure.com/hexa1049coding/coding1049/_git/coding1049
   25b7ae0..8ee9d69  master -> master

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$ ls
'Day-2 notes .pdf'   README.md    abc    file1

DELL@DESKTOP-HPGQI4S MINGW64 ~/Downloads/coding1049/coding1049 (master)
$
```

File1 got added



dev.azure.com/hexa1049coding/coding1049/_git/coding1049?version=GBmaster

Azure DevOps / hexa1049coding / coding1049 / Repos / Files / coding1049

coding1049: abc, Day-2 notes .pdf, file1, README.md

master / Type to find a file or folder...

Files — Contents | History

You updated master Just now — Create a pull request

Name	Last change	Commits
abc	4m ago	25b7ae0d Adding abc file to project Vuthur Sriga...
Day-2 notes .pdf	7m ago	e249e091 first file upload hexauser1049
file1	Just now	8ee9d69c Adding file1 Vuthur Sriganga
README.md	8m ago	0a42aa67 Added README.md hexauser1049

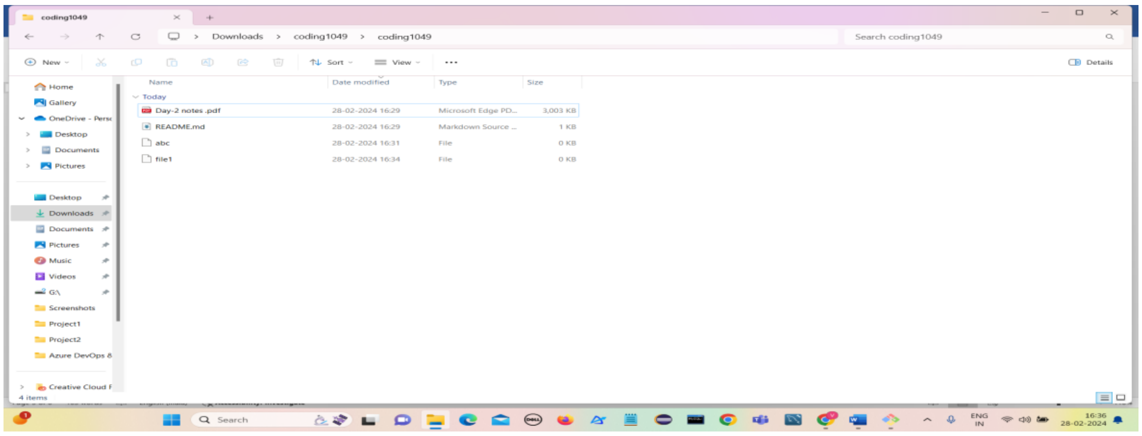
Introduction

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:

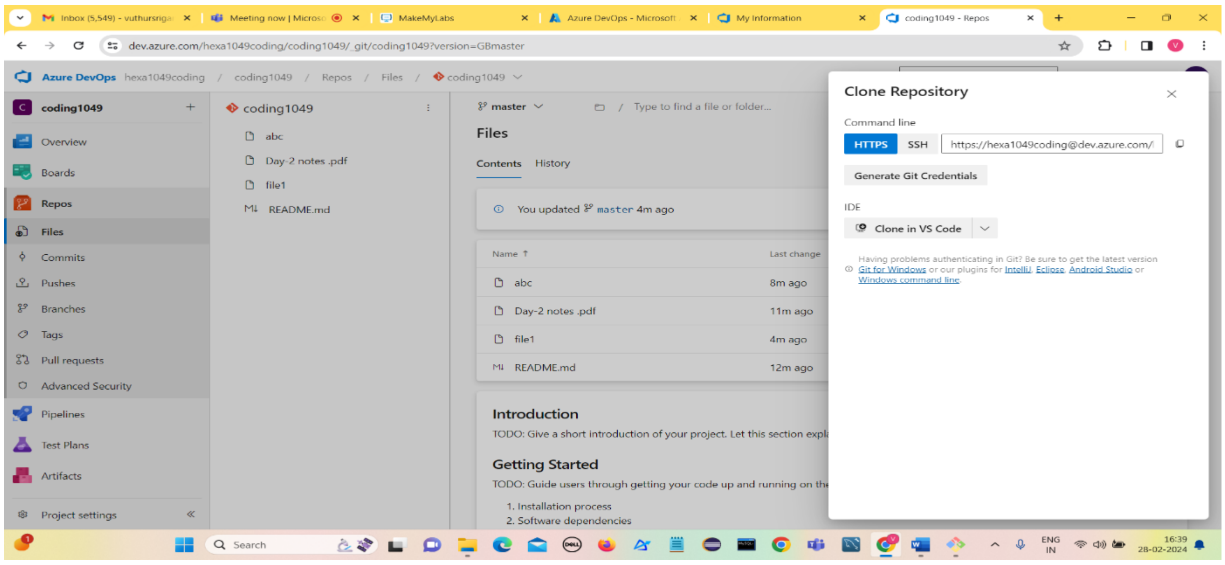
1. Installation process
2. Software dependencies



Downloads > coding1049 > coding1049

Name	Date modified	Type	Size
Day-2 notes .pdf	28-02-2024 16:29	Microsoft Edge PD...	3,003 KB
README.md	28-02-2024 16:29	Markdown Source ...	1 KB
abc	28-02-2024 16:31	File	0 KB
file1	28-02-2024 16:34	File	0 KB

Cloning using VS code

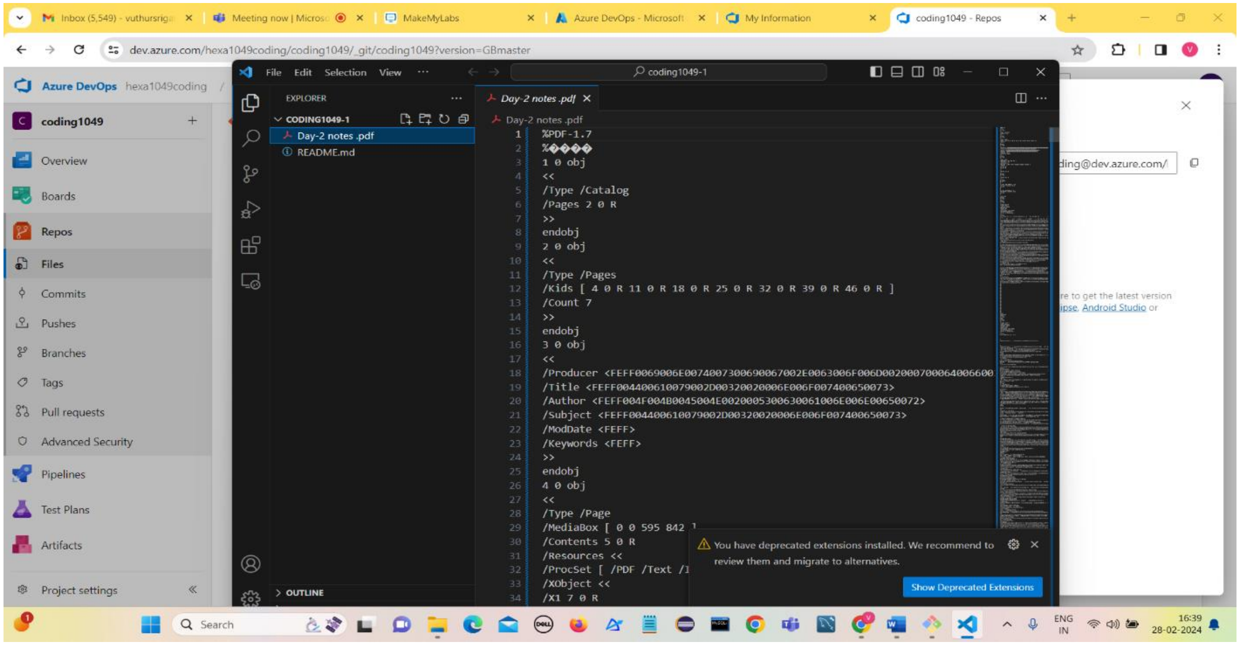


TODO: Give a short introduction of your project. Let this section expl

TODO: Guide users through getting your code up and running on the

1. Installation process
2. Software dependencies

Having problems authenticating in Git? Be sure to get the latest version Git for Windows or our plugins for IntelliJ, Eclipse, Android Studio or Windows command line


dev.azure.com/hexa1049coding/coding1049/_git/coding1049?version=GBmaster
File
Edit
Selection
View
coding1049-1
EXPLORER
CODING1049-1
Day-2 notes .pdf
README.md
OUTLINE
Show Deprecated Extensions

```
%PDF-1.7
%����
1 0 obj
<<
/Type /Catalog
/Pages 2 0 R
>>
endobj
2 0 obj
<<
/Type /Pages
/Kids [ 4 0 R 11 0 R 18 0 R 25 0 R 32 0 R 39 0 R 46 0 R ]
/Count 7
>>
endobj
3 0 obj
<<
/Producer <FEFF0069006E0074007300690067002E0063006F006D0020007000640066600
/Title <FEFF004400610079002D00320020006E006F00740065 0073>
/Author <FEFF004F004B0045004E002000530063 0061006E006E00650072>
/Subject <FEFF004400610079002D00320020006E006F00740065 0073>
/ModDate <FEFF>
/Keywords <FEFF>
>>
endobj
4 0 obj
<<
/Type /Page
/MediaBox [ 0 0 595 842 ]
/Contents 5 0 R
/Resources <<
/ProcSet [ /PDF /Text /I
/XObject <<
/X1 7 0 R
```

You have deprecated extensions installed. We recommend to review them and migrate to alternatives.

git commands used:

git init
touch filename
git add (filename) or git add .
git commit -m "samplecommitmsg"
git push origin main/master (local to azure repos)
git pull origin main/master (azure repos to local)
git push -u origin --all
git checkout main

2) Leverage the practices of CI/CD Using Azure Data Engineering

Leverage the practices of CI/CD Using Azure Data Engineering:

- CI stands for Continuous Integration and CD stands for Continuous Deployment.
- CI/CD in deployment of data pipeline mainly focuses on automating data operations and transformations.
- Using CI/CD for data pipeline automation has become more critical in ensuring the development velocity of processes.
- They include training machine learning models, supporting a data science team, doing large-scale data analysis, business intelligence or data visualization, supporting the growth of unstructured data collection, and other business needs.

Example:

We are building a toy train track. Every time we add a new piece, we want to ensure it fits perfectly and train doesnot come out from track.

Continuous Integration: Every time we add a new track piece, we immediately test it by running the toy train through it. This ensures that adding new piece has not caused any problems. If there's an issue, we can fix it immediately before it becomes a bigger problem.

Continuous Deployment: Once it is confirmed that our new piece fits and the train runs smoothly, we will show it everyone stating it is working perfectly. In other words, as soon as your changes are verified, they're made live and functional in the main track (production environment).

Continuous Integration:
CI checks and tests every new piece of code (or data transformation logic) you add to your data pipeline.

Continuous Deployment:

CD ensures that once tested and approved, this code gets added to the live system without manual intervention.

Uses of Continuous Integration:

1) Automated Testing
2) Version Control
3) Consistent Environment
4) Data Quality Checks

Uses of Continuous Deployment:

1) Automated Deployment
2) Monitoring and Alerts
3) Rollbacks
4) Infrastructure as Code (IaC)

3) Explain the architecture of the Azure Synpase .

- Azure Synapse Analytics is designed to streamline the process of ingesting, preparing, managing, and analyzing data at scale, empowering organizations to derive valuable insights and make data-driven decisions more efficiently.
- It is a cloud-based analytics service provided by Microsoft Azure.
- It integrates big data and data warehousing functionalities, allowing users to analyze and visualize large volumes of data for business intelligence and decision-making purposes.
- It combines enterprise data warehousing, big data analytics, and data integration into a single service, providing a unified experience for data engineers, data scientists, and business analysts.

Components of Azure Synapse Architecture:

1) Synapse SQL

Azure Synapse SQL is a distributed SQL engine designed for large-scale data processing. It is based on Microsoft SQL Server and can handle large data workloads. Azure Synapse SQL allows users to query data stored in various sources, including data lakes, data warehouses, and operational databases, using standard SQL syntax. It also supports both serverless and dedicated options for processing queries, depending on the user's needs.

i) Dedicated SQL Pool

- The aim of utilizing Dedicated SQL Pools is to effectively store a large amount of data and able to perform queries efficiently.
- It stores the data in a columnar format for easy retrieval.
- Its performance is determined by the Data Warehouse Units (DWUs) we choose while creating the resource.
- One of the benefits is that we can temporarily halt its resources to save on costs and start it again whenever required.
- However, you should note that storage costs still apply even when the Dedicated SQL Pool is paused.

ii) Serverless SQL Pool

- The Serverless SQL Pool is a tool for querying data in your data lake.
- We can use it to access your data with T-SQL syntax, which lets us to query the data without needing to copy or load it into a specific storage location.
- We can also connect to the tool through the T-SQL interface and use various business intelligence and ad-hoc querying tools, including commonly used drivers.

2) Spark

- Spark in Azure Synapse Analytics is fully managed and comes with pre-installed libraries for Python, R, and Scala.
- The Spark component in Azure Synapse Analytics is highly scalable, enabling users to process large amounts of data quickly and efficiently.
- It also integrates with other Azure services such as Azure Data Factory, Azure Stream Analytics, and Azure Event Hubs.

3) Synapse Pipelines

- Azure Synapse Pipelines is a cloud-based data integration service that allows users to create, schedule, and manage data pipelines.
- It is part of the Azure Synapse Analytics workspace and provides an graphical interface for building and managing data pipelines.

- With Synapse Pipelines, users can connect to various data sources, transform and process data using a wide range of built-in transformations and custom code, and load the data into various destinations, including Azure Synapse Analytics, Azure Data Lake Storage, and Azure SQL Database.

4) Synapse Studio

- Azure Synapse Studio is a web-based integrated development environment (IDE) that provides a unified experience for data integration, data warehousing, big data, and AI tasks.
- It is a key component of the Azure Synapse Analytics architecture and provides a collaborative workspace for data engineers, data scientists, and business analysts.
- Synapse Studio allows users to create and manage multiple Synapse workspaces from a single location.

Vuthur Sriganga

vuthursriganga@gmail.com