LOYALIST COLLEGE

In-class Application Exercise 2

AISC2013 - Deployment of AI Solutions 02

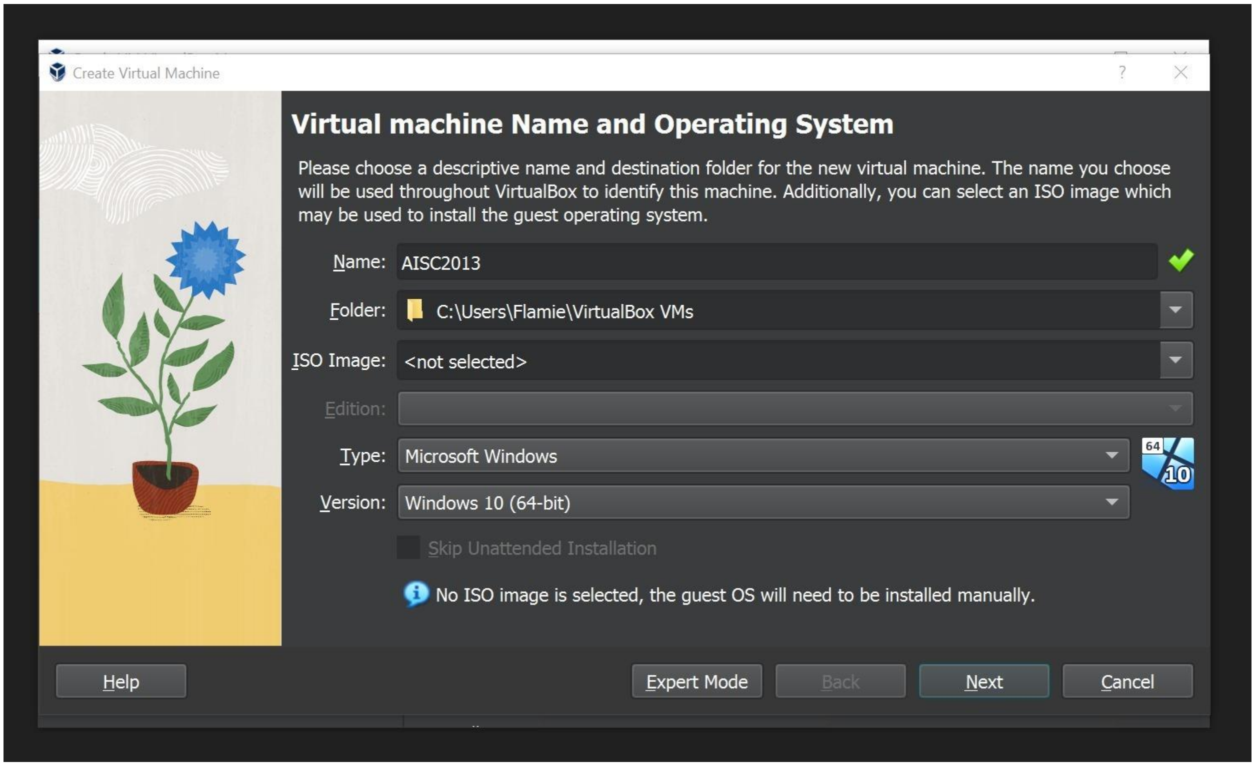
Submitted to: Qasim Ali
Submitted by: Group K

Name	ID	Contribution
Pooja Katrodiya	500221151	Performed Task 2
Kritika	500220642	Performed Task 1
Nency Amrutiya	500221156	Performed Task 3
Dipeshkumar Shah	500221438	Completed Task 4
Mir Bilal Mohsin	500208059	Carried out the first 3 initial steps for task 3 which was then taken over by nency since my computer was posing problems executing any further.

Task 1: Development Environment using VirtualBox

1. Create a Virtual Machine:

Open VirtualBox and create a new virtual machine. Specify the necessary details such as the VM name, type, and version.



Create Virtual Machine

Virtual machine Name and Operating System

Please choose a descriptive name and destination folder for the new virtual machine. The name you choose will be used throughout VirtualBox to identify this machine. Additionally, you can select an ISO image which may be used to install the guest operating system.

Name: AISC2013

Folder: C:\Users\Flamie\VirtualBox VMs

ISO Image: <not selected>

Edition:

Type: Microsoft Windows

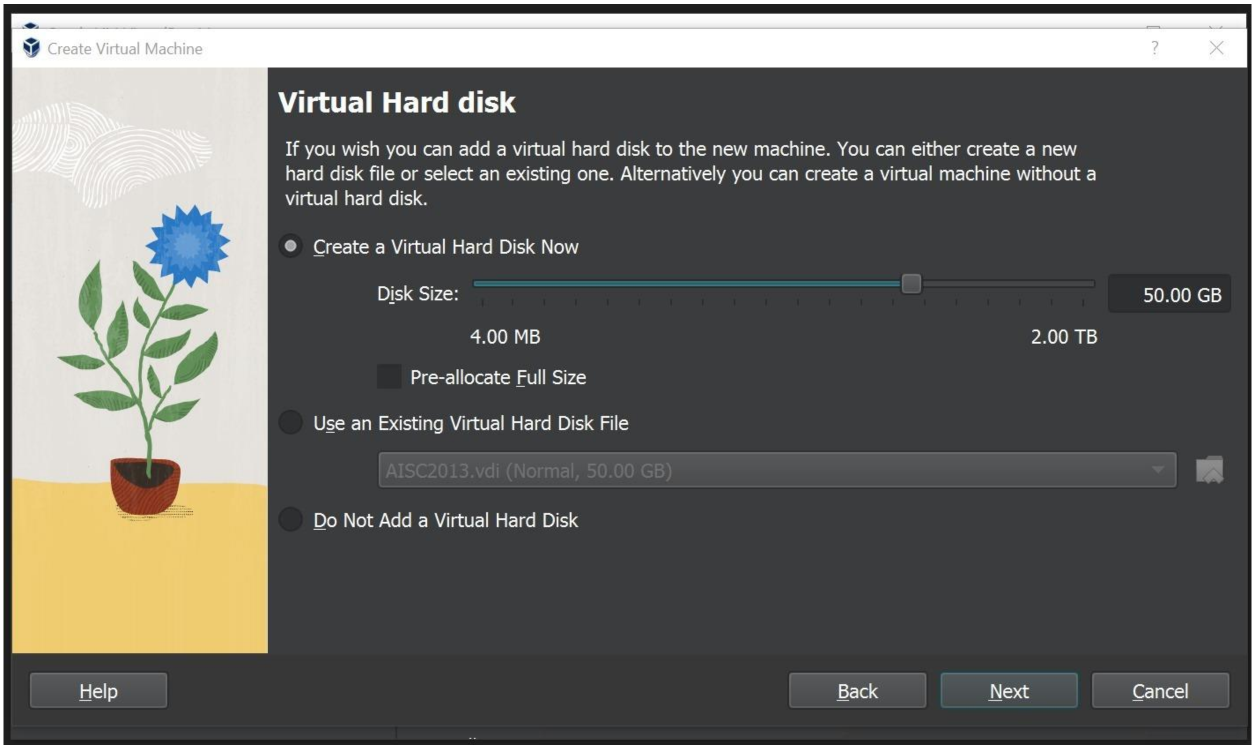
Version: Windows 10 (64-bit)

□ Skip Unattended Installation

No ISO image is selected, the guest OS will need to be installed manually.

2. Configure VM Settings:

Adjust VM settings such as memory, CPU, and network configurations according to your AI solution's requirements.



Create Virtual Machine

Virtual Hard disk

If you wish you can add a virtual hard disk to the new machine. You can either create a new hard disk file or select an existing one. Alternatively you can create a virtual machine without a virtual hard disk.

● Create a Virtual Hard Disk Now

Disk Size: 50.00 GB

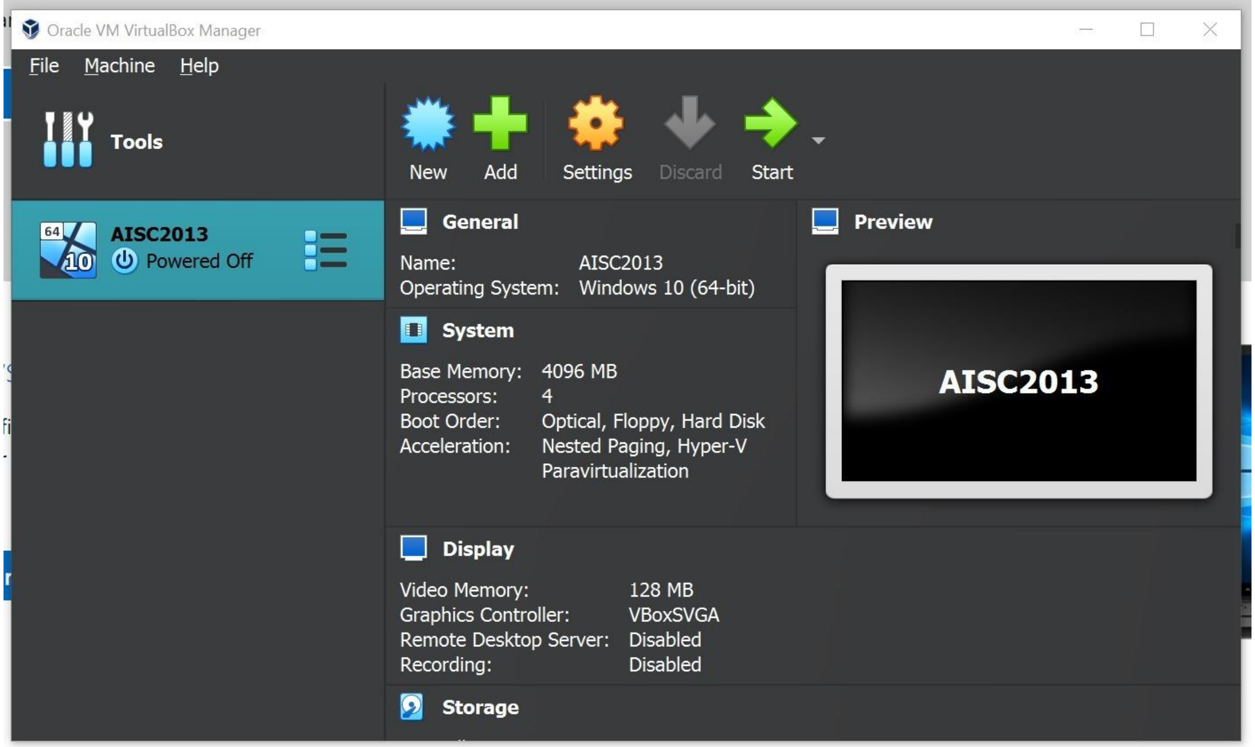
4.00 MB 2.00 TB

□ Pre-allocate Full Size

○ Use an Existing Virtual Hard Disk File

AISC2013.vdi (Normal, 50.00 GB)

○ Do Not Add a Virtual Hard Disk



Oracle VM VirtualBox Manager

AISC2013
Powered Off

General

Name: AISC2013
Operating System: Windows 10 (64-bit)

System

Base Memory: 4096 MB
Processors: 4
Boot Order: Optical, Floppy, Hard Disk
Acceleration: Nested Paging, Hyper-V Paravirtualization

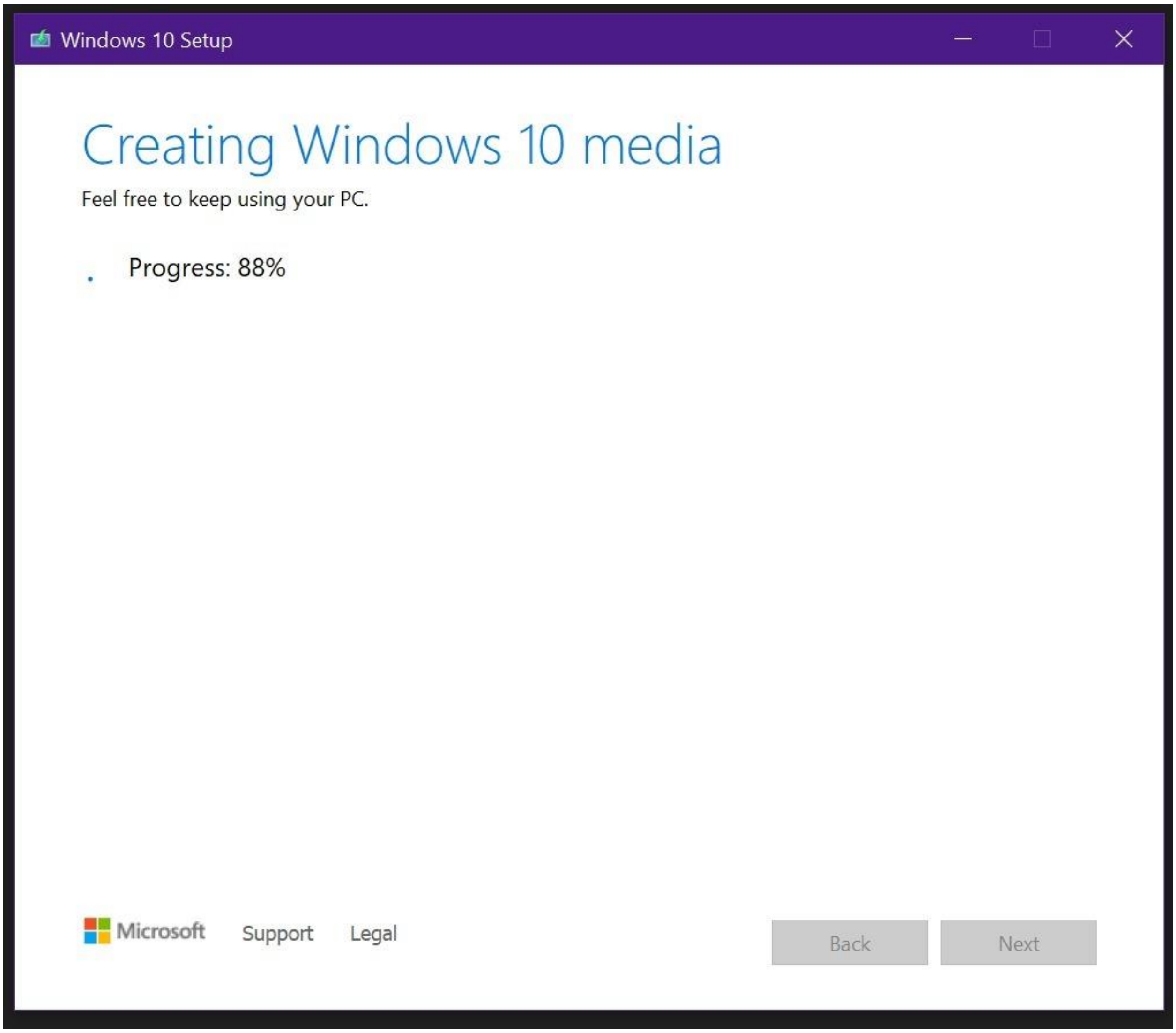
Preview

AISC2013

Display

Video Memory: 128 MB
Graphics Controller: VBoxSVGA
Remote Desktop Server: Disabled
Recording: Disabled

Storage



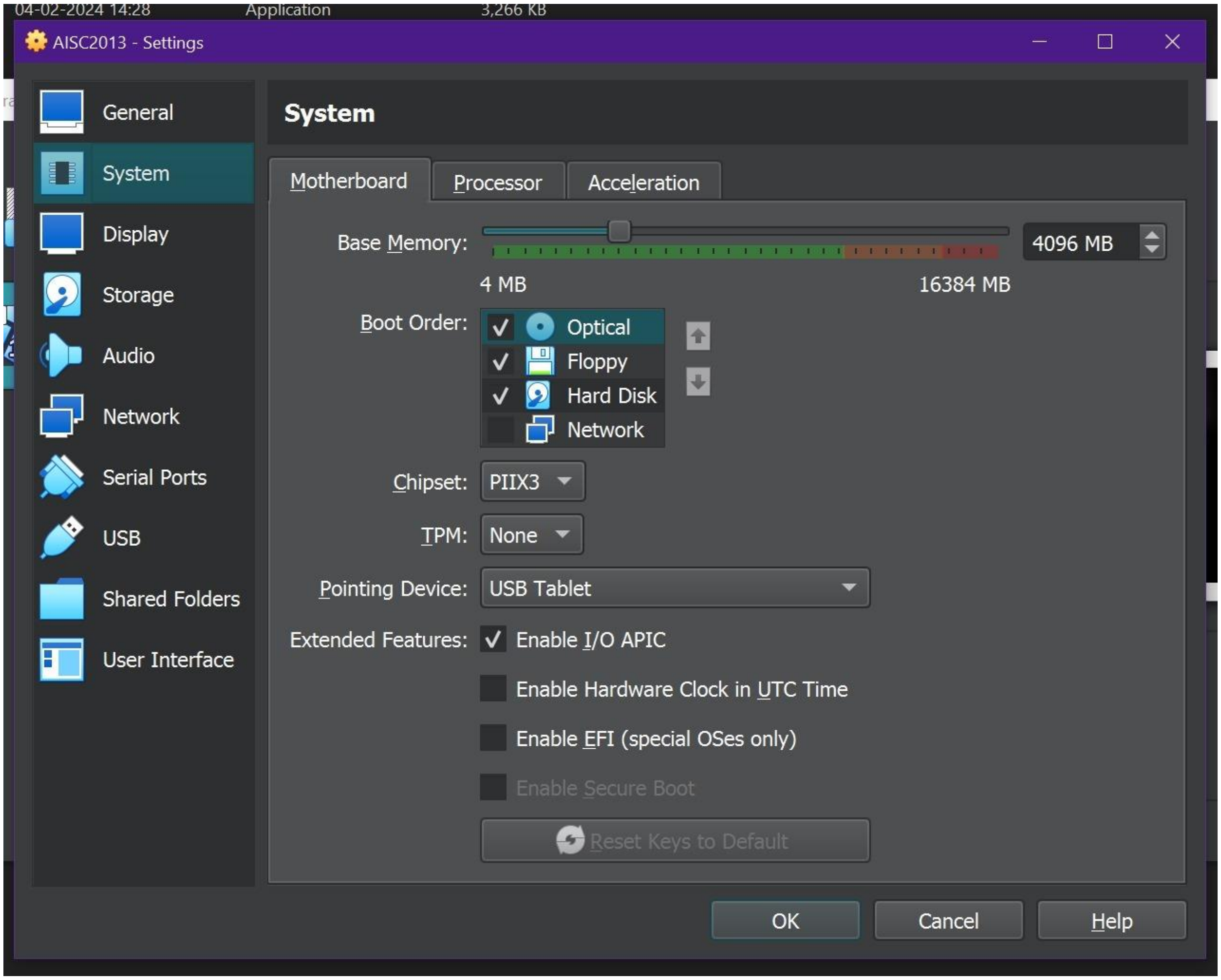
Creating Windows 10 media

Feel free to keep using your PC.

Progress: 88%

3. Attach ISO Image:

Attach the Windows 10 ISO image to the VM. Start the VM to initiate the Windows installation process.



Base Memory: 4096 MB

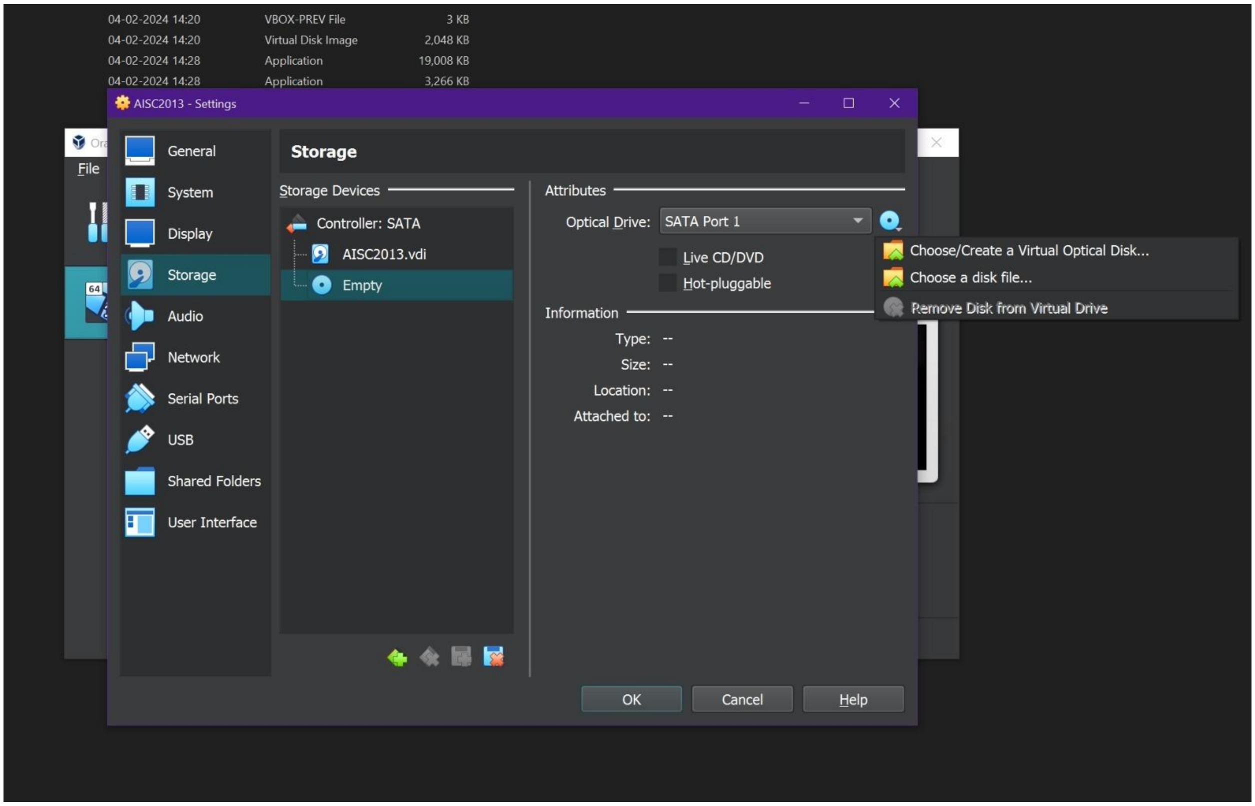
Boot Order:
- ☑ Optical
- ☑ Floppy
- ☑ Hard Disk
- □ Network

Chipset: PIIX3

TPM: None

Pointing Device: USB Tablet

Extended Features:
- ☑ Enable I/O APIC
- □ Enable Hardware Clock in UTC Time
- □ Enable EFI (special OSes only)
- □ Enable Secure Boot



```
04-02-2024 14:20    VBOX-PREV File         3 KB
04-02-2024 14:20    Virtual Disk Image     2,048 KB
04-02-2024 14:28    Application            19,008 KB
04-02-2024 14:28    Application            3,266 KB
```

AISC2013 - Settings

Storage

Storage Devices

- Controller: SATA
  - AISC2013.vdi
  - Empty

Attributes

Optical Drive: SATA Port 1

□ Live CD/DVD

□ Hot-pluggable

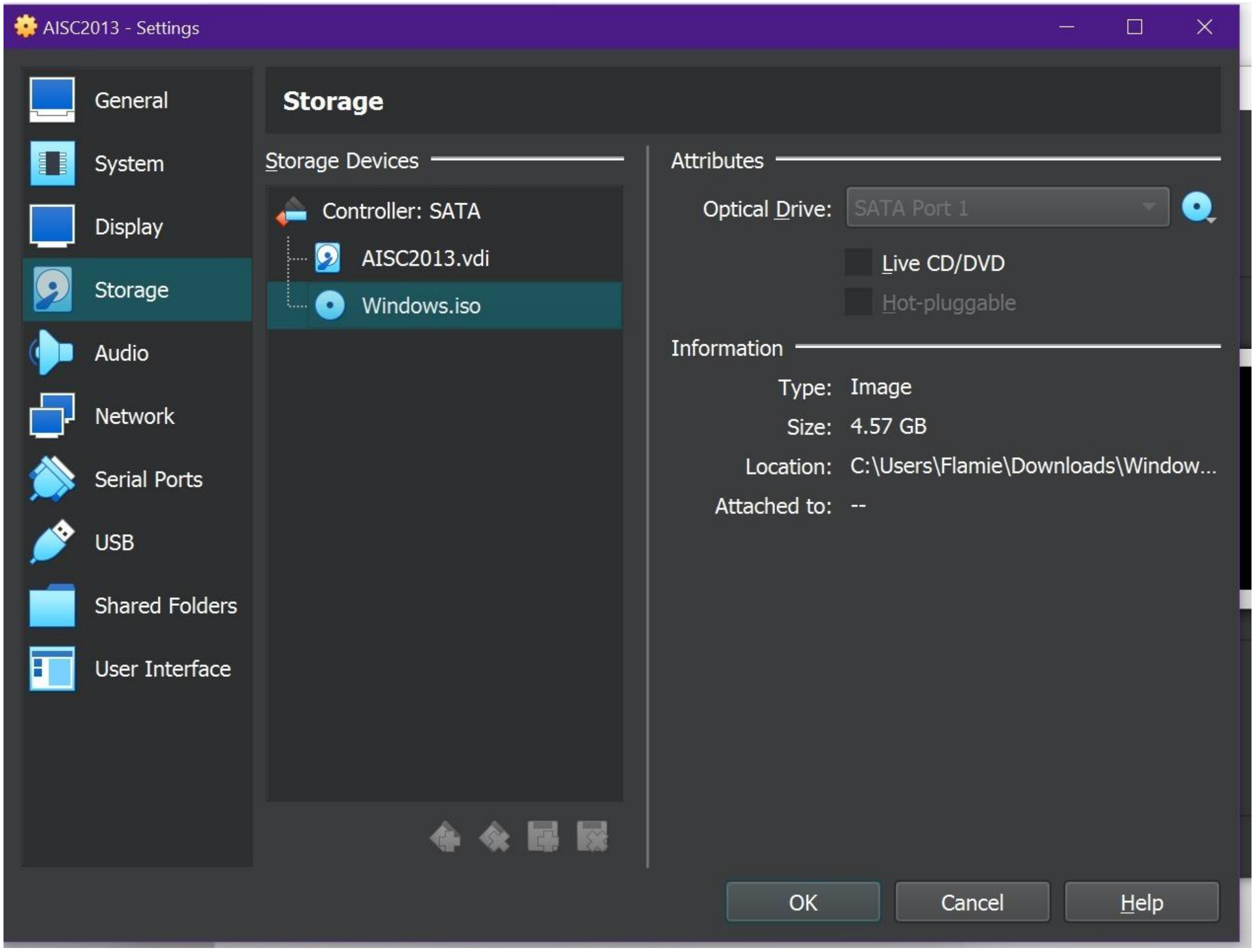
Information

Type: --

Size: --

Location: --

Attached to: --



Storage

Storage Devices

- Controller: SATA
  - AISC2013.vdi
  - Windows.iso

Attributes

Optical Drive: SATA Port 1

□ Live CD/DVD

□ Hot-pluggable

Information

Type: Image

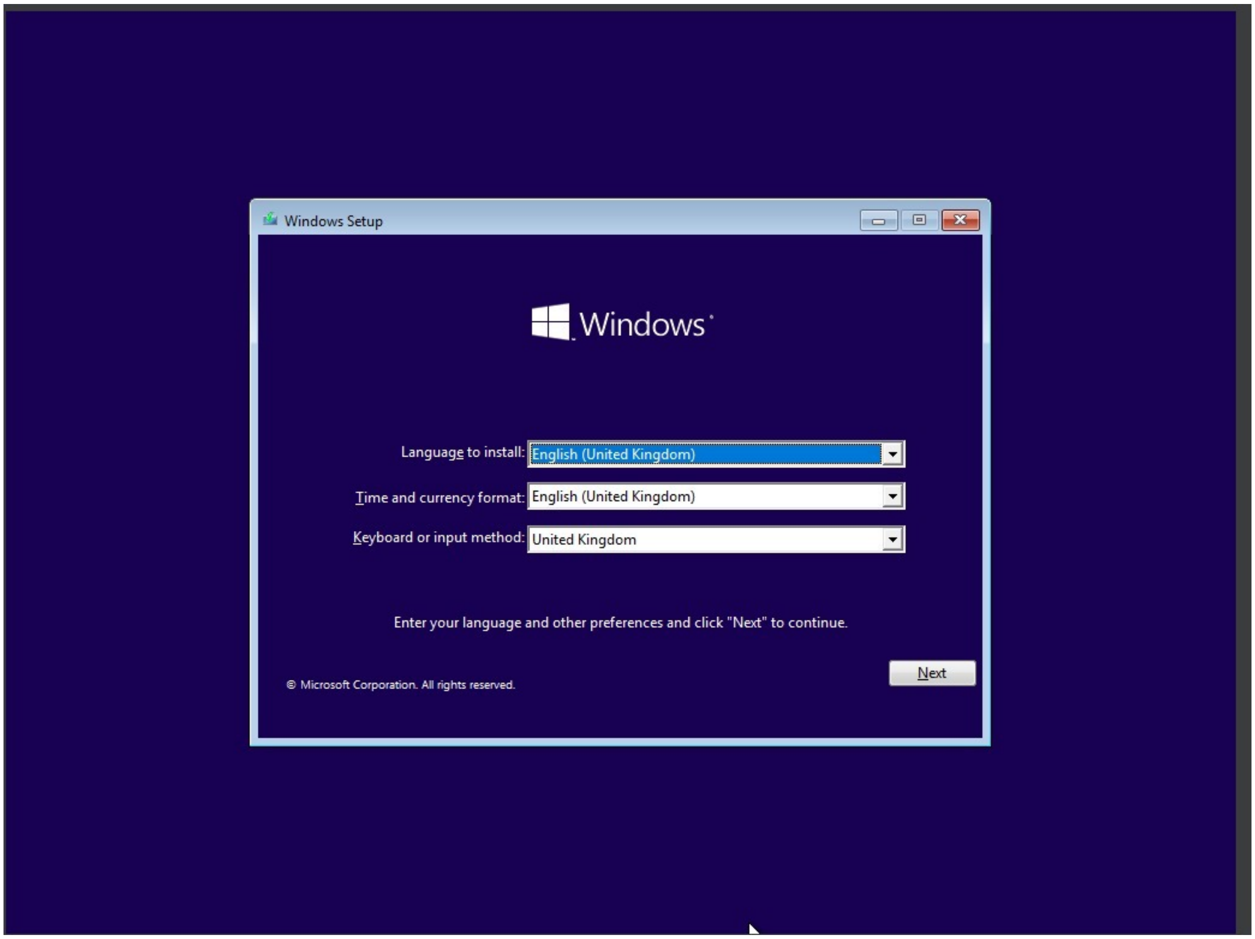
Size: 4.57 GB

Location: C:\Users\Flamie\Downloads\Window...

Attached to: --

4. Install Windows:

Follow the on-screen instructions to install Windows 10 on the virtual machine.

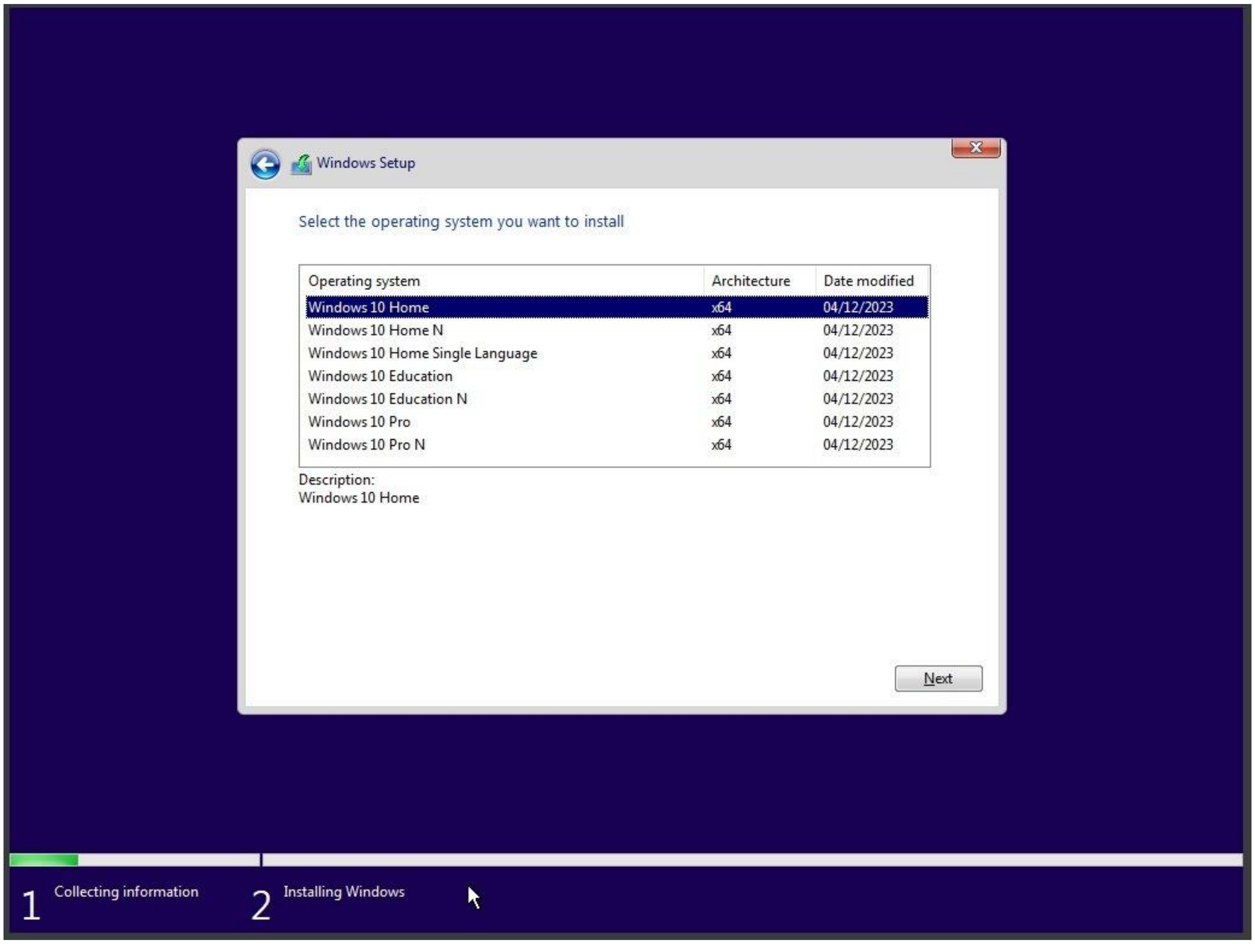


Language to install: English (United Kingdom)

Time and currency format: English (United Kingdom)

Keyboard or input method: United Kingdom

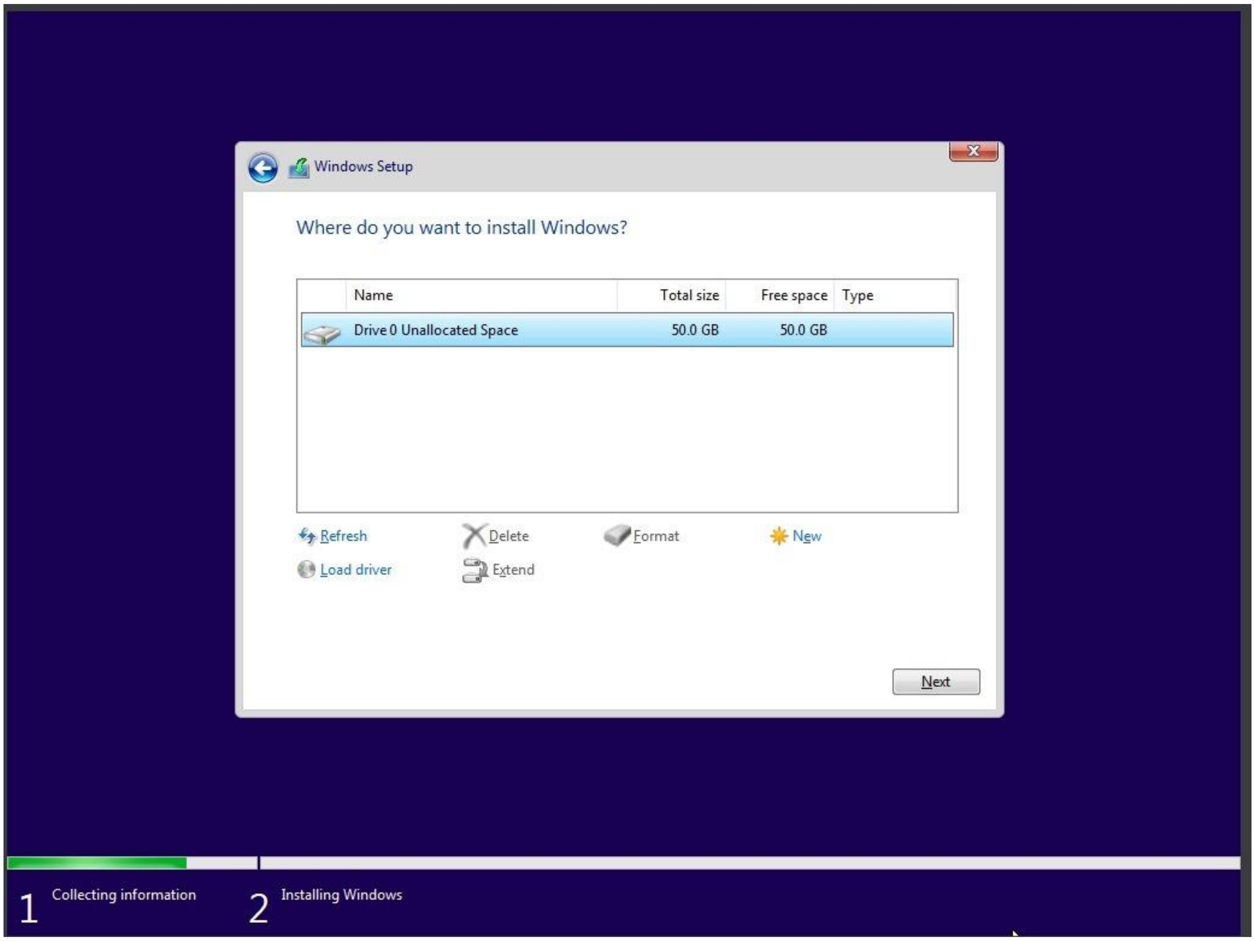
Enter your language and other preferences and click "Next" to continue.



Select the operating system you want to install

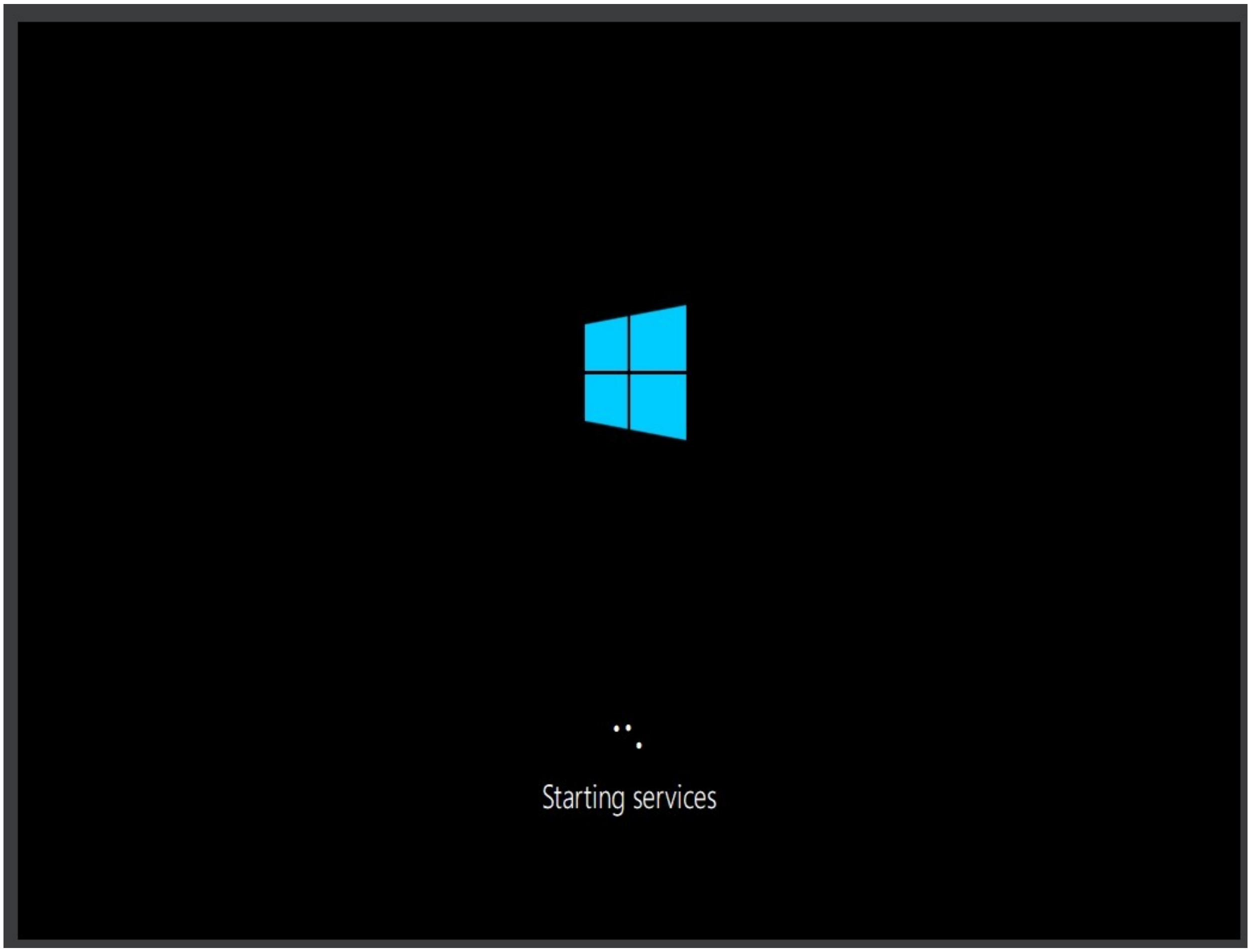
Operating system	Architecture	Date modified
Windows 10 Home	x64	04/12/2023
Windows 10 Home N	x64	04/12/2023
Windows 10 Home Single Language	x64	04/12/2023
Windows 10 Education	x64	04/12/2023
Windows 10 Education N	x64	04/12/2023
Windows 10 Pro	x64	04/12/2023
Windows 10 Pro N	x64	04/12/2023

Description:
Windows 10 Home

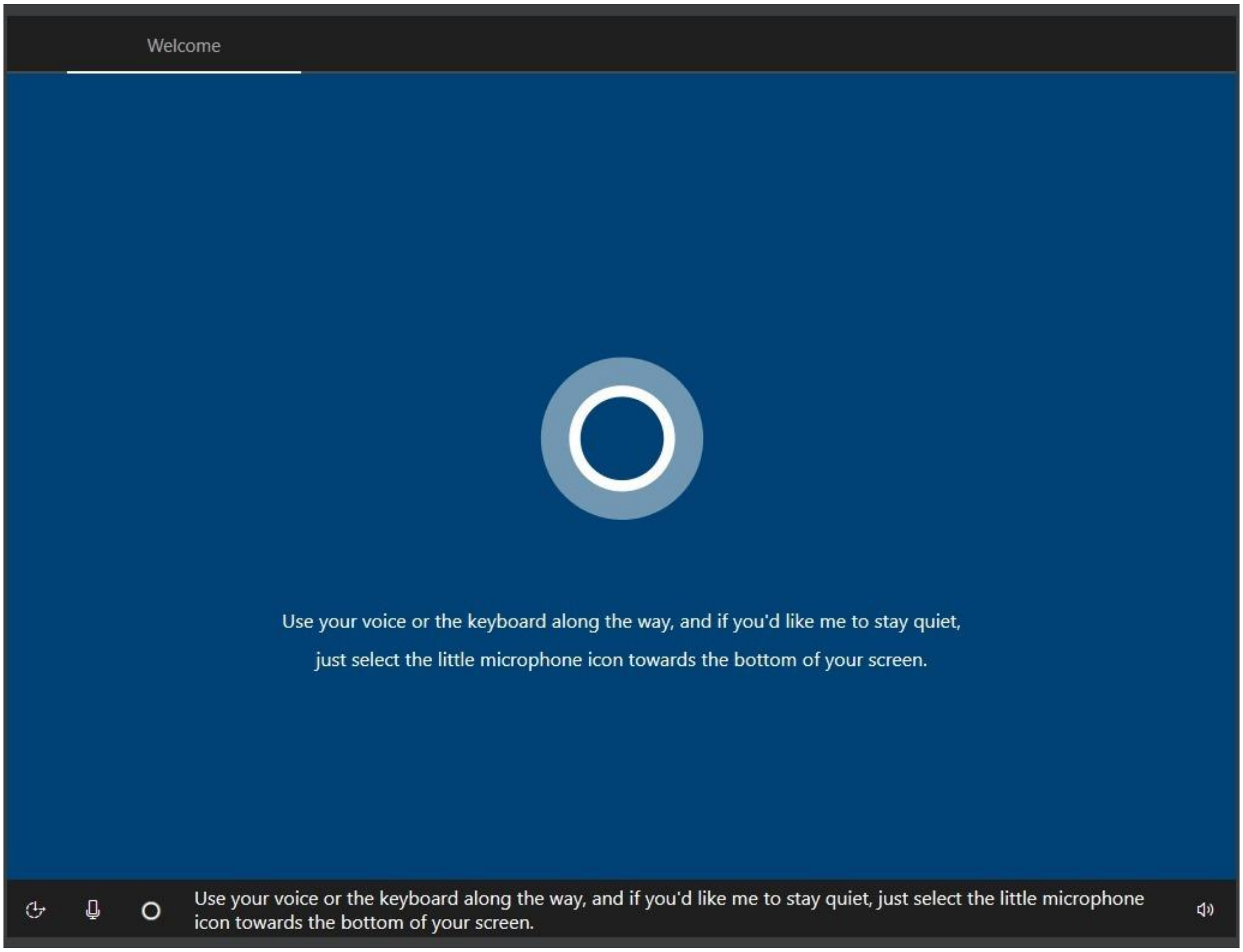


Where do you want to install Windows?

	Name	Total size	Free space	Type
	Drive 0 Unallocated Space	50.0 GB	50.0 GB	

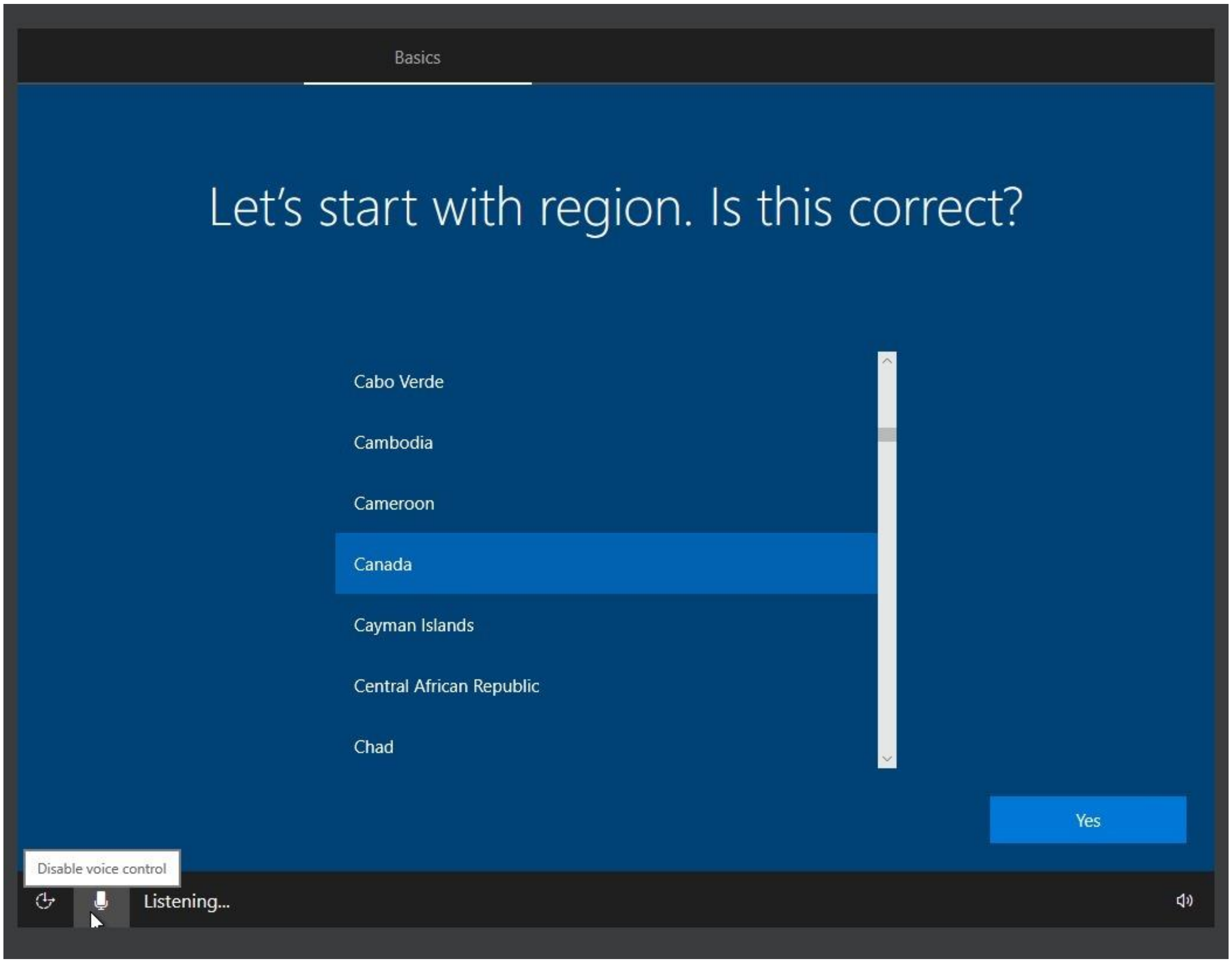


Starting services



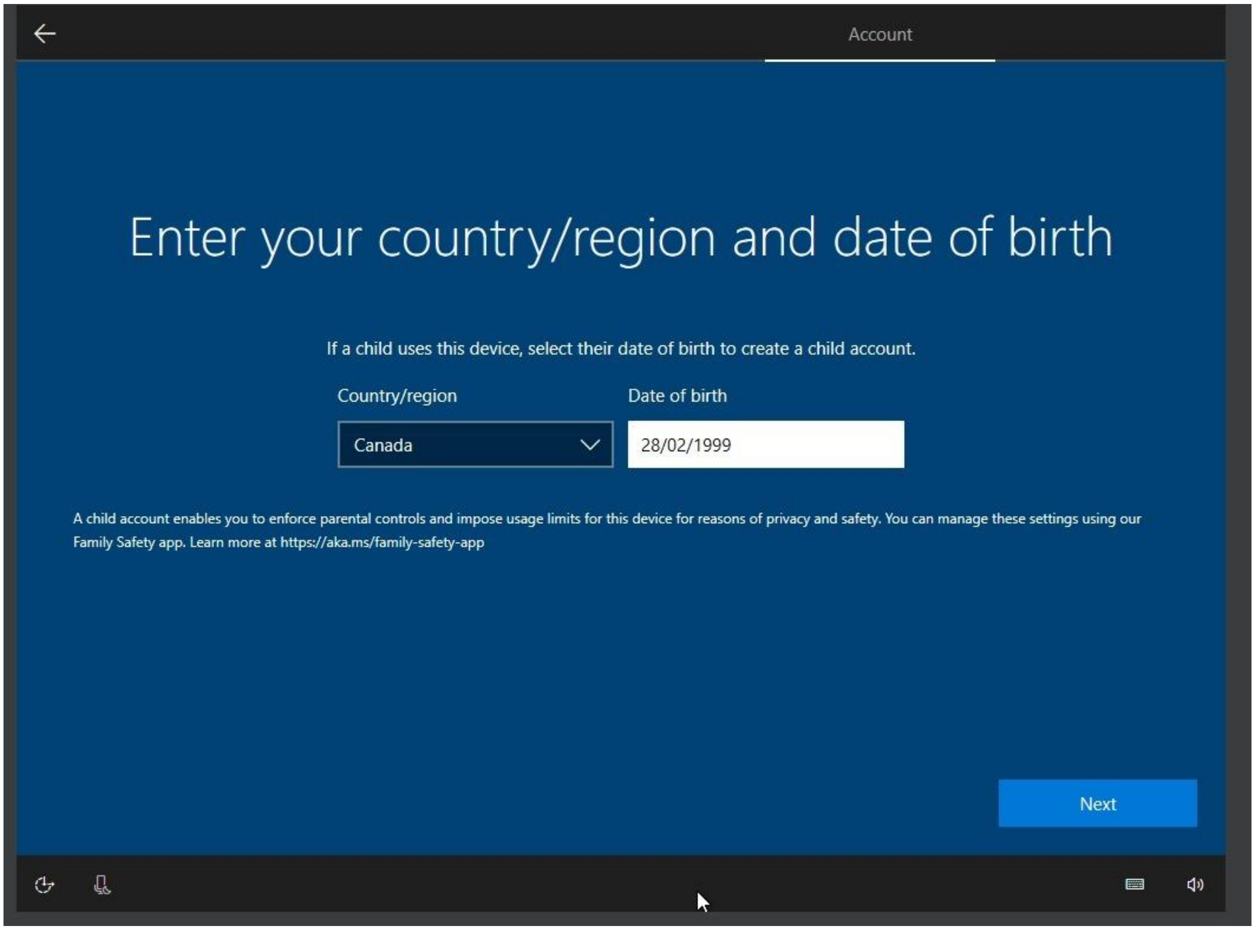
Use your voice or the keyboard along the way, and if you'd like me to stay quiet, just select the little microphone icon towards the bottom of your screen.

Use your voice or the keyboard along the way, and if you'd like me to stay quiet, just select the little microphone icon towards the bottom of your screen.



Let’s start with region. Is this correct?

Canada



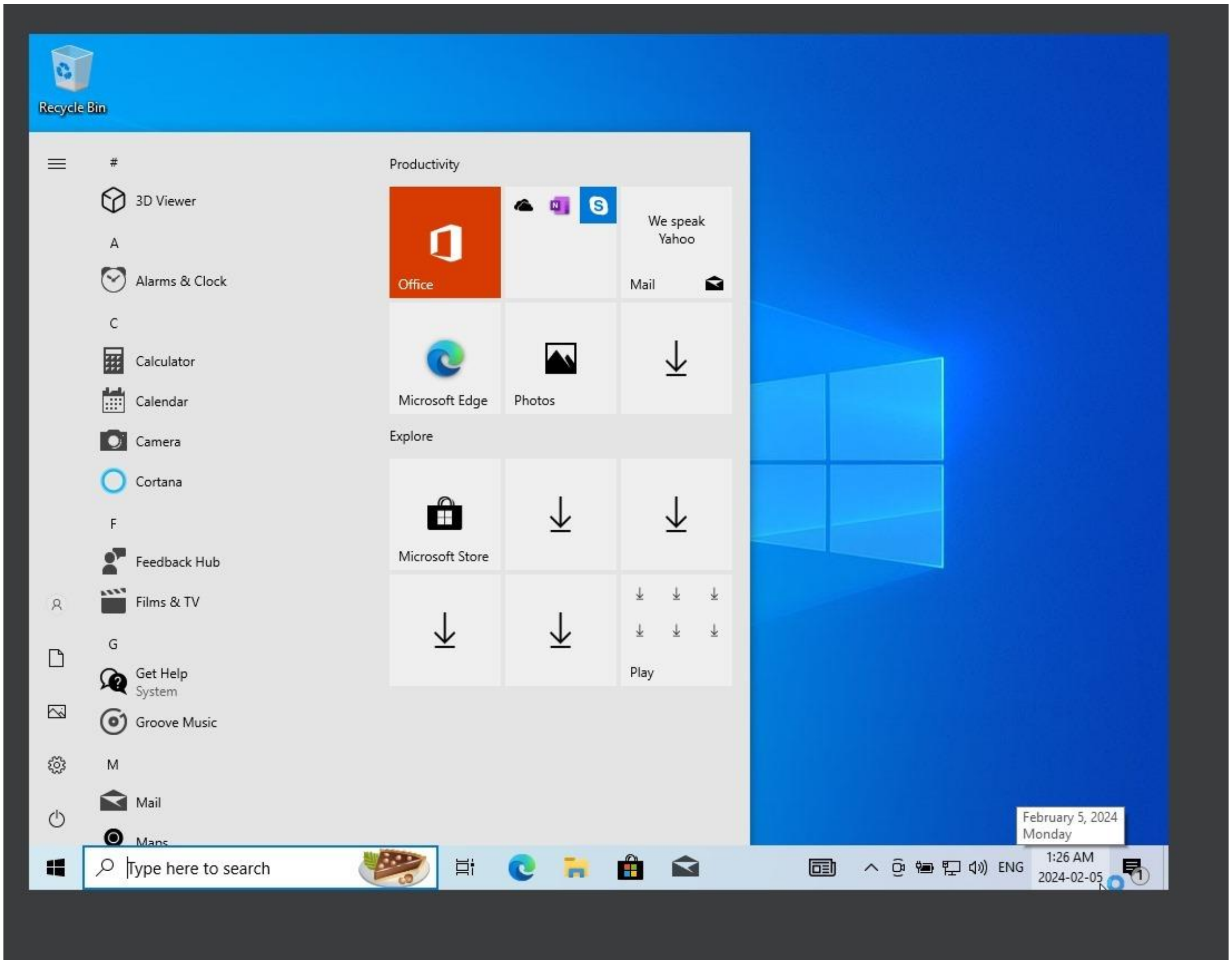
Enter your country/region and date of birth

If a child uses this device, select their date of birth to create a child account.

Country/region: Canada

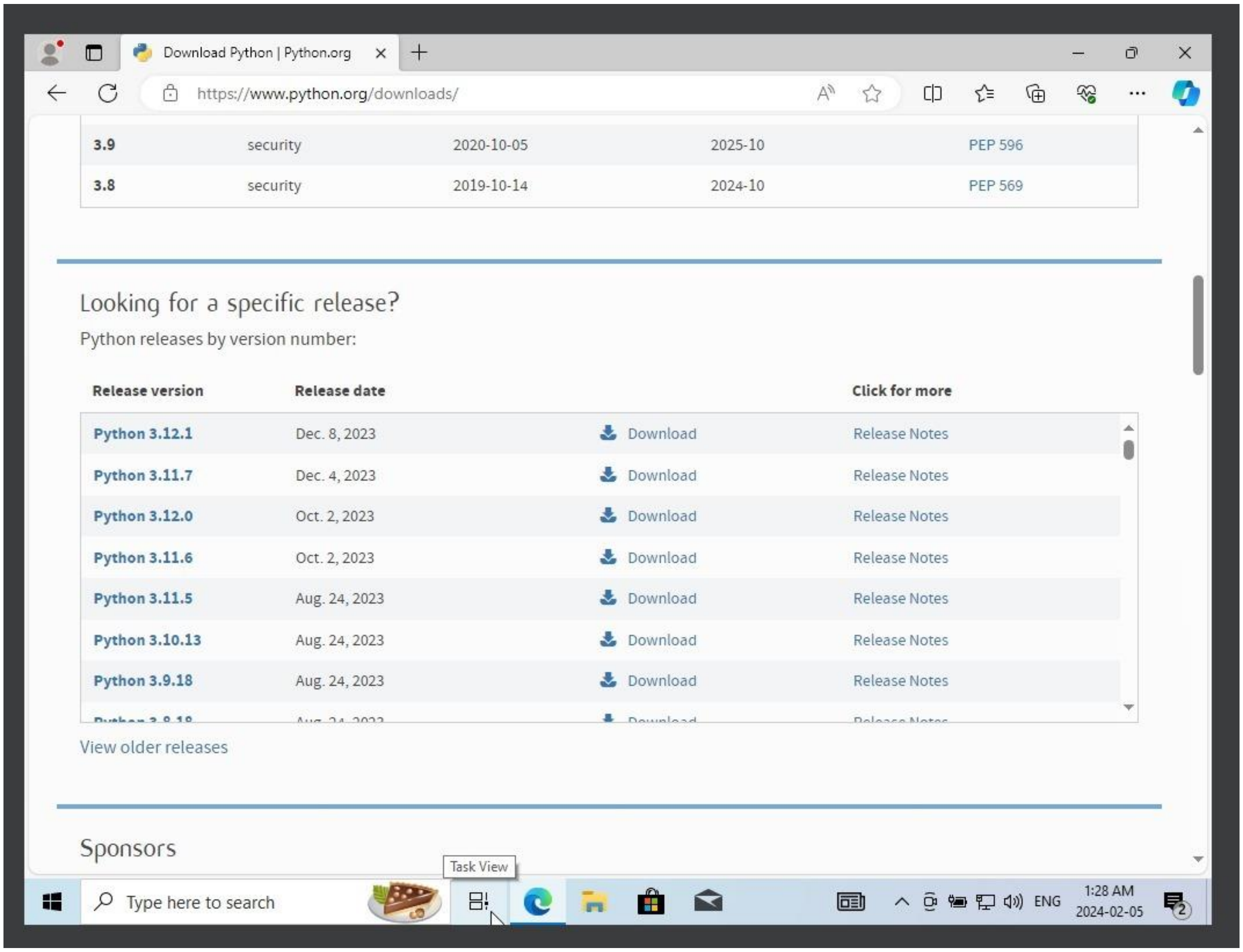
Date of birth: 28/02/1999

A child account enables you to enforce parental controls and impose usage limits for this device for reasons of privacy and safety. You can manage these settings using our Family Safety app. Learn more at https://aka.ms/family-safety-app


Recycle Bin
#
3D Viewer
A
Alarms & Clock
C
Calculator
Calendar
Camera
Cortana
F
Feedback Hub
Films & TV
G
Get Help
System
Groove Music
M
Mail
Maps
Productivity
Office
We speak Yahoo
Mail
Microsoft Edge
Photos
Explore
Microsoft Store
Play
Type here to search
ENG
1:26 AM
2024-02-05
February 5, 2024
Monday

5. Install Python and Dependencies:

After Windows installation, install Python and any necessary dependencies for your AI solution.


Download Python | Python.org
https://www.python.org/downloads/
Task View
Type here to search
ENG
1:28 AM
2024-02-05

3.9	security	2020-10-05	2025-10	PEP 596
3.8	security	2019-10-14	2024-10	PEP 569

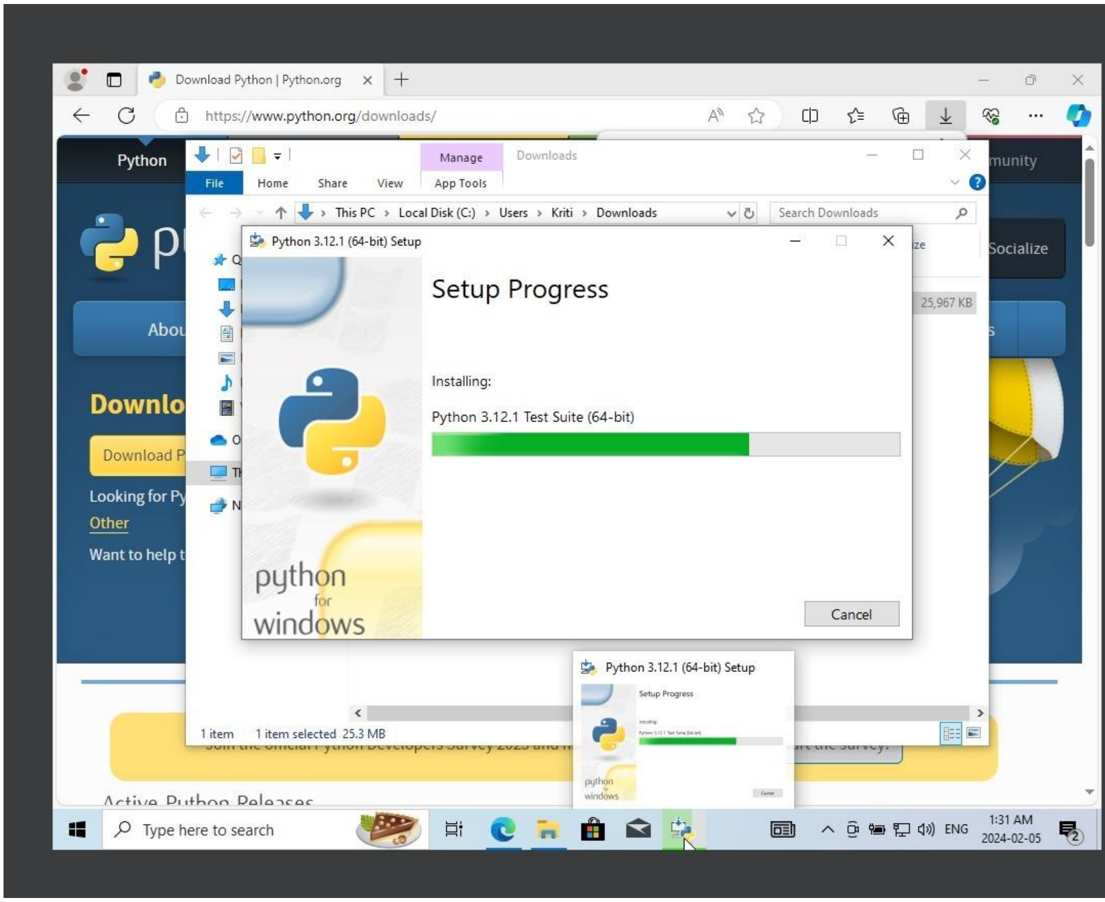
Looking for a specific release?

Python releases by version number:

Release version	Release date		Click for more
Python 3.12.1	Dec. 8, 2023	Download	Release Notes
Python 3.11.7	Dec. 4, 2023	Download	Release Notes
Python 3.12.0	Oct. 2, 2023	Download	Release Notes
Python 3.11.6	Oct. 2, 2023	Download	Release Notes
Python 3.11.5	Aug. 24, 2023	Download	Release Notes
Python 3.10.13	Aug. 24, 2023	Download	Release Notes
Python 3.9.18	Aug. 24, 2023	Download	Release Notes
Python 3.8.18	Aug. 24, 2023	Download	Release Notes

View older releases

Sponsors



Python 3.12.1 (64-bit) Setup

Setup Progress

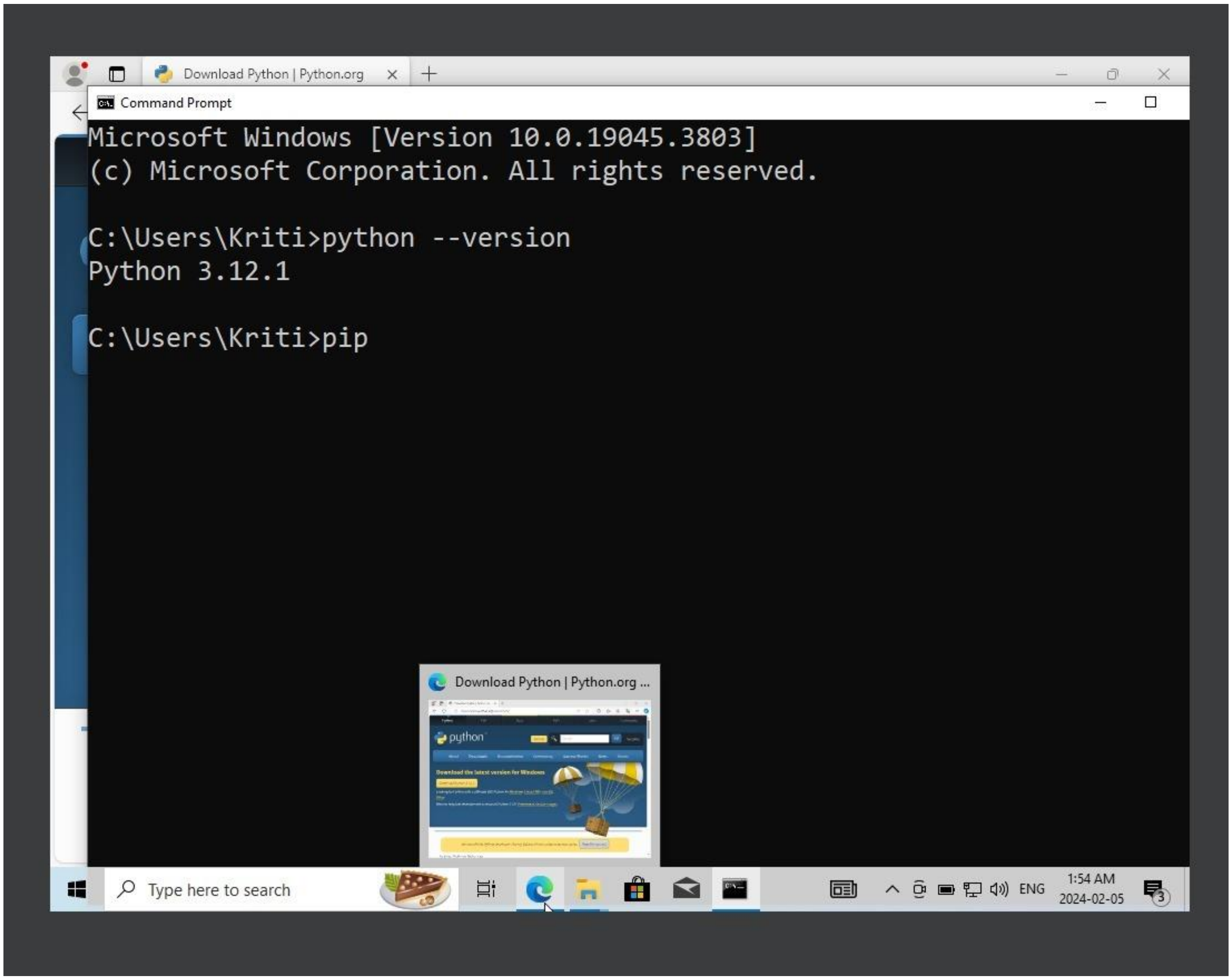
Installing:

Python 3.12.1 Test Suite (64-bit)

Download Python | Python.org
https://www.python.org/downloads/
Manage
Downloads
File
Home
Share
View
App Tools
This PC › Local Disk (C:) › Users › Kriti › Downloads
Search Downloads
Cancel
1 item 1 item selected 25.3 MB
25,967 KB
Type here to search
1:31 AM
2024-02-05
ENG

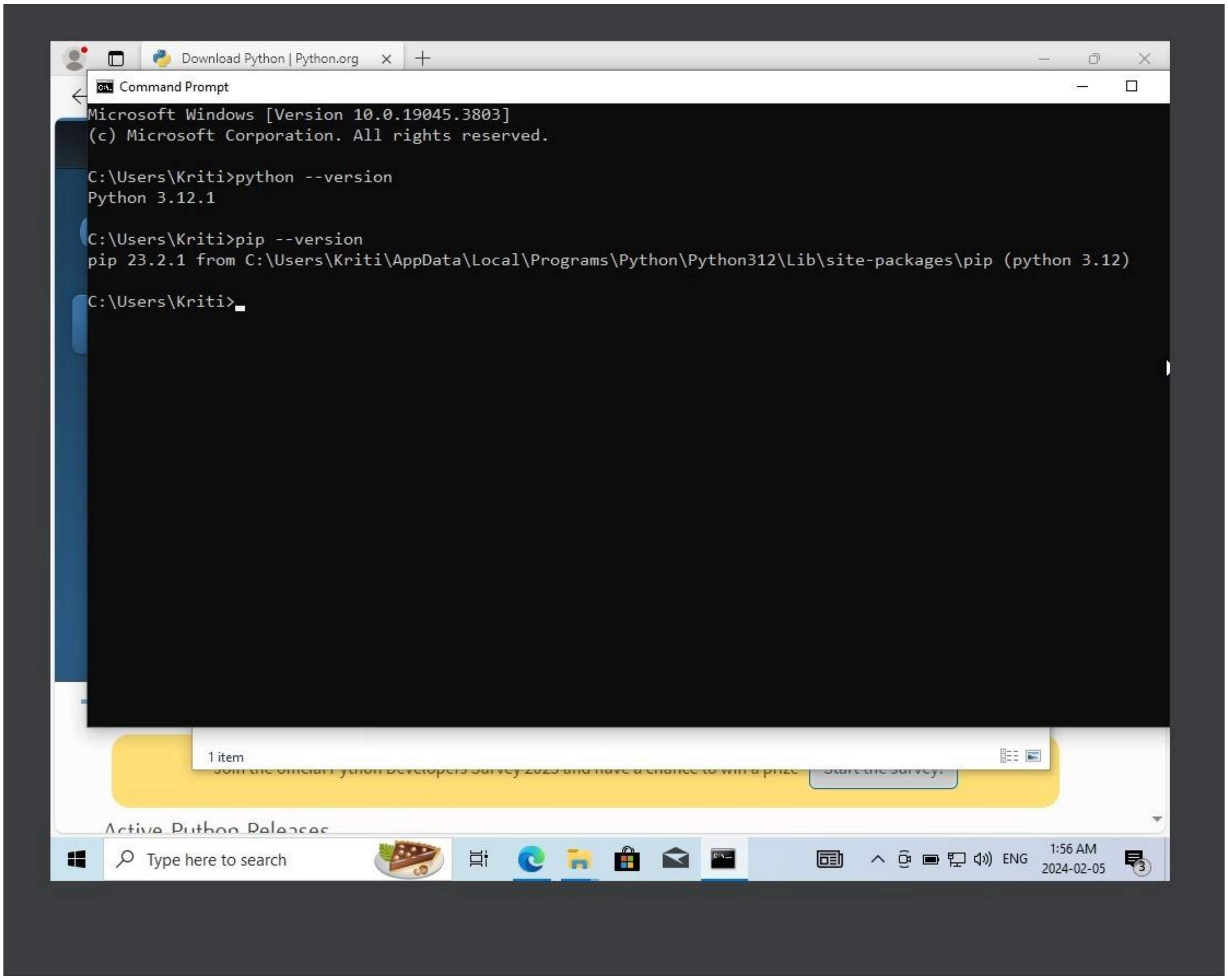
6. Verify Environment:

Verify that Python and dependencies are installed successfully.



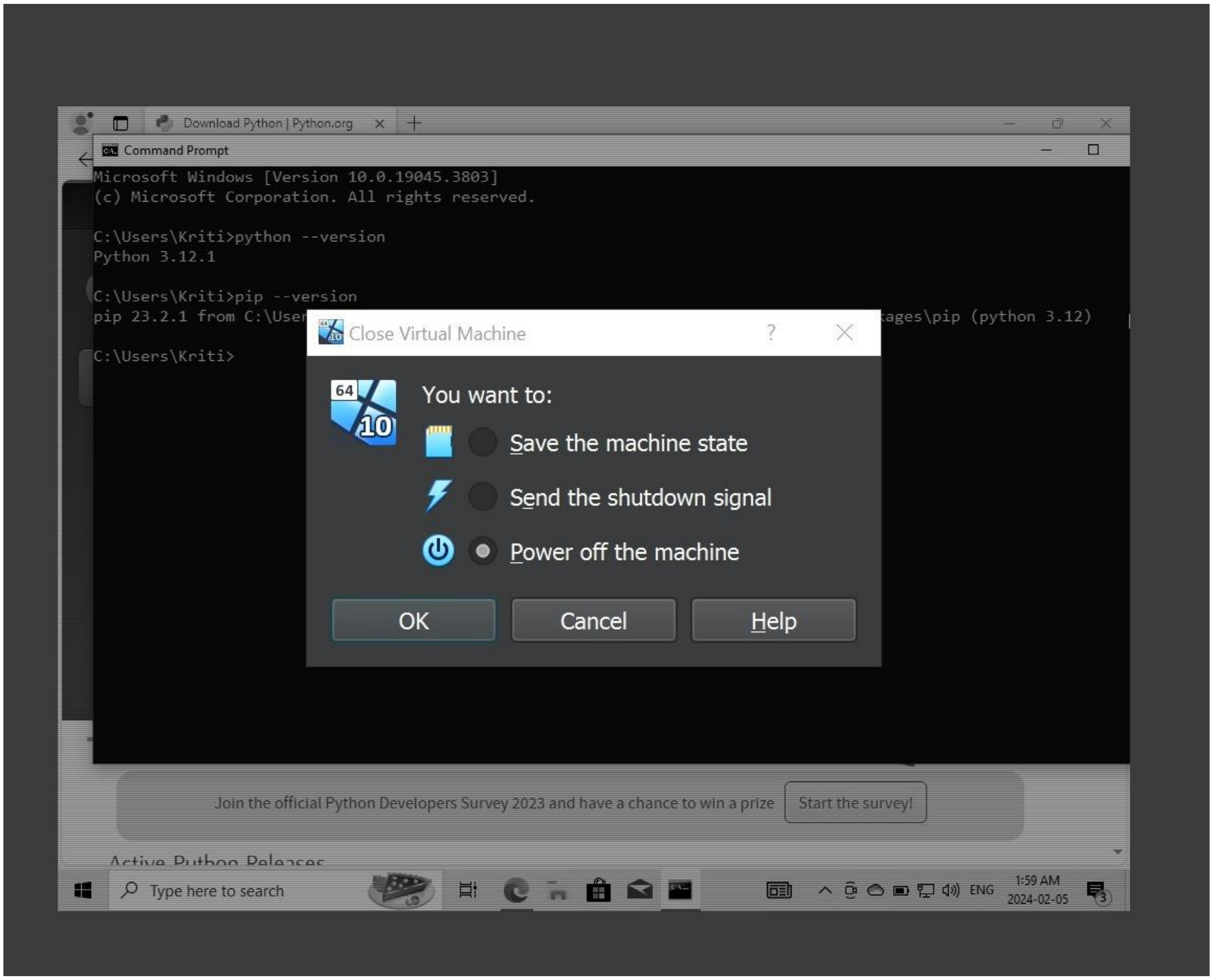
```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kriti>python --version
Python 3.12.1

C:\Users\Kriti>pip
```

Download Python | Python.org
Command Prompt
Download Python | Python.org ...
Type here to search
ENG
1:54 AM
2024-02-05



Download Python | Python.org
Command Prompt
1 item
Join the official Python Developers Survey 2023 and have a chance to win a prize
Start the survey!
Active Python Releases
Type here to search
ENG
1:56 AM
2024-02-05

```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kriti>python --version
Python 3.12.1

C:\Users\Kriti>pip --version
pip 23.2.1 from C:\Users\Kriti\AppData\Local\Programs\Python\Python312\Lib\site-packages\pip (python 3.12)

C:\Users\Kriti>
```



Download Python | Python.org
Command Prompt

```
Microsoft Windows [Version 10.0.19045.3803]
(c) Microsoft Corporation. All rights reserved.

C:\Users\Kriti>python --version
Python 3.12.1

C:\Users\Kriti>pip --version
pip 23.2.1 from C:\User                                                          ages\pip (python 3.12)

C:\Users\Kriti>
```

Close Virtual Machine

You want to:

○ Save the machine state

○ Send the shutdown signal

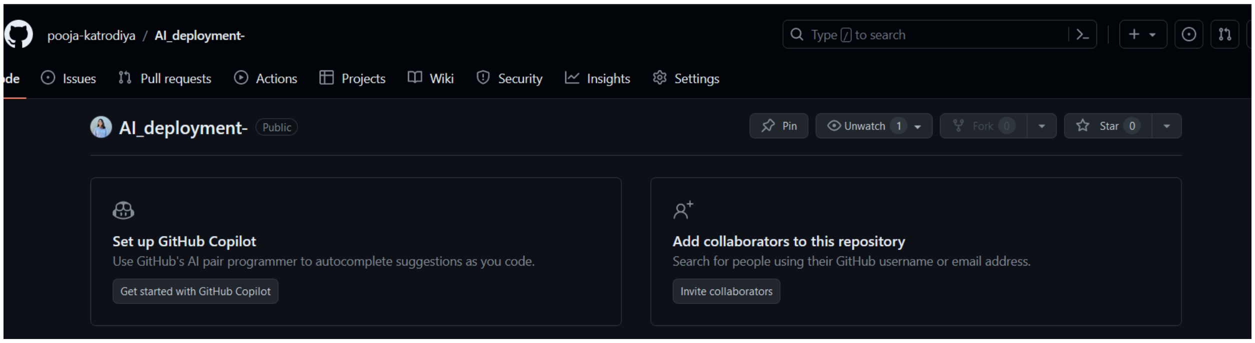
● Power off the machine

64
10
OK
Cancel
Help
Join the official Python Developers Survey 2023 and have a chance to win a prize
Start the survey!
Active Python Releases
Type here to search
ENG
1:59 AM
2024-02-05

Task 2: Version Control (Link: Github)

1. Create a Git Repository:

Initialize a new Git repository on your local machine.



Set up GitHub Copilot

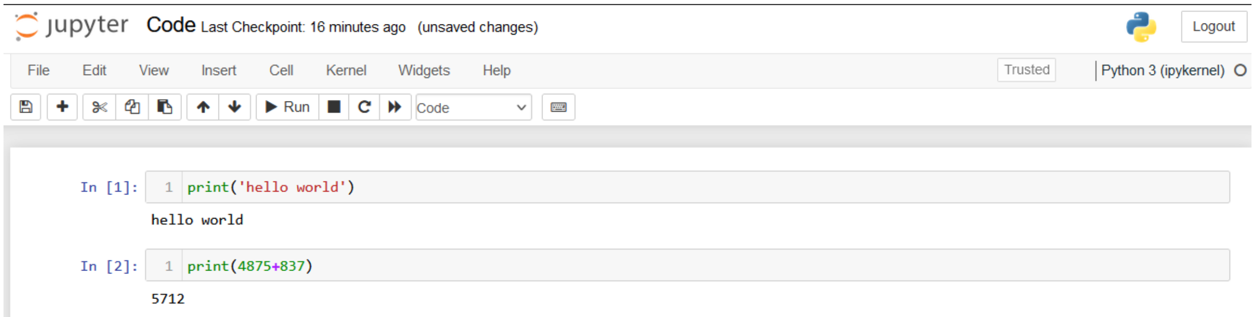
Use GitHub's AI pair programmer to autocomplete suggestions as you code.

Add collaborators to this repository

Search for people using their GitHub username or email address.

2. Write AI Solution:

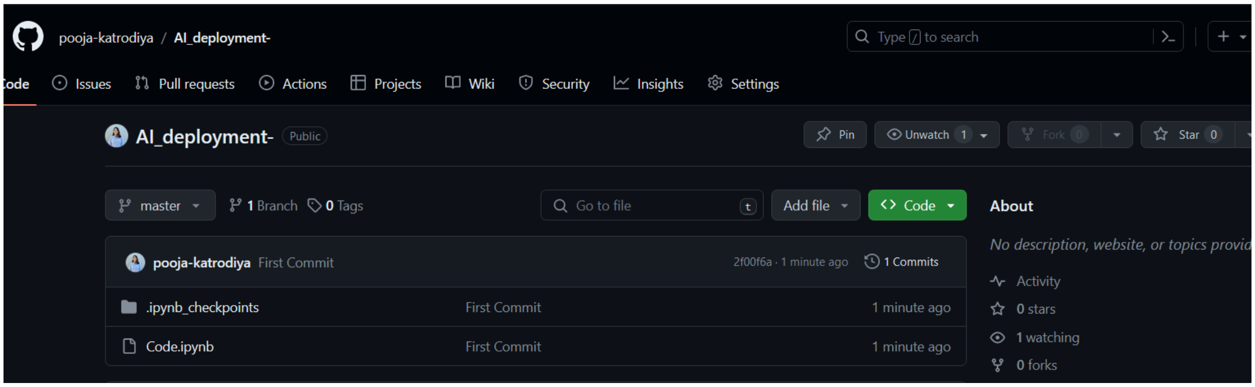
Write a simple Python script or use an existing one. This can be a basic "Hello, World!" script or a simple function.



```
In [1]: print('hello world')

        hello world

In [2]: print(4875+837)

        5712
```

3. Commit Initial Code:

Add and commit the initial code to the Git repository.



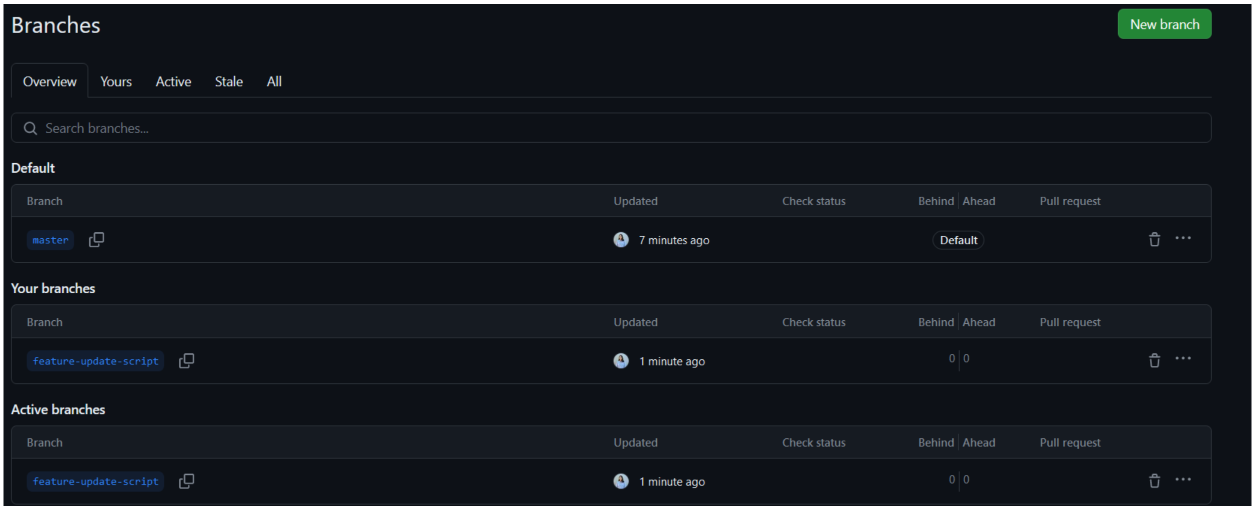
pooja-katrodiya First Commit 2f00f6a · 1 minute ago 1 Commits

.ipynb_checkpoints	First Commit	1 minute ago
Code.ipynb	First Commit	1 minute ago

No description, website, or topics provid

4. Create a Feature Branch:

Create a new branch for a feature or modification. For example: git checkout -b feature-update-script.



Default

Branch	Updated	Check status	Behind	Ahead	Pull request
master	7 minutes ago		Default		

Your branches

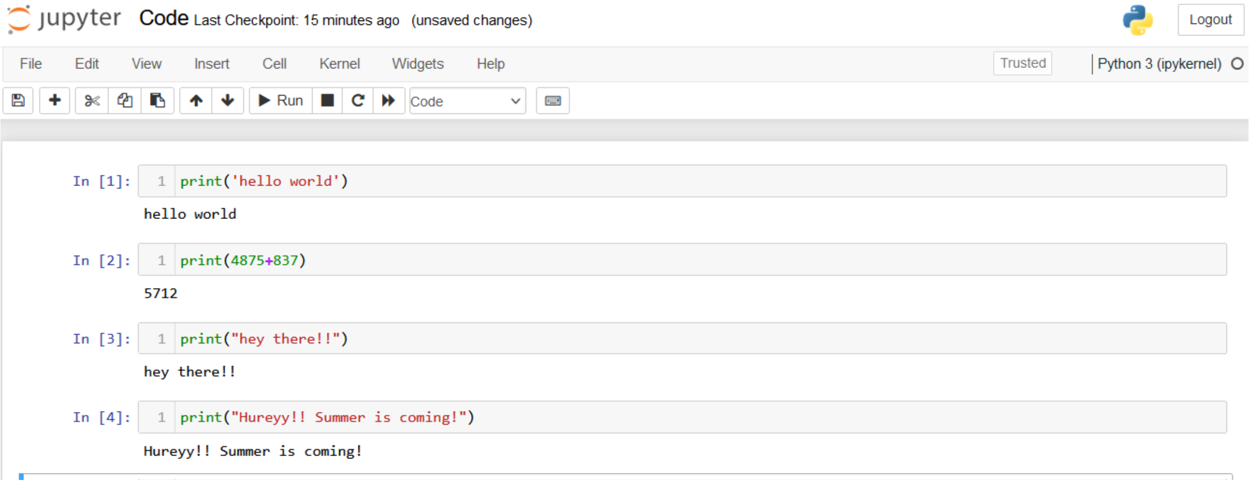
Branch	Updated	Check status	Behind	Ahead	Pull request
feature-update-script	1 minute ago		0	0	

Active branches

Branch	Updated	Check status	Behind	Ahead	Pull request
feature-update-script	1 minute ago		0	0	

5. Make a Modification:

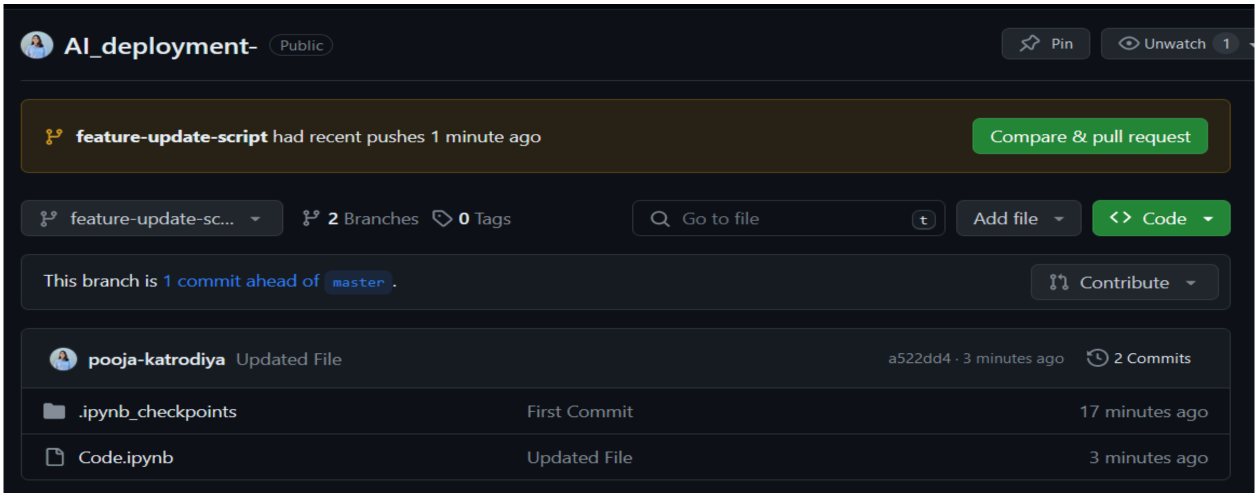
Modify the Python script. For instance, add a new function or change an existing one.



```
In [1]: print('hello world')
hello world

In [2]: print(4875+837)
5712

In [3]: print("hey there!!")
hey there!!

In [4]: print("Hureyy!! Summer is coming!")
Hureyy!! Summer is coming!
```

6. Commit Changes:

Commit the changes to the feature branch.



feature-update-script had recent pushes 1 minute ago

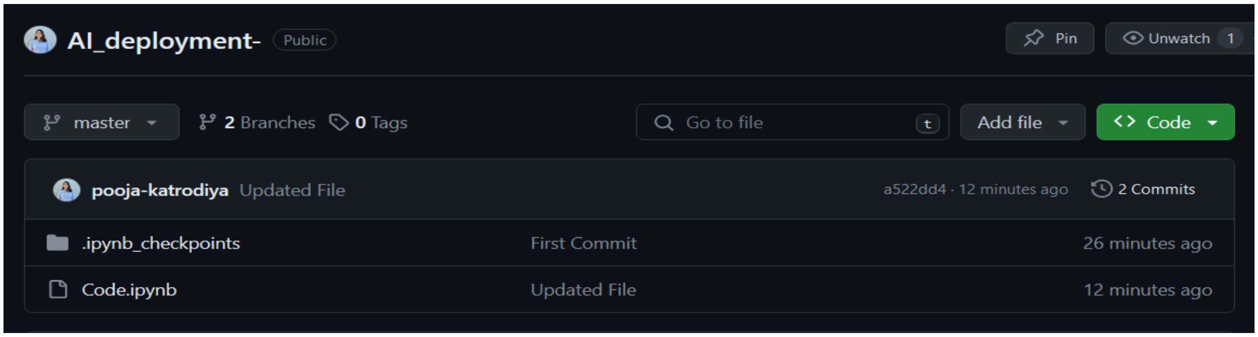
This branch is 1 commit ahead of `master`.

pooja-katrodiya Updated File a522dd4 · 3 minutes ago

.ipynb_checkpoints	First Commit	17 minutes ago
Code.ipynb	Updated File	3 minutes ago

7. Merge Back to Main:

Switch back to the main branch (git checkout main) and merge the feature branch into it (git merge feature-update-script).



pooja-katrodiya Updated File a522dd4 · 12 minutes ago

.ipynb_checkpoints	First Commit	26 minutes ago
Code.ipynb	Updated File	12 minutes ago

First, I created a Git repository named **'AI_deployment'** and then created a Python file. Next, I opened a command prompt where the Python file was saved. In the command prompt, I followed these commands to push the code file:

1. git init: This initialized a new Git repository locally.
2. git remote add origin https://github.com/pooja-katrodiya/AI_deployment-.git: This linked the local repository to a remote repository.
3. git status: I checked the status of the local repository.
4. git commit: I saved changes to the local repository with a descriptive message('First Commit').
5. git push: I uploaded my local commits to the remote repository.

After that, I created a new feature branch for modification with the following code: git checkout -b feature-update-script. I updated my Python file and committed that code into the feature branch, then merged it back into the main branch.

Task 3: Build and Deploy

1. Choose a CI/CD Tool: Choose a CI/CD tool (e.g., Jenkins, GitLab CI, GitHub Actions) for automating the deployment pipeline.

Answer: We are going with GitHub for deployment of pipeline.

2. Configure Pipeline: Configure the pipeline to trigger on changes to the main branch.

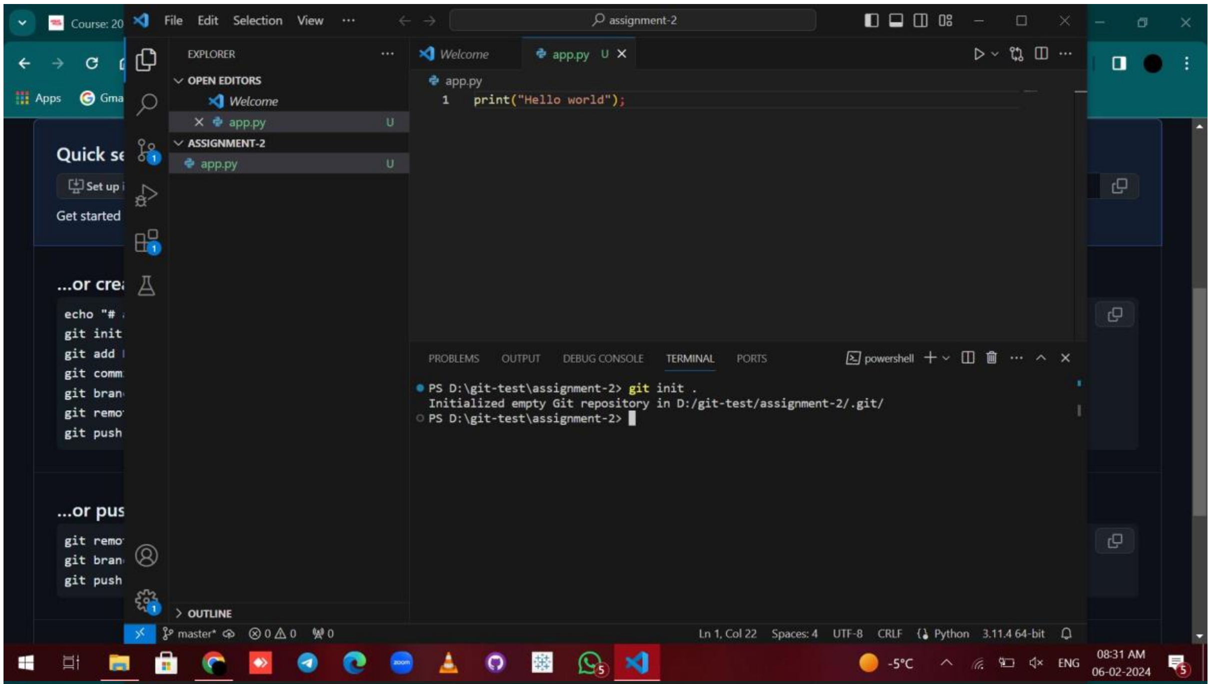
3. Define Build Stage: Define a build stage that includes linting and testing the Python script. Write necessary scripts or configurations.

4. Deploy to Local Environment: If tests pass, add a deployment stage to deploy the AI solution to a simple local environment. For example, you might use Vagrant to create another instance for deployment.

5. Pipeline Execution: Manually trigger the pipeline (or wait for it to be triggered by a push to the main branch) and observe the pipeline execution.

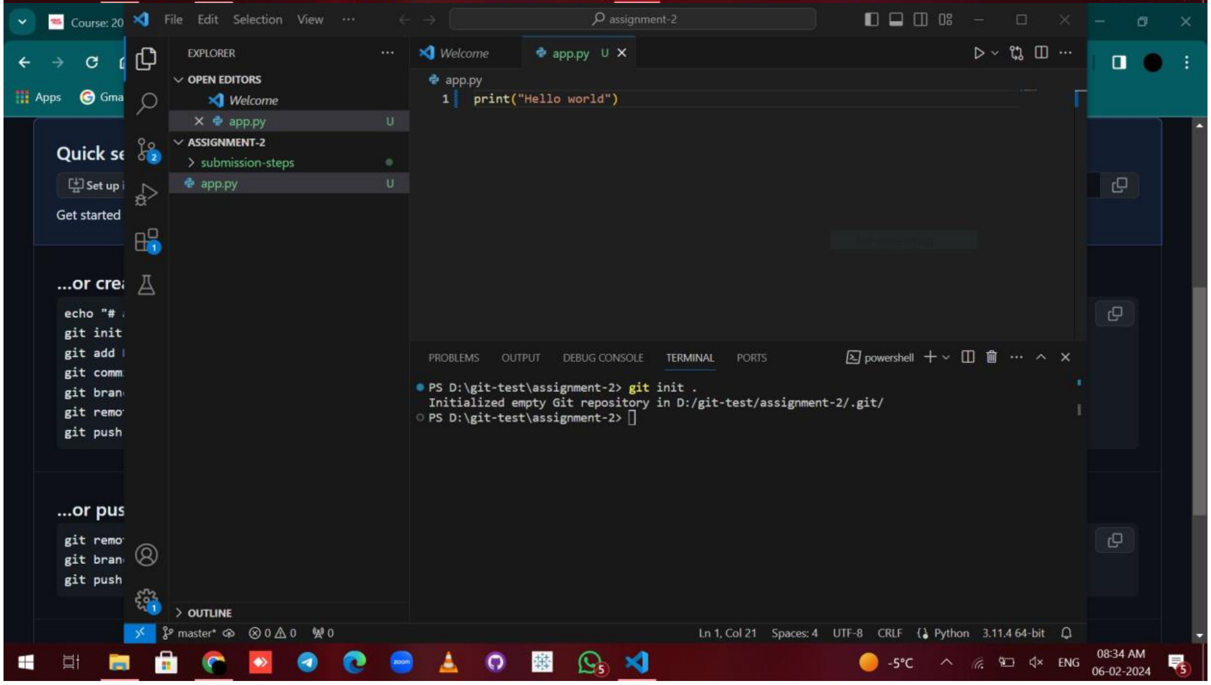
6. Verify Deployment: Connect to the deployed environment and verify that the AI solution is deployed successfully.

Answer: We have attached each step used to accomplish the task mentioned above.



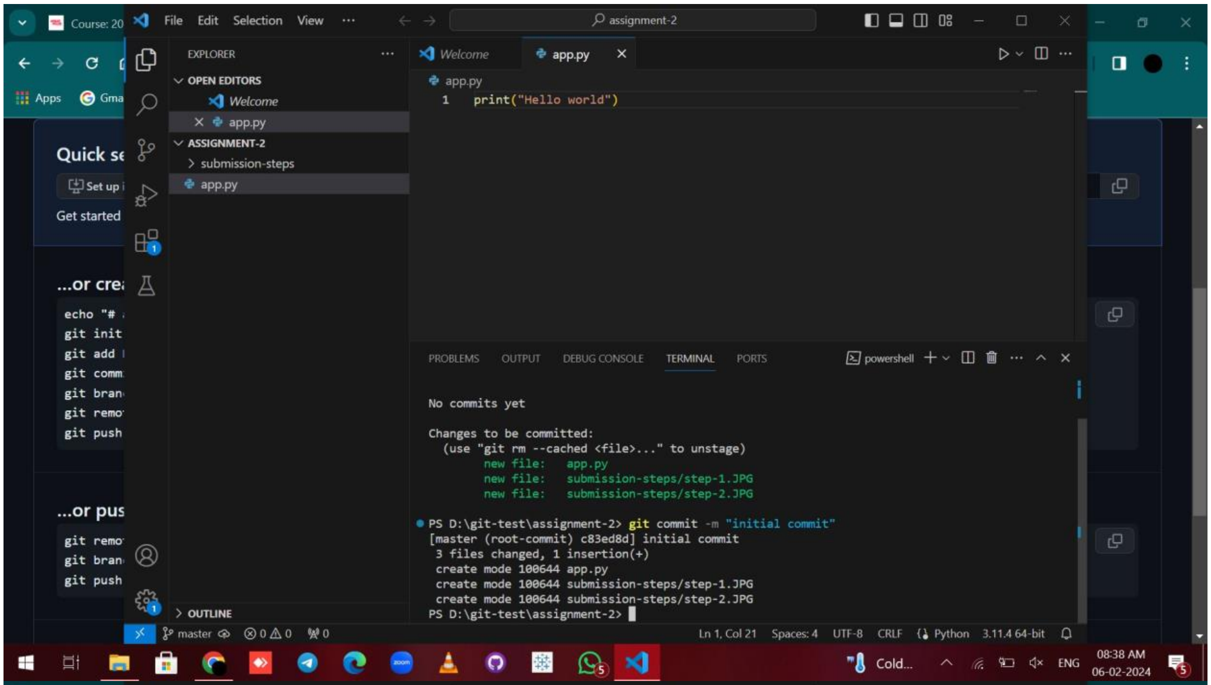
```
print("Hello world");
```

```
PS D:\git-test\assignment-2> git init .
Initialized empty Git repository in D:/git-test/assignment-2/.git/
PS D:\git-test\assignment-2>
```



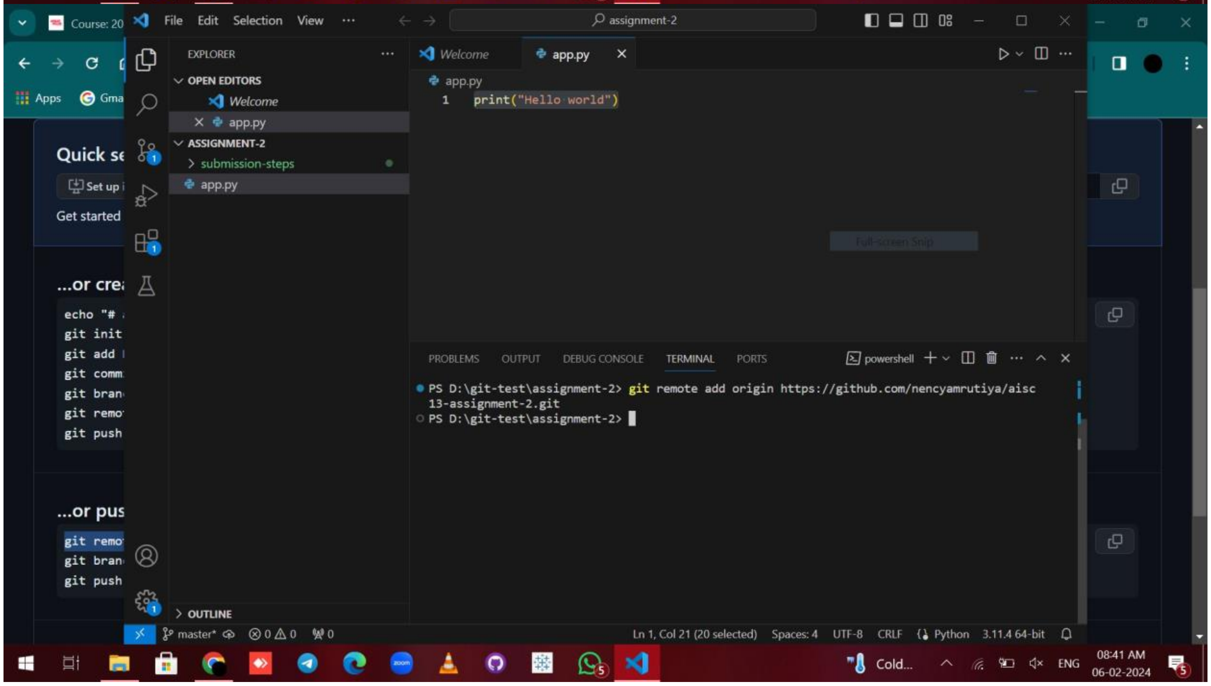
```
print("Hello world")
```

```
PS D:\git-test\assignment-2> git init .
Initialized empty Git repository in D:/git-test/assignment-2/.git/
PS D:\git-test\assignment-2>
```



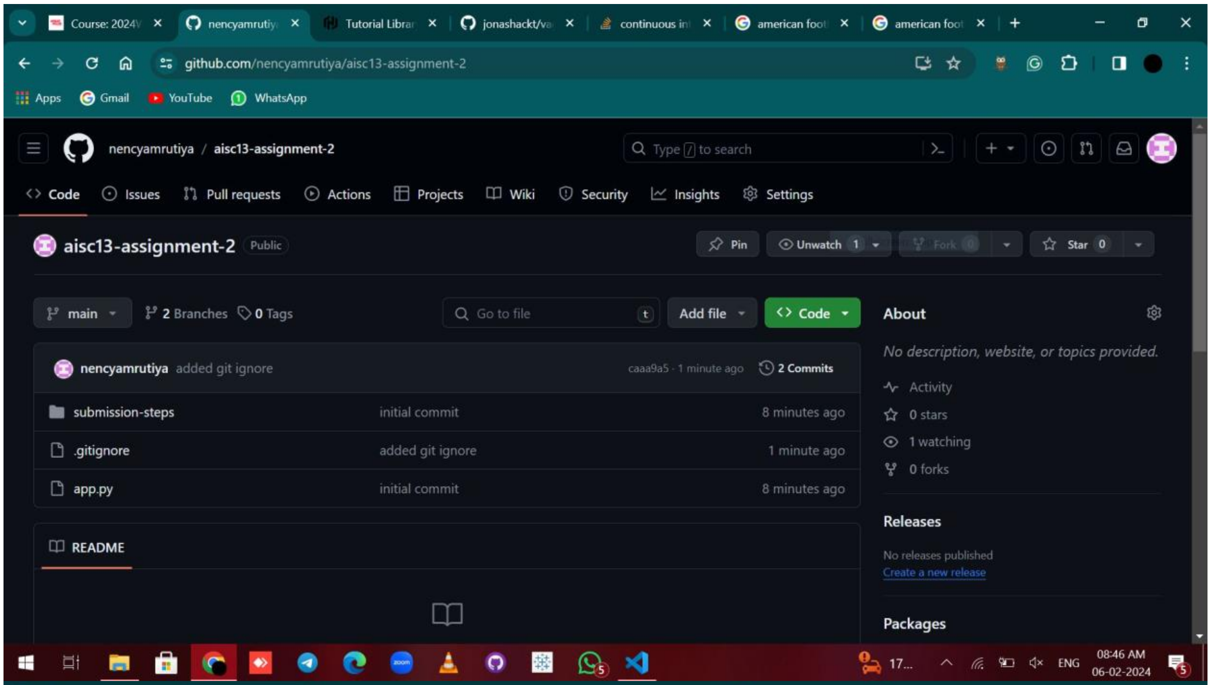
```
print("Hello world")
```

```
No commits yet

Changes to be committed:
  (use "git rm --cached <file>..." to unstage)
        new file:   app.py
        new file:   submission-steps/step-1.JPG
        new file:   submission-steps/step-2.JPG

PS D:\git-test\assignment-2> git commit -m "initial commit"
[master (root-commit) c83ed8d] initial commit
 3 files changed, 1 insertion(+)
 create mode 100644 app.py
 create mode 100644 submission-steps/step-1.JPG
 create mode 100644 submission-steps/step-2.JPG
PS D:\git-test\assignment-2>
```



```
print("Hello world")
```

```
PS D:\git-test\assignment-2> git remote add origin https://github.com/nencyamrutiya/aisc
13-assignment-2.git
PS D:\git-test\assignment-2>
```



github.com/nencyamrutiya/aisc13-assignment-2

nencyamrutiya / aisc13-assignment-2

aisc13-assignment-2 Public

main 2 Branches 0 Tags

nencyamrutiya added git ignore caaa9a5 · 1 minute ago 2 Commits

submission-steps	initial commit	8 minutes ago
.gitignore	added git ignore	1 minute ago
app.py	initial commit	8 minutes ago

README

About

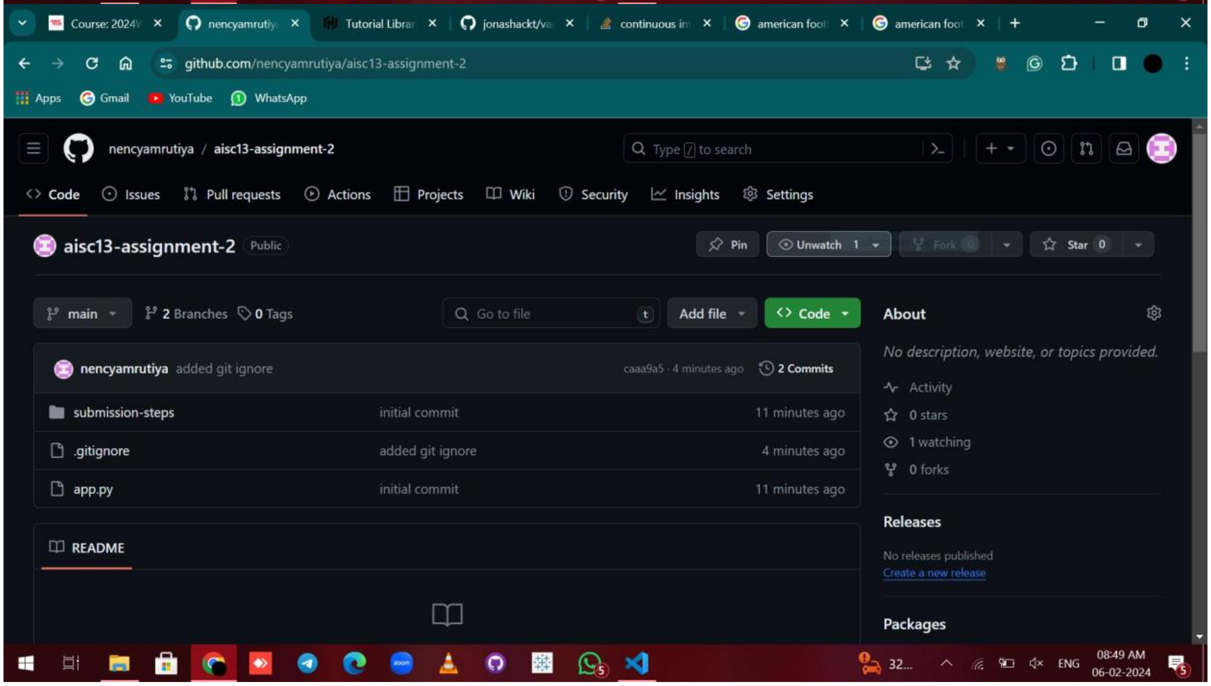
No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages



github.com/nencyamrutiya/aisc13-assignment-2

nencyamrutiya / aisc13-assignment-2

aisc13-assignment-2 Public

main 2 Branches 0 Tags

nencyamrutiya added git ignore caaa9a5 · 4 minutes ago 2 Commits

submission-steps	initial commit	11 minutes ago
.gitignore	added git ignore	4 minutes ago
app.py	initial commit	11 minutes ago

README

About

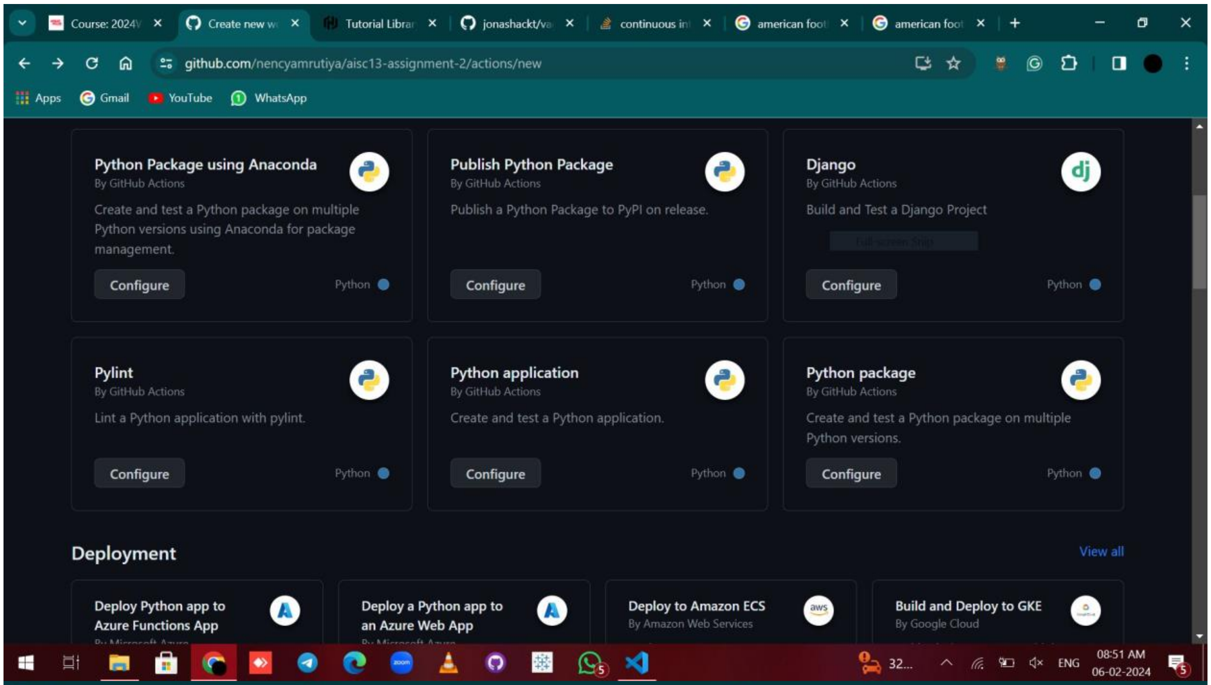
No description, website, or topics provided.

Activity
0 stars
1 watching
0 forks

Releases

No releases published
Create a new release

Packages



Python Package using Anaconda
By GitHub Actions
Create and test a Python package on multiple Python versions using Anaconda for package management.
Configure — Python

Publish Python Package
By GitHub Actions
Publish a Python Package to PyPI on release.
Configure — Python

Django
By GitHub Actions
Build and Test a Django Project
Configure — Python

Pylint
By GitHub Actions
Lint a Python application with pylint.
Configure — Python

Python application
By GitHub Actions
Create and test a Python application.
Configure — Python

Python package
By GitHub Actions
Create and test a Python package on multiple Python versions.
Configure — Python

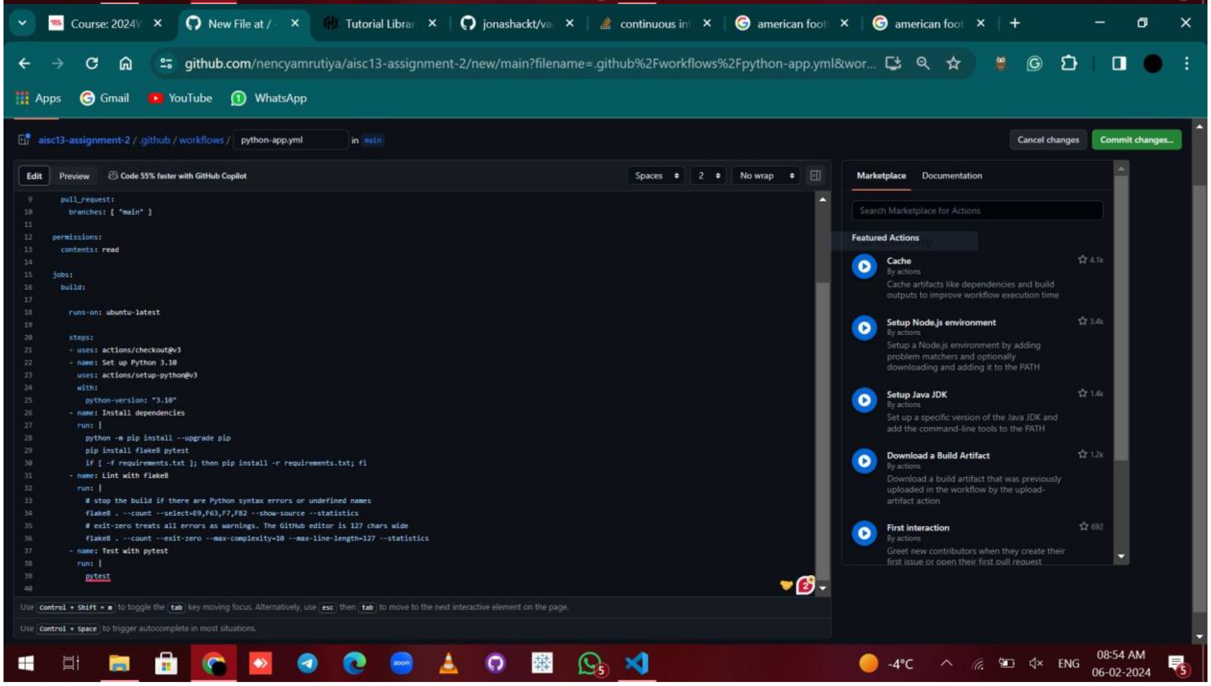
Deployment — View all

Deploy Python app to Azure Functions App
By Microsoft Azure

Deploy a Python app to an Azure Web App
By Microsoft Azure

Deploy to Amazon ECS
By Amazon Web Services

Build and Deploy to GKE
By Google Cloud



aisc13-assignment-2 / .github / workflows / python-app.yml in main

```
  pull_request:
    branches: [ "main" ]

permissions:
  contents: read

jobs:
  build:

    runs-on: ubuntu-latest

    steps:
    - uses: actions/checkout@v3
    - name: Set up Python 3.10
      uses: actions/setup-python@v3
      with:
        python-version: "3.10"
    - name: Install dependencies
      run: |
        python -m pip install --upgrade pip
        pip install flake8 pytest
        if [ -f requirements.txt ]; then pip install -r requirements.txt; fi
    - name: Lint with flake8
      run: |
        # stop the build if there are Python syntax errors or undefined names
        flake8 . --count --select=E9,F63,F7,F82 --show-source --statistics
        # exit-zero treats all errors as warnings. The GitHub editor is 127 chars wide
        flake8 . --count --exit-zero --max-complexity=10 --max-line-length=127 --statistics
    - name: Test with pytest
      run: |
        pytest
```

Use Control + Shift + m to toggle the tab key moving focus. Alternatively, use esc then tab to move to the next interactive element on the page.

Use Control + Space to trigger autocomplete in most situations.

Featured Actions

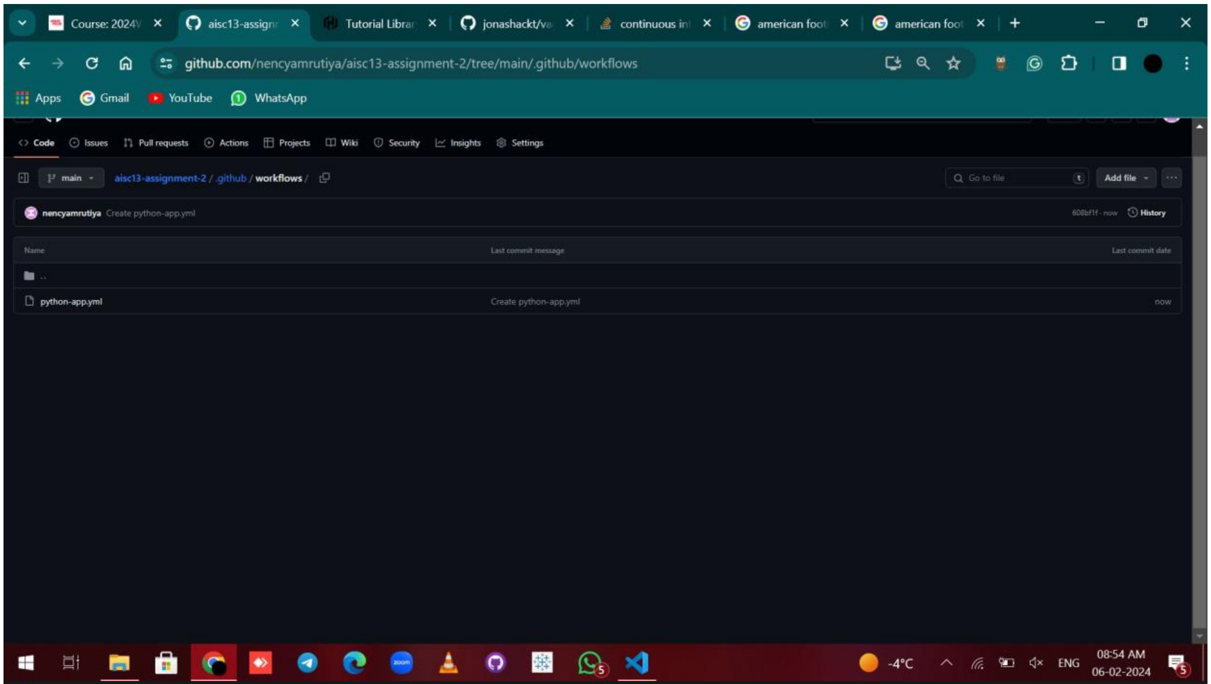
Cache ☆ 4.1k
By actions
Cache artifacts like dependencies and build outputs to improve workflow execution time

Setup Node.js environment ☆ 3.4k
By actions
Setup a Node.js environment by adding problem matchers and optionally downloading and adding it to the PATH

Setup Java JDK ☆ 1.4k
By actions
Set up a specific version of the Java JDK and add the command-line tools to the PATH

Download a Build Artifact ☆ 1.2k
By actions
Download a build artifact that was previously uploaded in the workflow by the upload-artifact action

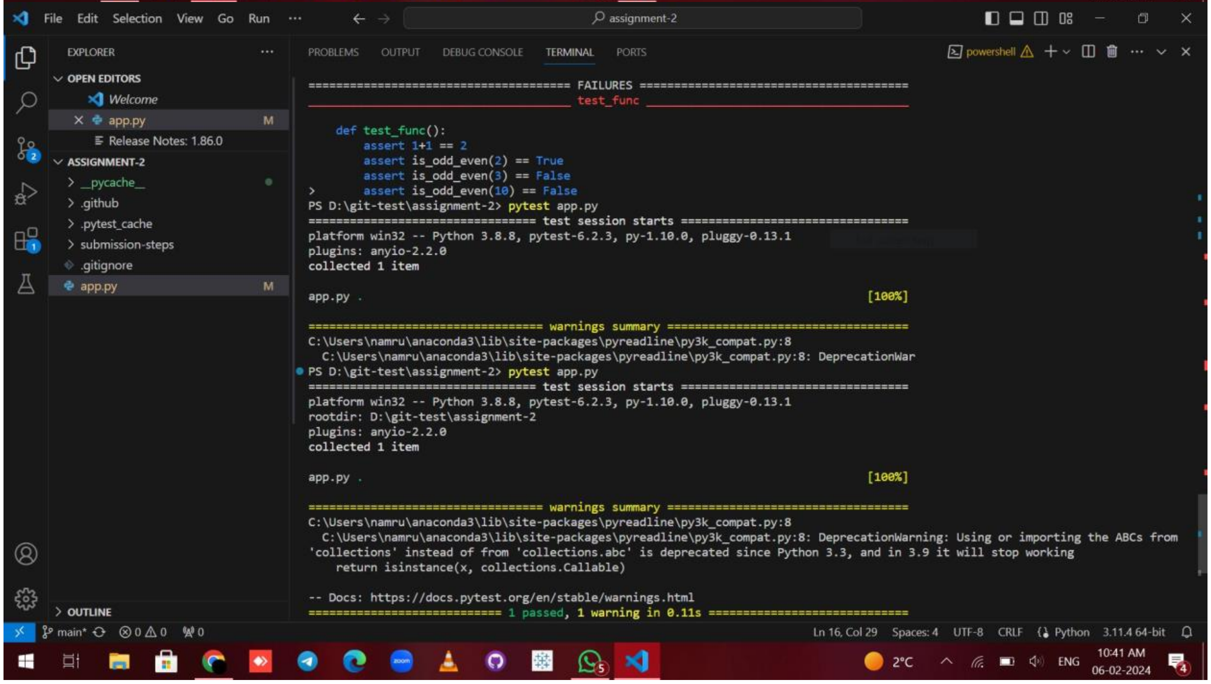
First interaction ☆ 692
By actions
Greet new contributors when they create their first issue or open their first pull request



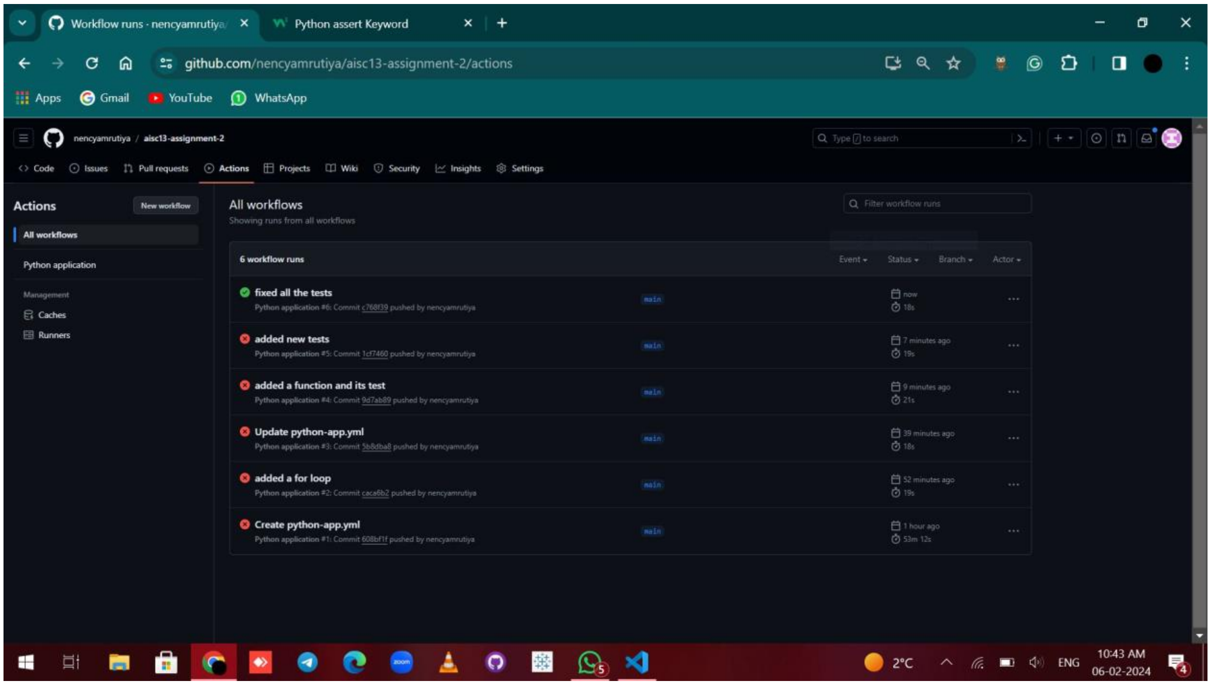
aisc13-assignment-2 / .github / workflows /

nencyamrutiya Create python-app.yml 608bf1f · now

Name	Last commit message	Last commit date
..		
python-app.yml	Create python-app.yml	now



```
====================================== FAILURES ======================================
_____________________________________ test_func ______________________________________

    def test_func():
        assert 1+1 == 2
        assert is_odd_even(2) == True
        assert is_odd_even(3) == False
>       assert is_odd_even(10) == False
PS D:\git-test\assignment-2> pytest app.py
================================= test session starts =================================
platform win32 -- Python 3.8.8, pytest-6.2.3, py-1.10.0, pluggy-0.13.1
plugins: anyio-2.2.0
collected 1 item

app.py .                                                                        [100%]

================================== warnings summary ===================================
C:\Users\namru\anaconda3\lib\site-packages\pyreadline\py3k_compat.py:8
  C:\Users\namru\anaconda3\lib\site-packages\pyreadline\py3k_compat.py:8: DeprecationWar
PS D:\git-test\assignment-2> pytest app.py
================================= test session starts =================================
platform win32 -- Python 3.8.8, pytest-6.2.3, py-1.10.0, pluggy-0.13.1
rootdir: D:\git-test\assignment-2
plugins: anyio-2.2.0
collected 1 item

app.py .                                                                        [100%]

================================== warnings summary ===================================
C:\Users\namru\anaconda3\lib\site-packages\pyreadline\py3k_compat.py:8
  C:\Users\namru\anaconda3\lib\site-packages\pyreadline\py3k_compat.py:8: DeprecationWarning: Using or importing the ABCs from
'collections' instead of from 'collections.abc' is deprecated since Python 3.3, and in 3.9 it will stop working
    return isinstance(x, collections.Callable)

-- Docs: https://docs.pytest.org/en/stable/warnings.html
============================ 1 passed, 1 warning in 0.11s =============================
```

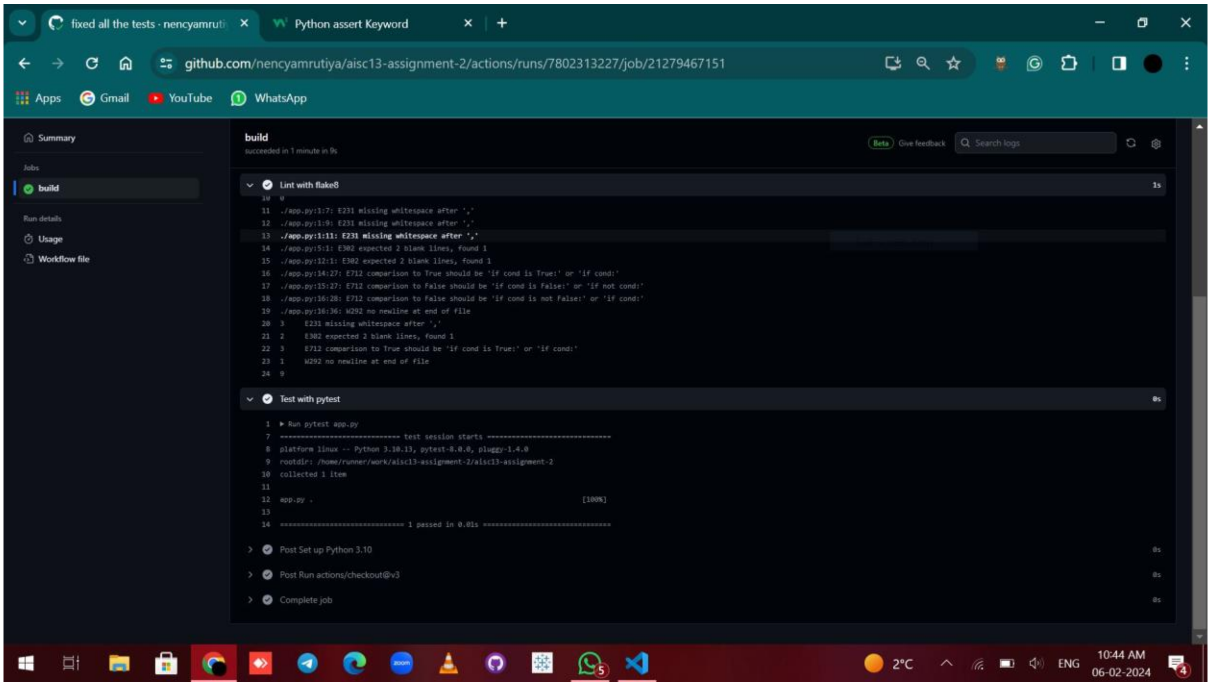


All workflows

Showing runs from all workflows

6 workflow runs

✓ fixed all the tests Python application #6: Commit c768f39 pushed by nencyamrutiya	main	now 18s	···
✗ added new tests Python application #5: Commit 1cf7460 pushed by nencyamrutiya	main	7 minutes ago 19s	···
✗ added a function and its test Python application #4: Commit 9d7ab89 pushed by nencyamrutiya	main	9 minutes ago 21s	···
✗ Update python-app.yml Python application #3: Commit 5b8dba8 pushed by nencyamrutiya	main	39 minutes ago 18s	···
✗ added a for loop Python application #2: Commit caca6b2 pushed by nencyamrutiya	main	52 minutes ago 19s	···
✗ Create python-app.yml Python application #1: Commit 608bf1f pushed by nencyamrutiya	main	1 hour ago 53m 12s	···



build

succeeded in 1 minute in 9s

✓ Lint with flake8 1s

```
0
./app.py:1:7: E231 missing whitespace after ','
./app.py:1:9: E231 missing whitespace after ','
./app.py:1:11: E231 missing whitespace after ','
./app.py:5:1: E302 expected 2 blank lines, found 1
./app.py:12:1: E302 expected 2 blank lines, found 1
./app.py:14:27: E712 comparison to True should be 'if cond is True:' or 'if cond:'
./app.py:15:27: E712 comparison to False should be 'if cond is False:' or 'if not cond:'
./app.py:16:28: E712 comparison to False should be 'if cond is not False:' or 'if cond:'
./app.py:16:36: W292 no newline at end of file
3     E231 missing whitespace after ','
2     E302 expected 2 blank lines, found 1
3     E712 comparison to True should be 'if cond is True:' or 'if cond:'
1     W292 no newline at end of file
9
```

✓ Test with pytest 0s

```
▶ Run pytest app.py
============================== test session starts ==============================
platform linux -- Python 3.10.13, pytest-8.0.0, pluggy-1.4.0
rootdir: /home/runner/work/aisc13-assignment-2/aisc13-assignment-2
collected 1 item

app.py .                                                                  [100%]

=============================== 1 passed in 0.01s ===============================
```

✓ Post Set up Python 3.10 0s

✓ Post Run actions/checkout@v3 0s

✓ Complete job 0s

Task 4: Documentation

Development Environment

We use the standard Python setup to build up a development environment on our local machine.

Version Control Process

We staged the changes to GitHub after the application was operational.

Branching was used to stage new features without affecting the main code.
We submitted a pull request to add the new feature to our primary production branch as soon as we felt secure.

CI/CD Pipeline

Used the Python application pipeline utilized in github actions included linting, unit testing, and testing support.

Set the trigger to every push on the 'main' branch of production.

Deployment Verification

After every commit we checked that our application is Passing all the checks on the pipeline.

Conclusion

The main lesson to be learned from this project is that a well-functioning CI/CD and version control system is essential for efficient production workflow.
The host IP address setting was the only difficulty we encountered. To be available across the entire network, we had to instruct our application to run on the 0.0.0.0 IP address.