**Application Development and Deployment Architecture**

**CSI3025- L35 + L36**

**LAB-FAT**

|  |  |
| --- | --- |
| 29th April 2024 | Ananya Singh |
|  | [20MIC0095] |

Code:

*ananya.py*

def factorial(n):

    """

    Calculate the factorial of a number.

    Args:

        n (int): The number to calculate factorial for.

    Returns:

        int: The factorial of n.

    """

    if n == 0:

        return 1

    return n \* factorial(n - 1)

def fibonacci(n):

    """

    Generate the Fibonacci sequence up to the nth term.

    Args:

        n (int): The number of terms in the Fibonacci sequence.

    Returns:

        list: The Fibonacci sequence.

    """

    if n <= 1:

        return n

    return fibonacci(n - 1) + fibonacci(n - 2)

if \_\_name\_\_ == "\_\_main\_\_":

    N = 17

    # Calculate factorial

    fact = factorial(N)

    print(f"Factorial of {N} is: {fact}")

    # Calculate Fibonacci sequence

    fib\_sequence = [fibonacci(i) for i in range(N)]

    print(f"Fibonacci sequence of {N} numbers is: {fib\_sequence}")

ci-cd.yml

name: Python CI

on:

  push:

    branches:

      - main

jobs:

  build:

    runs-on: ubuntu-latest

    steps:

    - name: Checkout code

      uses: actions/checkout@v2

    - name: Set up Python

      uses: actions/setup-python@v2

      with:

        python-version: 3.8

    - name: Install dependencies

      run: |

        python -m pip install --upgrade pip

    - name: Run Python script

      run: |

        python ananya.py

  check\_quality:

    runs-on: ubuntu-latest

    steps:

    - name: Checkout code

      uses: actions/checkout@v2

    - name: Set up Python

      uses: actions/setup-python@v2

      with:

        python-version: 3.8

    - name: Install dependencies

      run: |

        python -m pip install --upgrade pip

        pip install pylint

    - name: Run pylint

      run: |

        pylint ananya.py

Final Output:

A screenshot of a computer

Description automatically generated

A screenshot of a computer

Description automatically generated

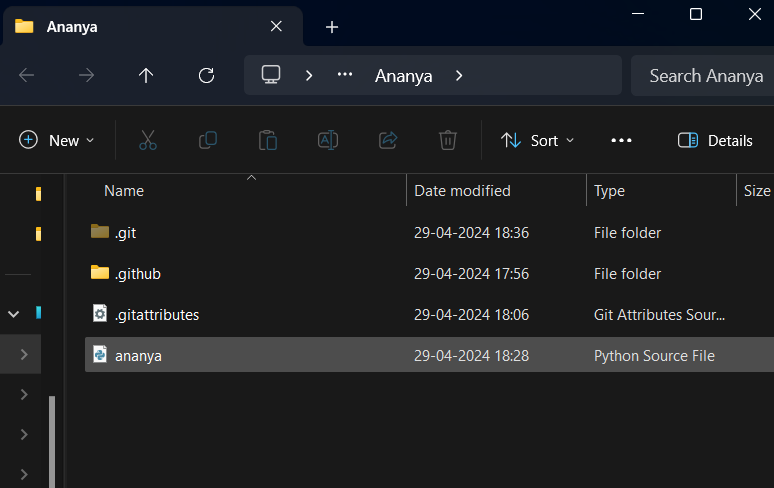
A screenshot of a computer

Description automatically generated

Here the code quality is 10/10; i.e. optimal

Procedure:

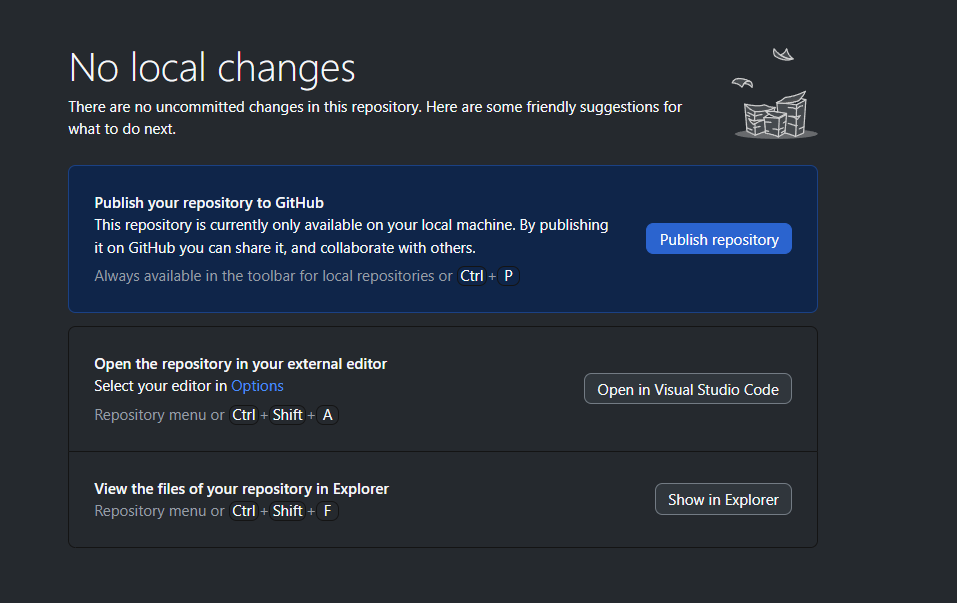
Step-1: Create a folder to save python script and workflow:



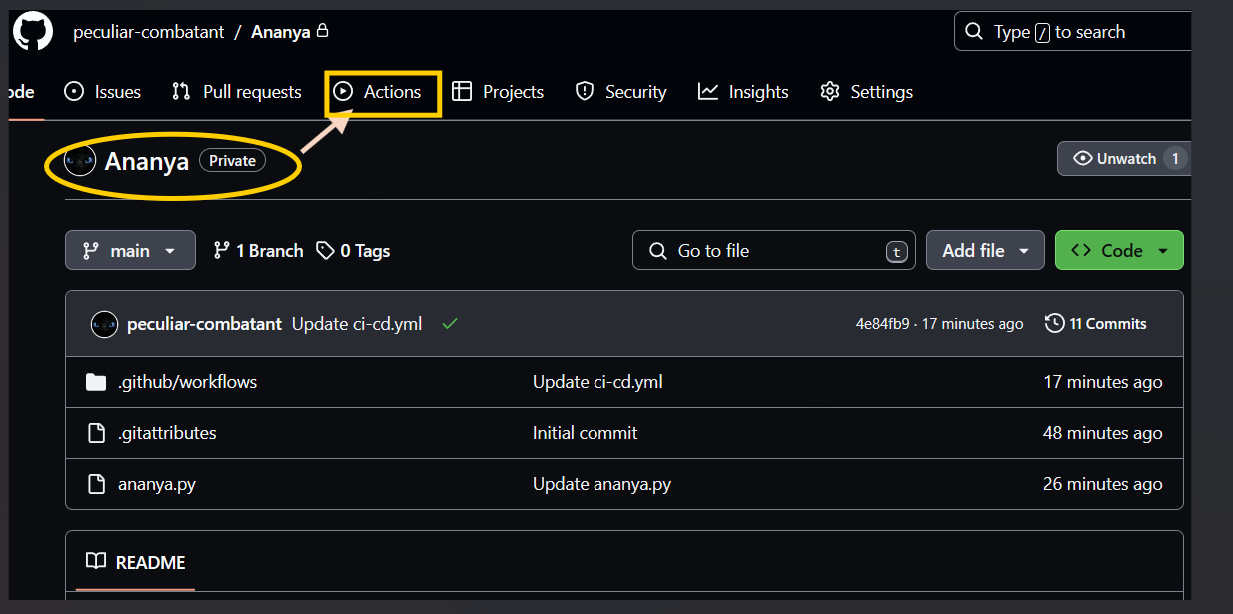
Step-2: Create a local repository and publish it:

A screenshot of a computer

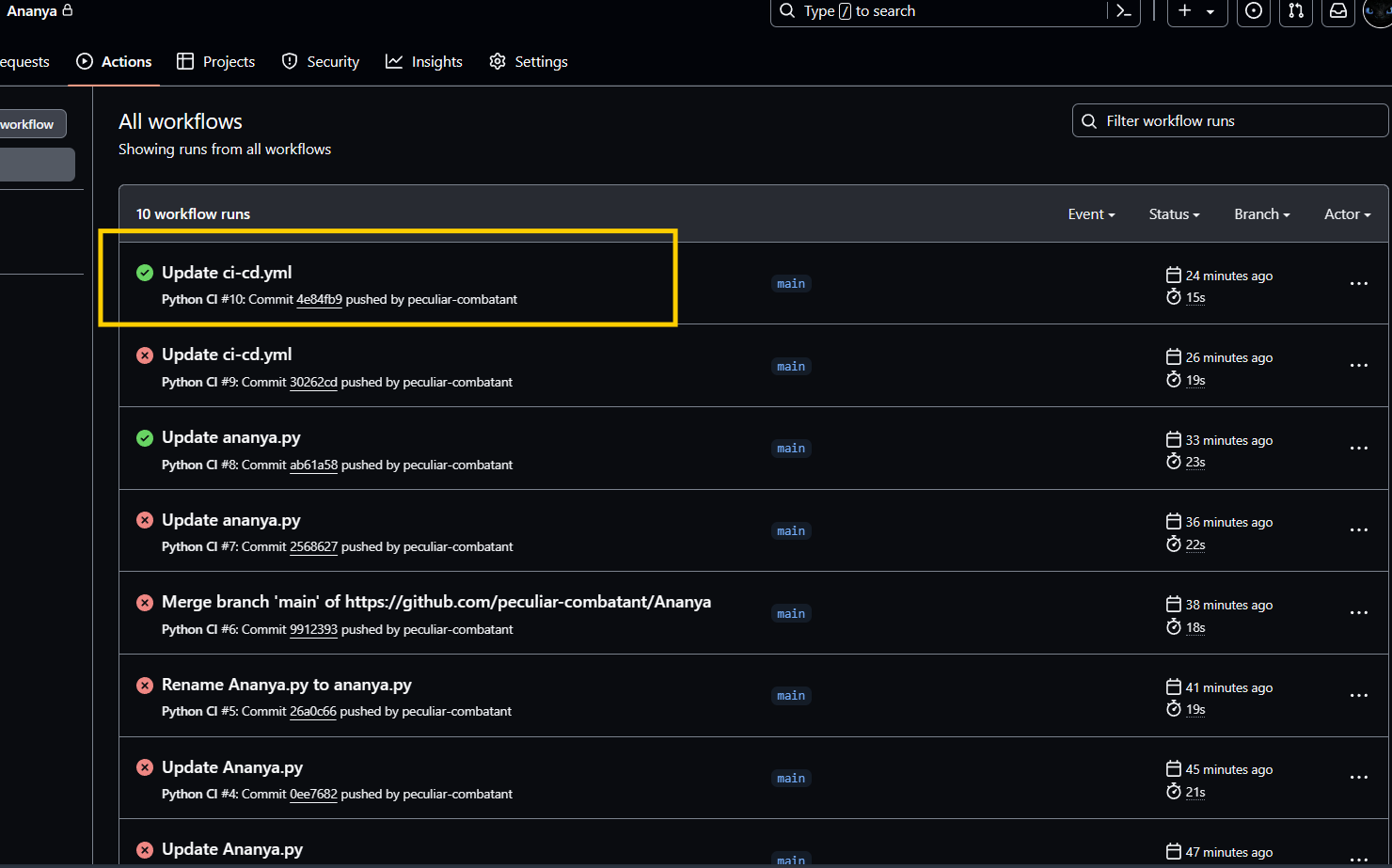
Description automatically generated



Step-3: Go to github and check your repository and click “Actions”:



Step-4 Check in workflow and keep on making changes until and unless the code runs successfully:



Step-5 The final workflow:

A screenshot of a computer

Description automatically generated