CONTAINER ORCHESTRATION AND INFRASTRUCTURE AUTOMATION

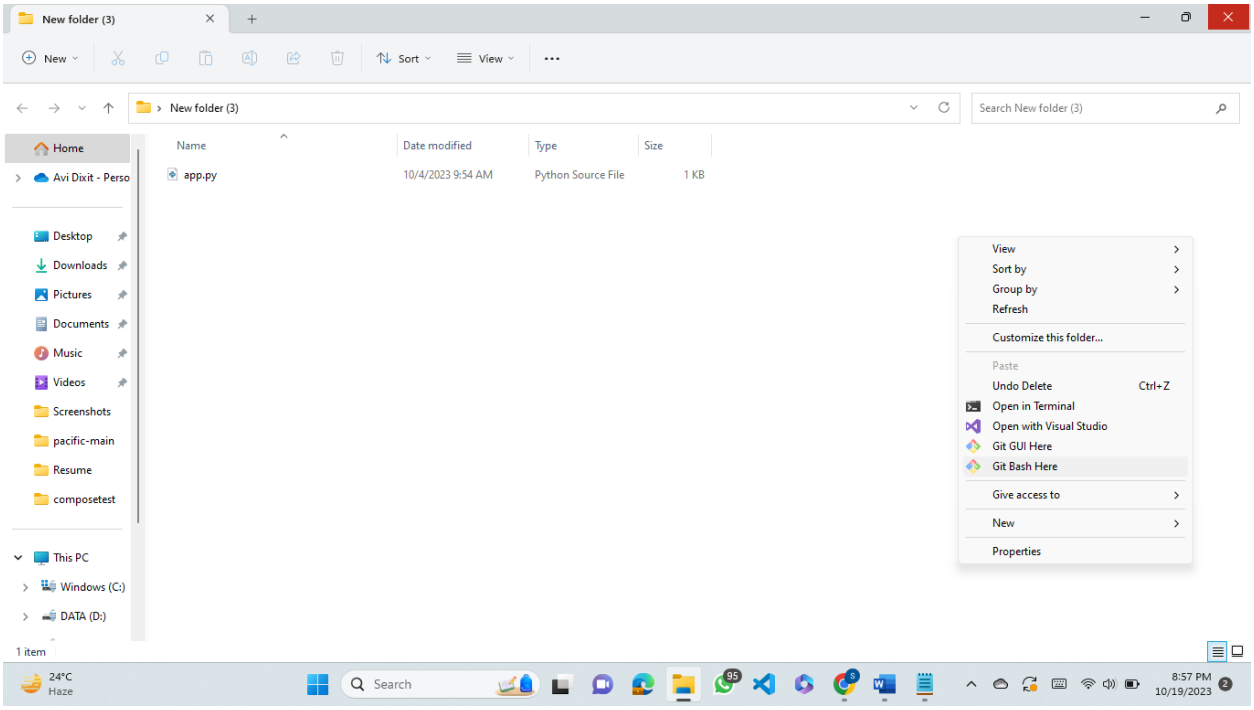
NAME- Shubhi Dixit

SAP ID- 500094571

BATCH- 05

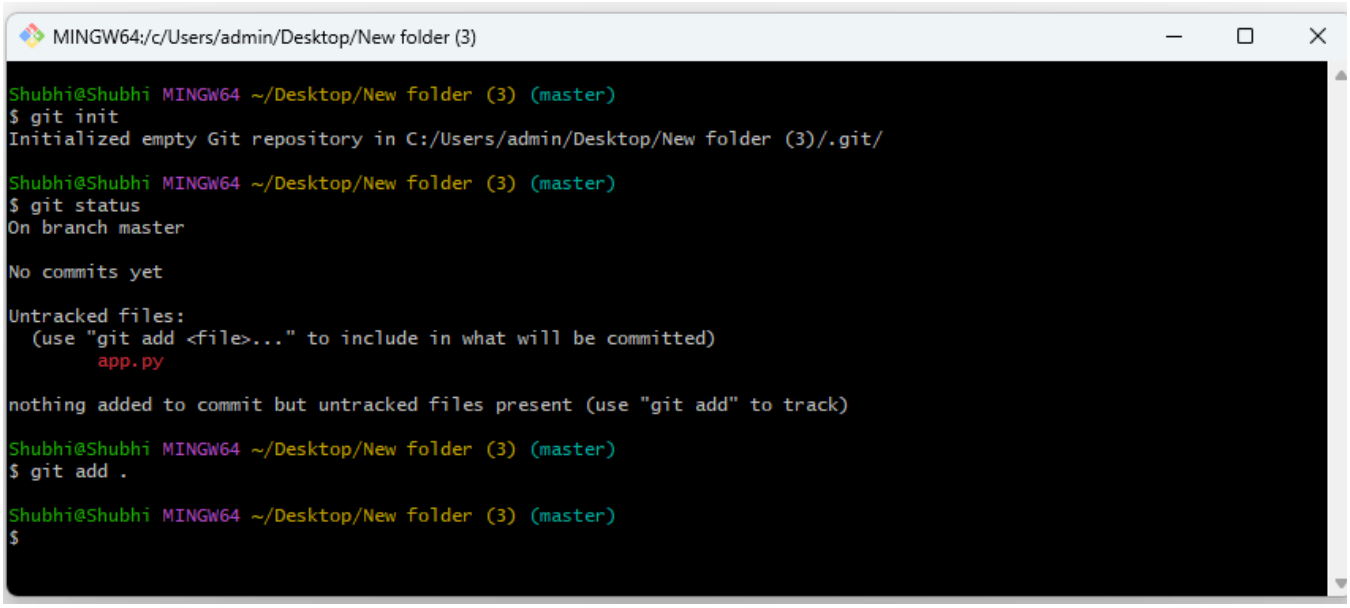
CI/CD Pipeline

Step 1: Go to your project directory and open git bash in the directory.



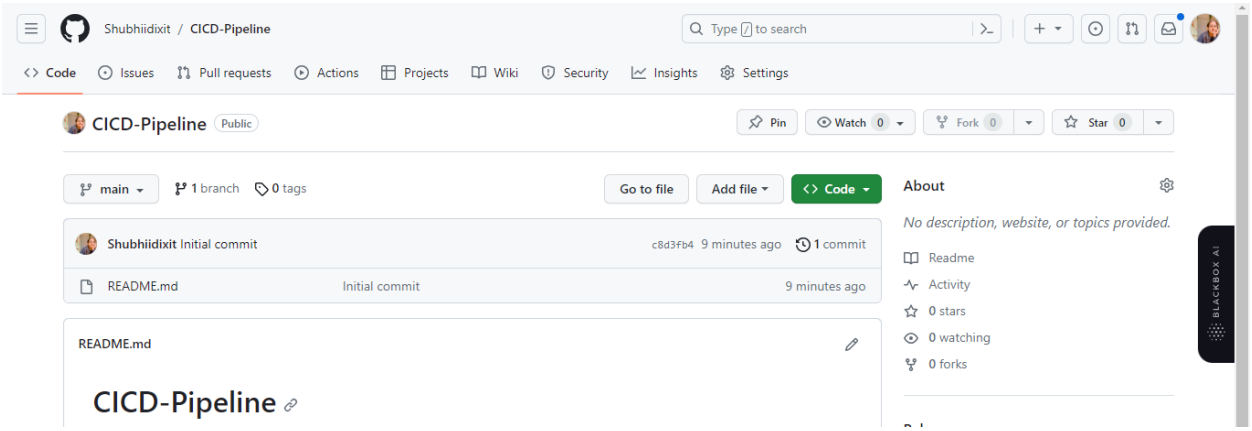
Step 2: Initialise an empty git repository in the terminal using the command:
git init

Check the status of your current files using the command:
git status



Step 3: Now before committing the changes, add the files to the staging area using the command:
git add .

Now commit the changes:
git commit -m “initial commit”
Step 3: Go to your github account and create a repository for the project.

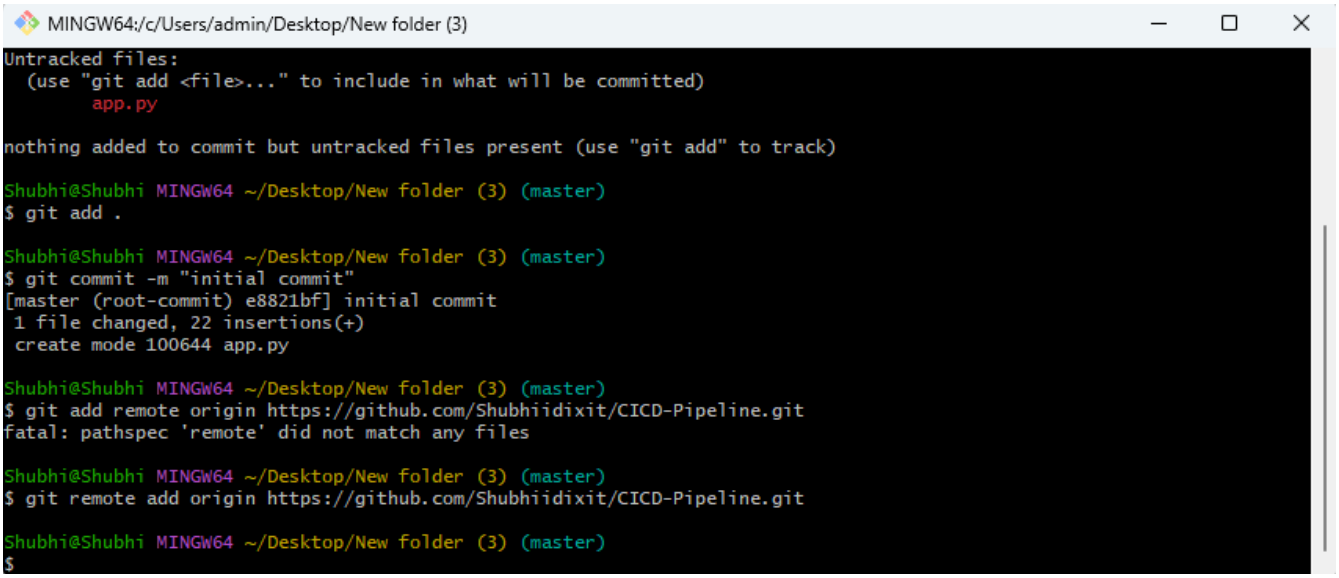


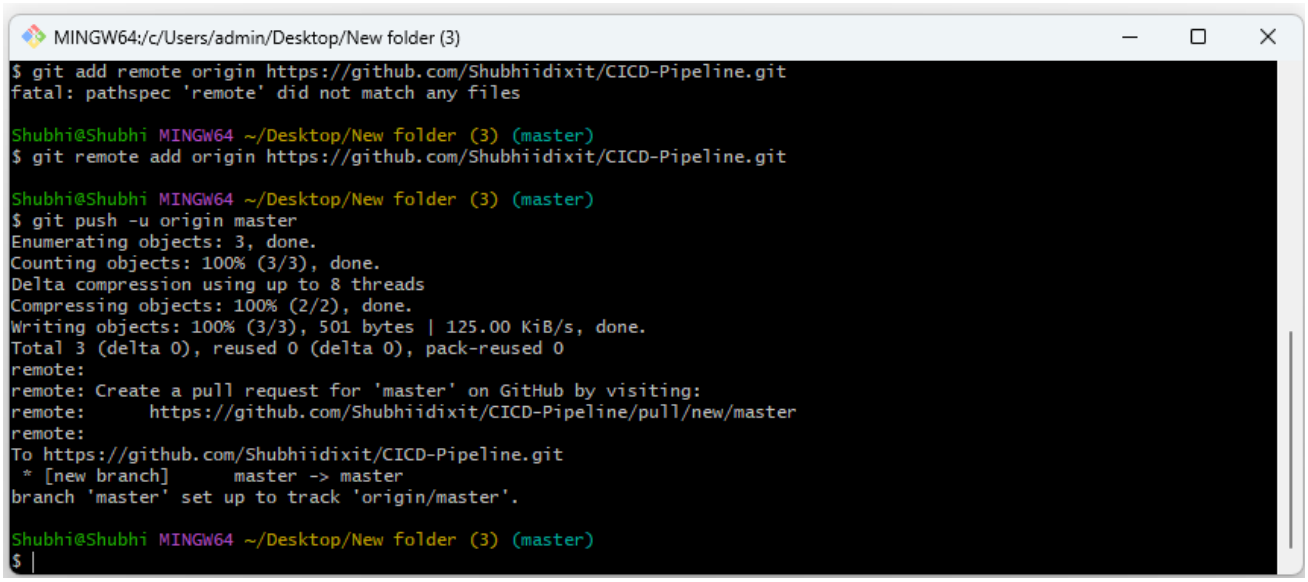
Step 4: After creating the repository, copy the repository url from the “**code**” section (highlighted in green) and go back to the git bash terminal.

Step 5: Run the command:
git remote add origin “paste the url that you copied”

Step 6: Push the code files to the github

git push -u origin master

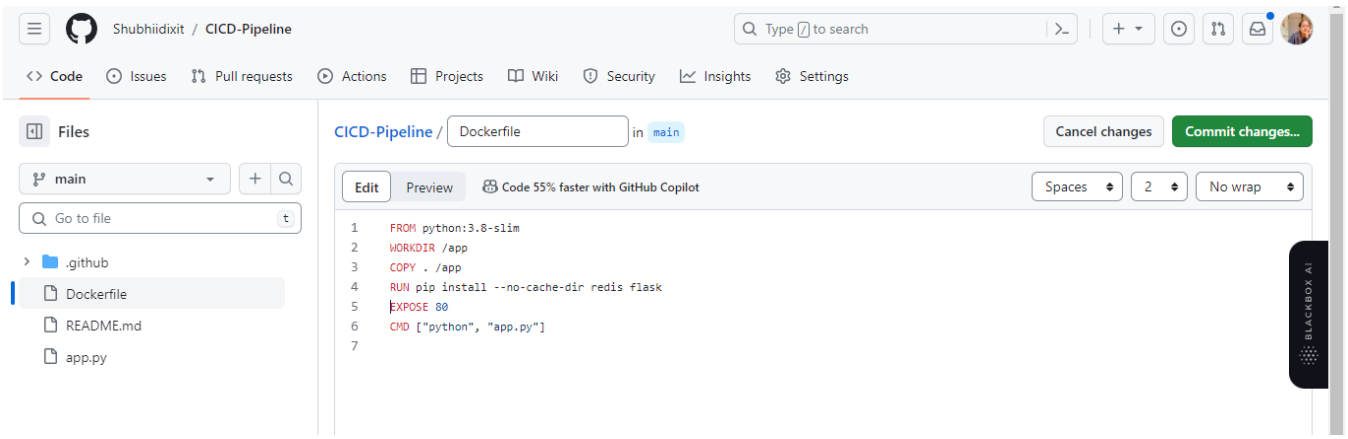


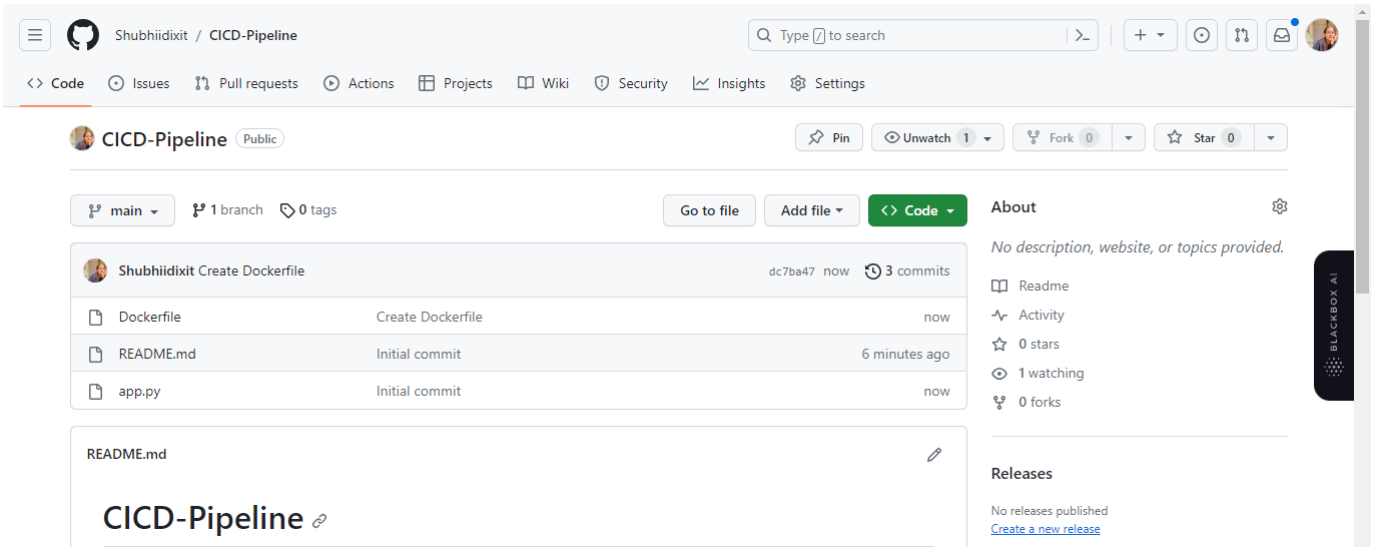


Go to your repository, refresh the page and check whether the files are uploaded or not.

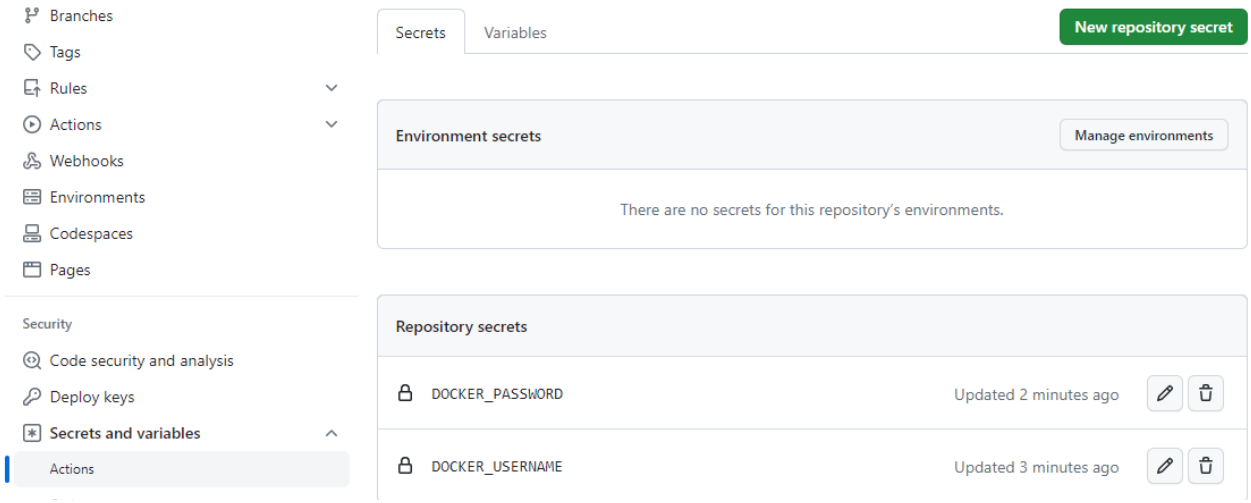
Step 7: Now go to “**Add file**” and then “**Create new file**” option
Create a dockerfile and include the following instructions:

```
FROM node:latest
WORKDIR /usr/src/app
COPY package.json ./
RUN npm install
COPY . .
```





After making the dockerfile commit changes in it and save.
Step 8: Go to "**Settings**" and go to "**Secret and Variables**" in the left panel.
Choose the "**Actions**" and then click on "**New Repository Secret**".
Add your docker hub username and passwords in the repository secret with the name:
DOCKER_USERNAME
DOCKER_PASSWORD

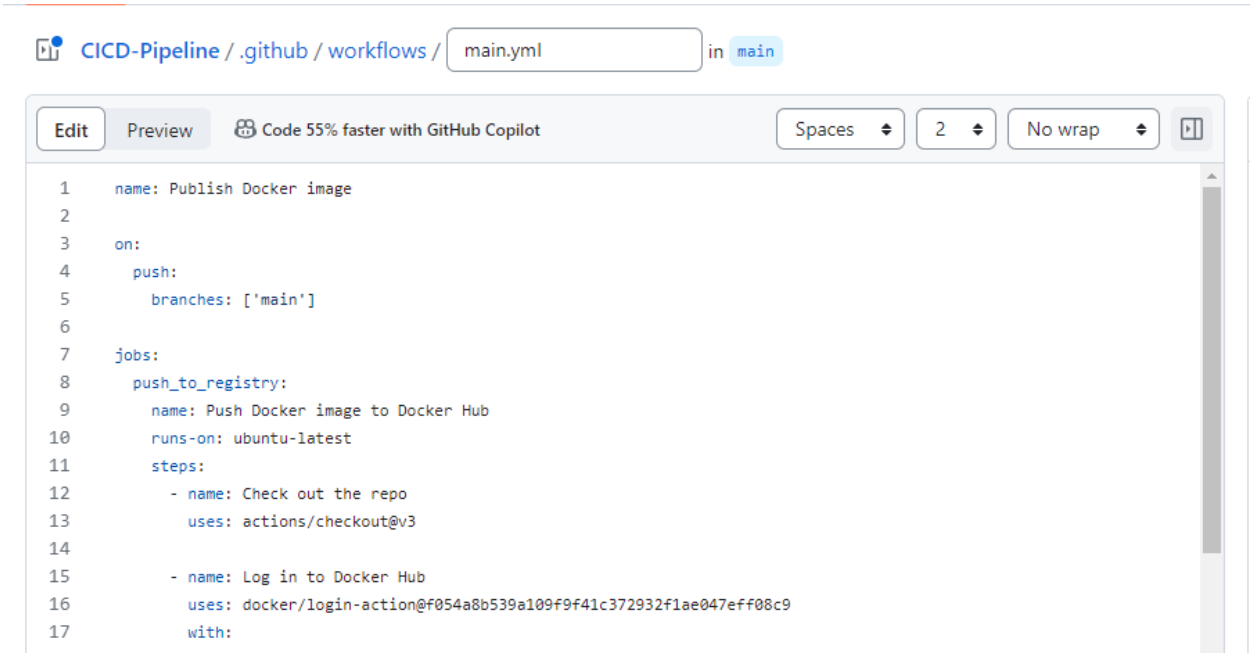


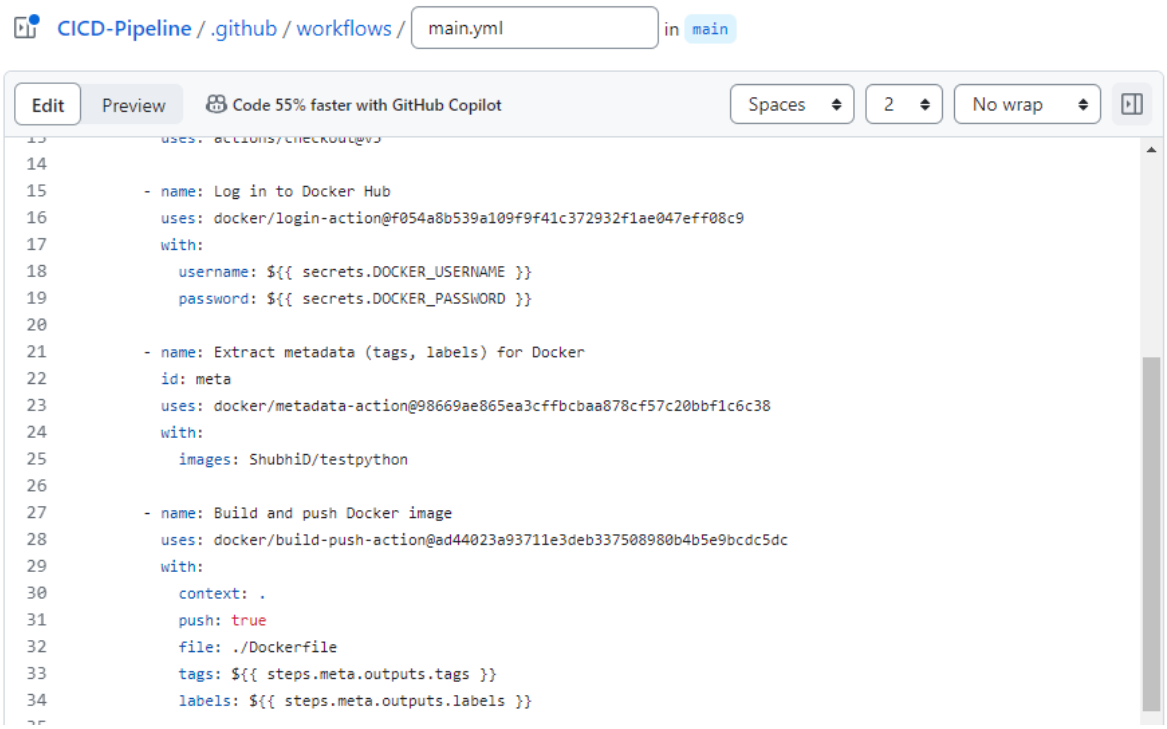
Step 9: Go to actions and click on "**set up a workflow yourself**" highlighted in blue at the top.

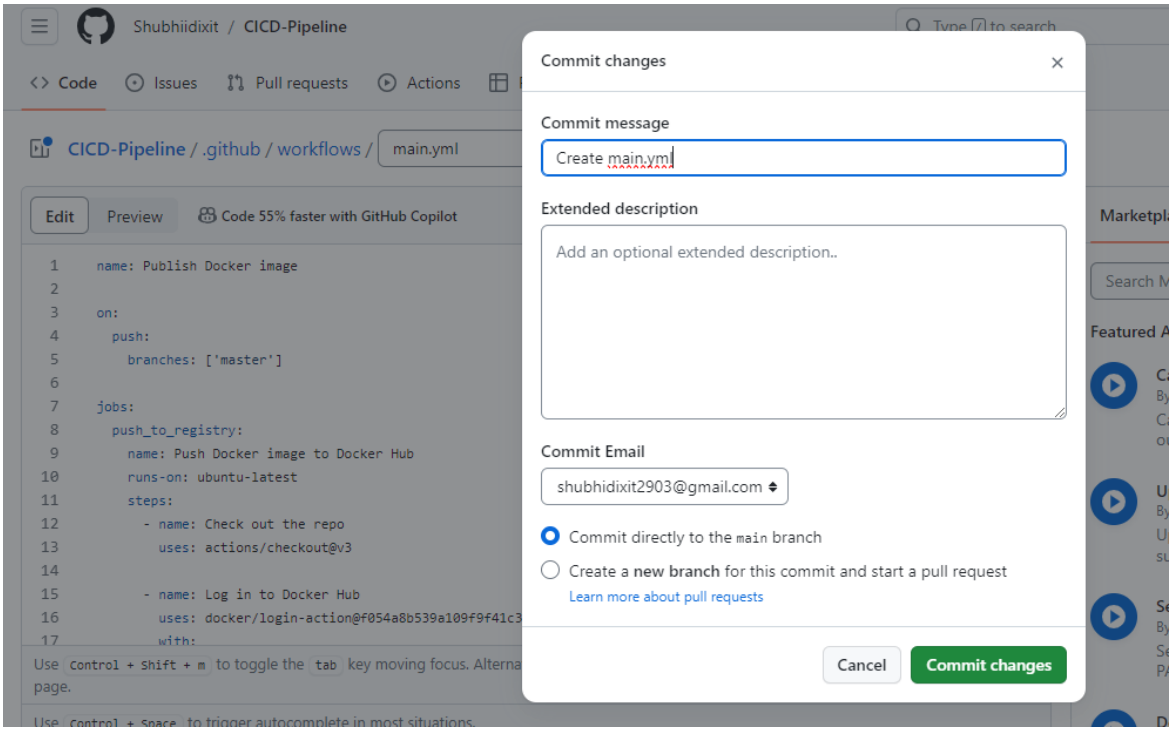
A **main.yaml** file will be opened. Inside the **main.yaml** file, copy the following code:

```
name: Publish Docker image


on:
  push:
 branches: ['master']

jobs:
 push_to_registry:
  name: Push Docker image to Docker Hub
 runs-on: ubuntu-latest
steps:
  - name: Check out the repo
    uses: actions/checkout@v3

 - name: Log in to Docker Hub
        uses: docker/login-action@f054a8b539a109f9f41c372932f1ae047eff08c9
        with:
          username: ${{ secrets.DOCKER_USERNAME }}
          password: ${{ secrets.DOCKER_PASSWORD }}

      - name: Extract metadata (tags, labels) for Docker
```

```
    id: meta
    uses: docker/metadata-action@98669ae865ea3cffbcbaa878cf57c20bbf1c6c38
    with:
      images: shubhid/testpython

  - name: Build and push Docker image
    uses: docker/build-push-action@ad44023a93711e3deb337508980b4b5e9bcdc5dc
    with:
      context: .
      push: true
      file: ./Dockerfile
      tags: ${{ steps.meta.outputs.tags }}
      labels: ${{ steps.meta.outputs.labels }}
```

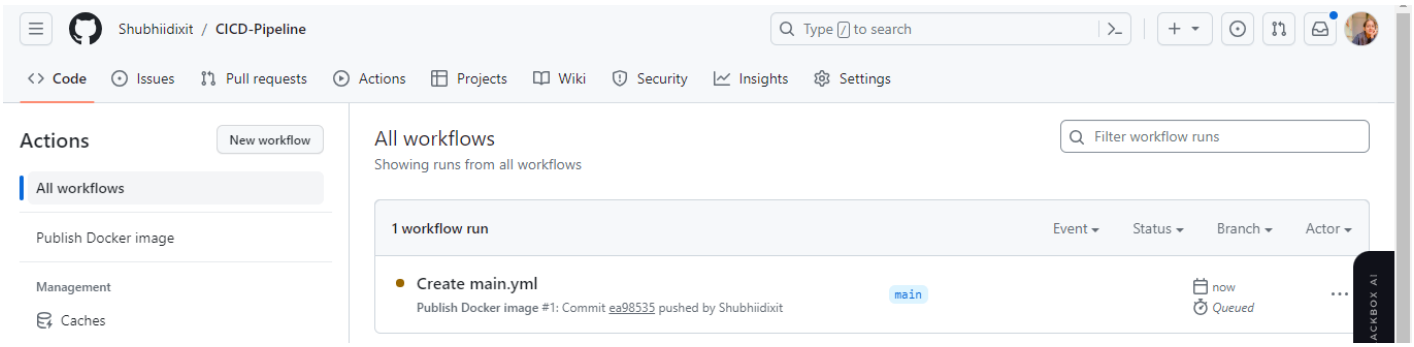


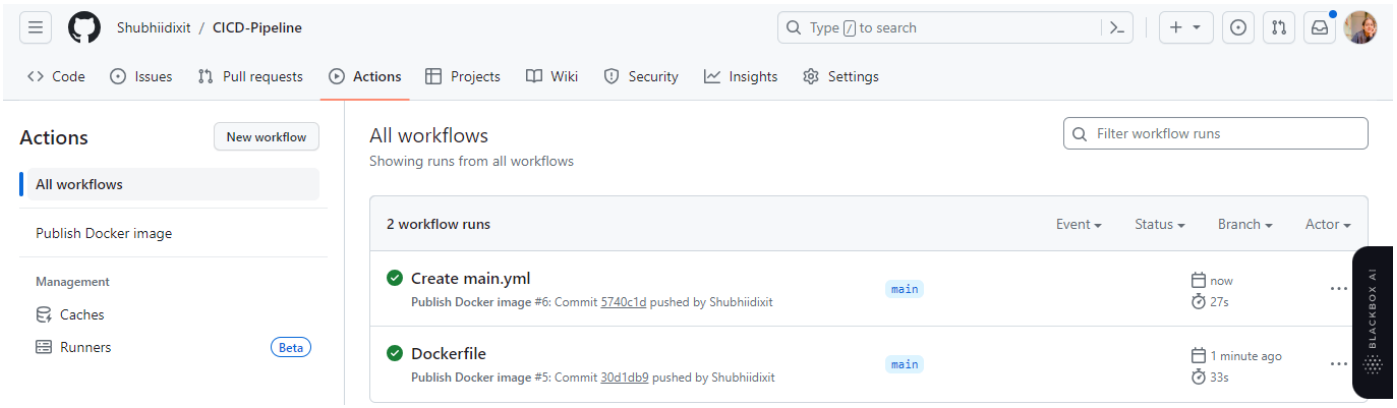




Step 10: After adding the code, commit the changes in the main.yaml file.

Step 11: Go to "**Actions**" and see whether the file is being created or not. If yes, then the instructions that are included in the yaml file will be executed and the image will be pushed to the docker hub.





Step 12: Login to your docker hub and check whether the image is pushed or not.