Configuring GitHub CLI

Steps to be followed:

1. Log in to GitHub.com.
2. Create a workflow to integrate with GitHub CLI.

Step 1: Log in to GitHub.com

1.1 Sign in to the GitHub portal at https://github.com

1.2 Navigate to your repository main page where we are going to create a workflow file.

Step 3: Create a workflow to integrate with GitHub CLI.

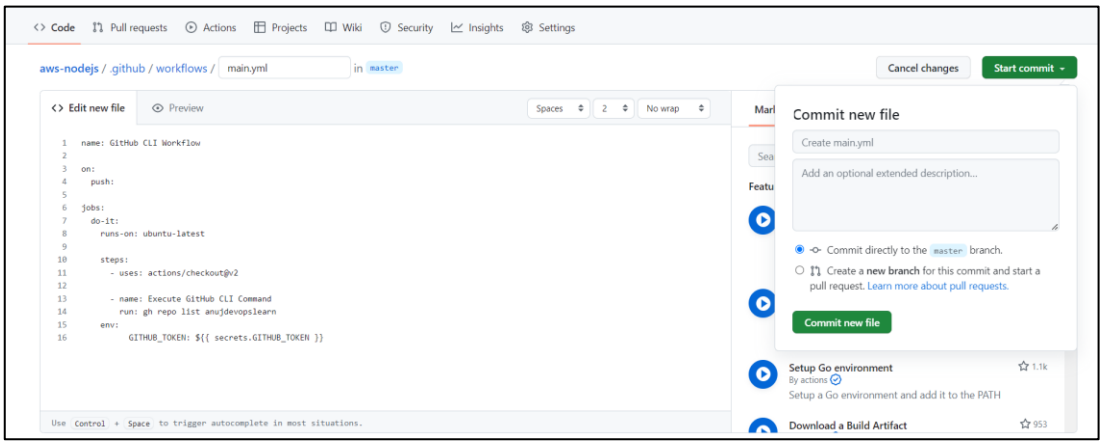
2.1 Next, we have to create a Docker workflow file to automate the CI process for building custom Docker images.

2.2 Create a .**github/workflows** directory in your repository in case the directory does not exist.

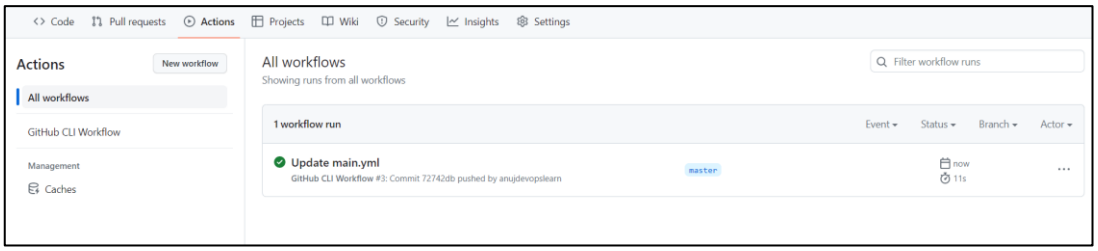
2.3 In the workflow directory, create a new workflow file with the name **github-cli.yml.**

```
name: GitHub CLI Workflow

on:
  push:

jobs:
  github-cli:
    runs-on: ubuntu-latest

    steps:
      - uses: actions/checkout@v2

      - name: Execute GitHub CLI Command
        run: gh repo list anujdevopslearn
        env:
          GITHUB_TOKEN: ${{ secrets.GITHUB_TOKEN }}
```

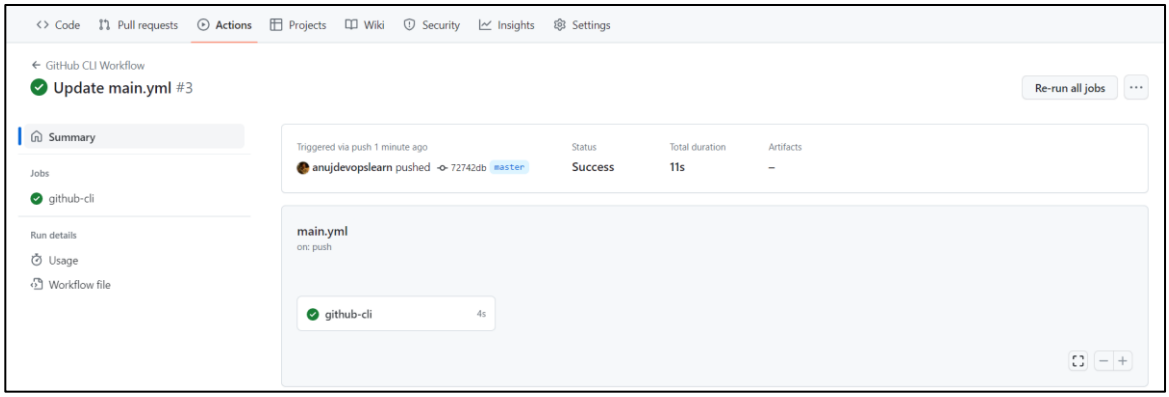
2.3 Once the file is created, click on **Commit new file** to save the workflow file in your code repository.

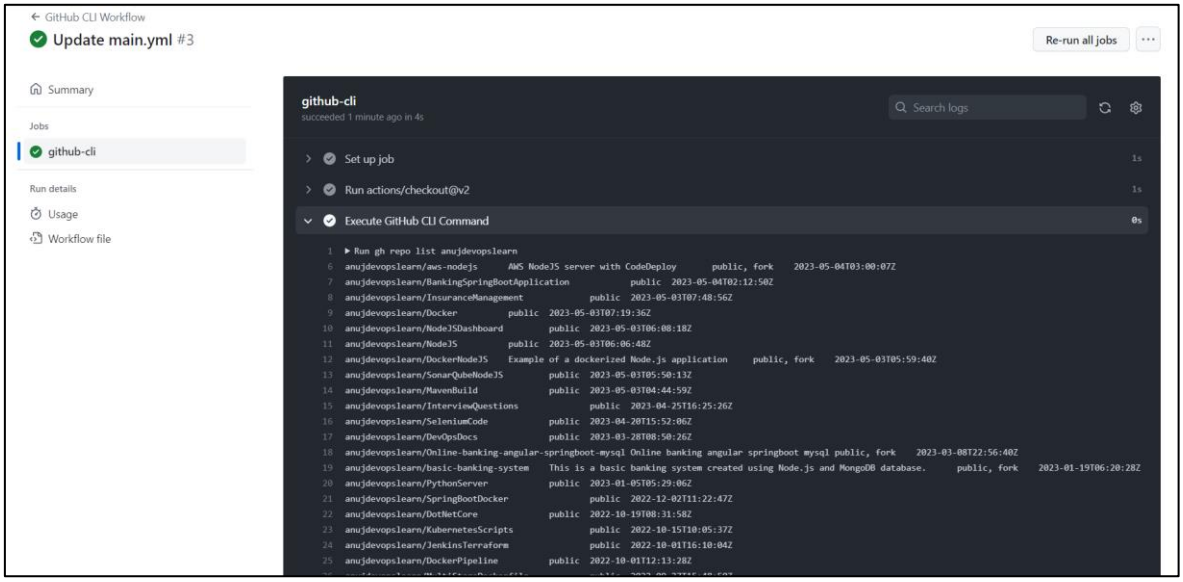


2.4 Next navigate to the **Actions** tab on your repository to access the workflow execution page as below:



2.5 Select the latest execution logs from GitHub actions and look for detailed execution logs as below:


Code
Pull requests
Actions
Projects
Wiki
Security
Insights
Settings
GitHub CLI Workflow
Update main.yml #3
Re-run all jobs
Summary
Jobs
github-cli
Run details
Usage
Workflow file
Triggered via push 1 minute ago
anujdevopslearn pushed 72742db master
Status
Success
Total duration
11s
Artifacts
–
main.yml
on: push
github-cli
4s


GitHub CLI Workflow
Update main.yml #3
Re-run all jobs
Summary
Jobs
github-cli
Run details
Usage
Workflow file
github-cli
succeeded 1 minute ago in 4s
Search logs
Set up job
1s
Run actions/checkout@v2
1s
Execute GitHub CLI Command
0s
▶ Run gh repo list anujdevopslearn
anujdevopslearn/aws-nodejs AWS NodeJS server with CodeDeploy public, fork 2023-05-04T03:00:07Z
anujdevopslearn/BankingSpringBootApplication public 2023-05-04T02:12:50Z
anujdevopslearn/InsuranceManagement public 2023-05-03T07:48:56Z
anujdevopslearn/Docker public 2023-05-03T07:19:36Z
anujdevopslearn/NodeJSDashboard public 2023-05-03T06:08:18Z
anujdevopslearn/NodeJS public 2023-05-03T06:06:48Z
anujdevopslearn/DockerNodeJS Example of a dockerized Node.js application public, fork 2023-05-03T05:59:40Z
anujdevopslearn/SonarQubeNodeJS public 2023-05-03T05:50:13Z
anujdevopslearn/MavenBuild public 2023-05-03T04:44:59Z
anujdevopslearn/InterviewQuestions public 2023-04-25T16:25:26Z
anujdevopslearn/SeleniumCode public 2023-04-20T15:52:06Z
anujdevopslearn/DevOpsDocs public 2023-03-28T08:50:26Z
anujdevopslearn/Online-banking-angular-springboot-mysql Online banking angular springboot mysql public, fork 2023-03-08T22:56:40Z
anujdevopslearn/basic-banking-system This is a basic banking system created using Node.js and MongoDB database. public, fork 2023-01-19T06:20:28Z
anujdevopslearn/PythonServer public 2023-01-05T05:29:06Z
anujdevopslearn/SpringBootDocker public 2022-12-02T11:22:47Z
anujdevopslearn/DotNetCore public 2022-10-19T08:31:58Z
anujdevopslearn/KubernetesScripts public 2022-10-15T10:05:37Z
anujdevopslearn/JenkinsTerraform public 2022-10-01T16:10:04Z
anujdevopslearn/DockerPipeline public 2022-10-01T12:13:28Z