**GitHub Actions**

**What is it?**

* GitHub Actions is a built-in automation tool within GitHub that lets you create custom automated workflows for your code.

It automates tasks based on events in your repository, like:

* **Code changes:** When you push new code, it can run tests, build your application, or deploy it automatically.
* **Pull requests:** When someone proposes changes, it can review the code, run tests, and provide feedback.
* **Issues:** When someone reports a problem, it can automatically trigger tasks to fix it.

**How it works:**

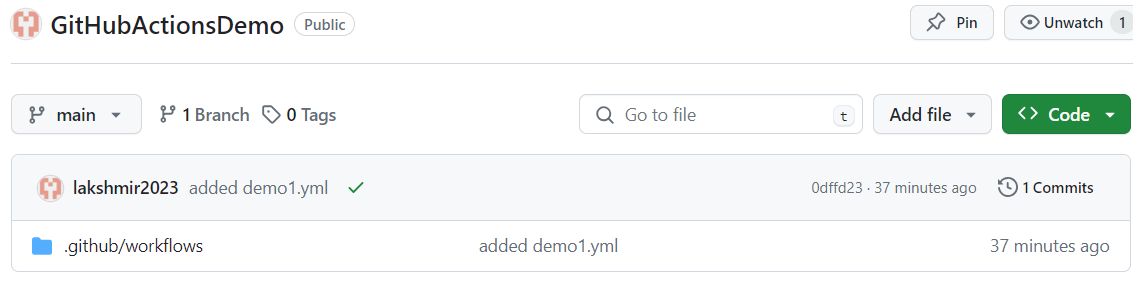
1. **Define your workflow:** You write a YAML file specifying the tasks you want to automate, called a "workflow." This file tells the robot what to do and when.
2. **Events Trigger:** These workflows can be triggered by various events, like pushing code, opening a pull request, or creating an issue.
3. **Jobs & Steps:** Tasks are divided into Jobs (think specific stages) and further into Steps (individual actions).
4. **Execution:** GitHub runs these steps on virtual machines, following your instructions within the workflow.
5. **Results & Feedback:** You can monitor the progress, view logs, and analyze results to understand how your code behaves.

**Where You Can Use GitHub Actions:**

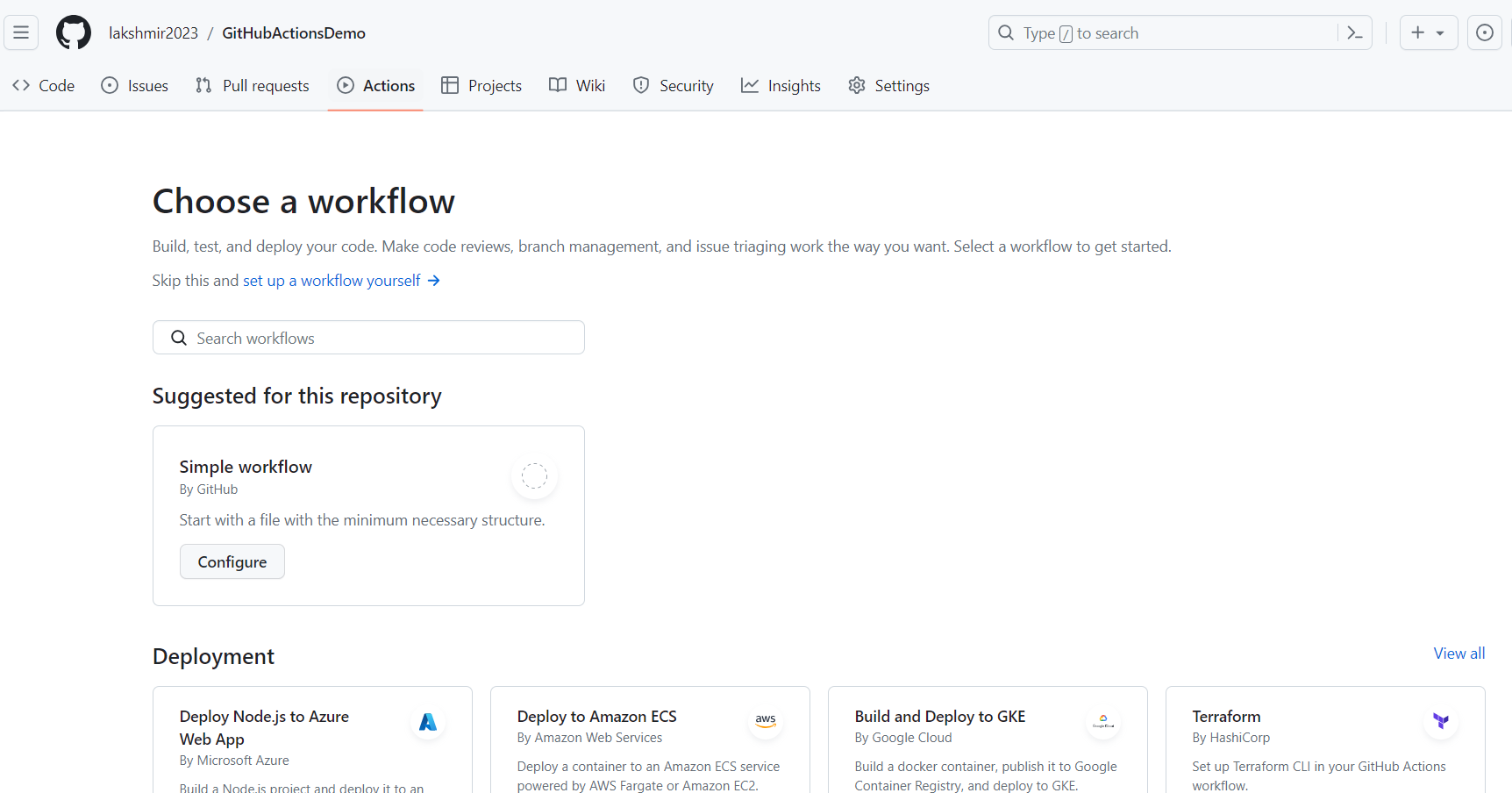
* **Continuous Integration (CI):** Automate tasks like building and testing your code upon every push or pull request, ensuring code quality and catching issues early.
* **Continuous Delivery (CD):**Automatically deploy your code to production environments once it passes CI, streamlining the release process.
* **Code Review and Analysis:** Integrate code review tools and static analysis checks into your workflow for improved code quality and security.
* **Infrastructure Management:** Automate tasks like provisioning and scaling infrastructure for your applications.
* **Project Management:** Trigger workflows based on issue creation, project board updates, or other project management events.

**Hands-on Demo:**

1. **Create a Repository:** Choose a name and description for your project.
2. **Action Folder:** Inside your repository, create a folder named .github/workflows.

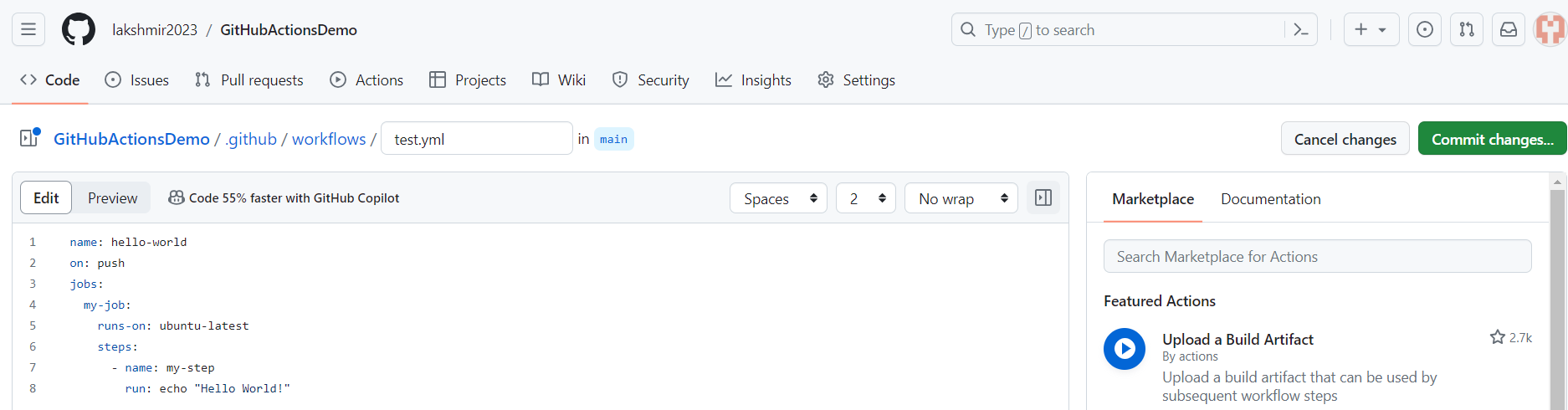


1. **Workflow File:** In this folder, create a new YAML file (e.g., main.yml).

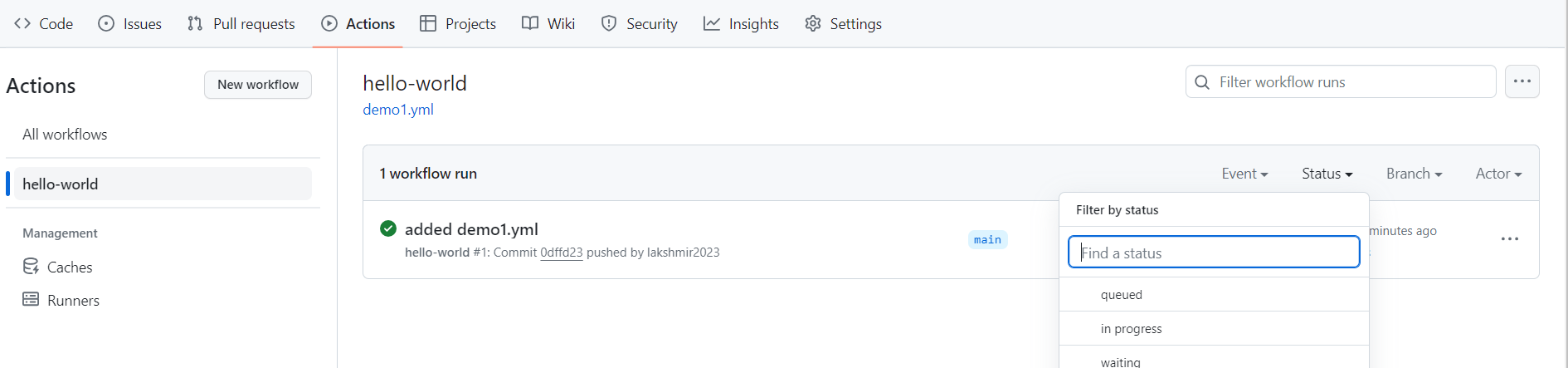


Select setup a workflow by yourself and name it.

1. **Write Your Workflow:** Paste the desired actions into the YAML file   
   (Browse github actions yaml to print hello world)

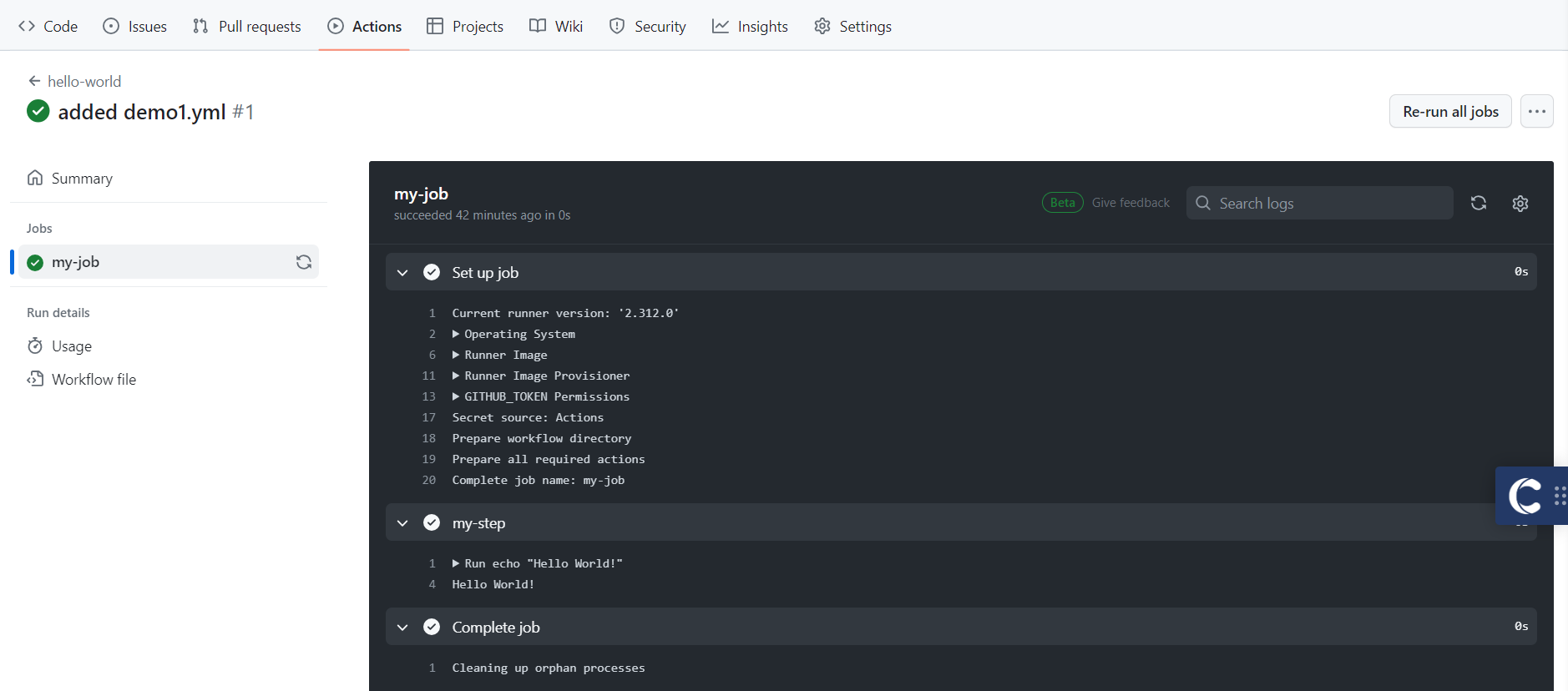


1. **Commit and Push:** Commit your changes and push them to your remote repository.
2. **View the Action:**Go to your repository's "Actions" tab and select the workflow name.



Able to filter our workflow runs based on event, status, branch, actor.

1. **Check Results:** Monitor the workflow's progress and results in the logs.



**Terms Explained:**

* **Workflow:** A collection of jobs defined in a YAML file, representing your desired workflow.
* **Events:** Triggers that initiate your workflow, like pushing code or opening a pull request.
* **Jobs:** Individual tasks within your workflow, each performing a specific action.
* **Steps:** The specific commands or actions executed within each job.
* **Chain Jobs**: Use :needs to specify that one job must finish before another starts.