

# Lab: GitHub Actions / Workflows

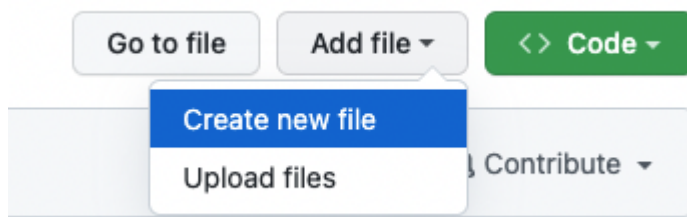
## Introduction

You only need a GitHub repository to create and run a GitHub Actions workflow. In this guide, you'll add a workflow that demonstrates some of the essential features of GitHub Actions.

The following example shows you how GitHub Actions jobs can be automatically triggered, where they run, and how they can interact with the code in your repository.

## Creating your first workflow

1. Create a `.github/workflows` directory in your repository on GitHub if this directory does not already exist.
2. In the `.github/workflows` directory, create a file named `github-actions-demo.yml`.
  - Select the **Add file** drop-down and click **Create new file**.

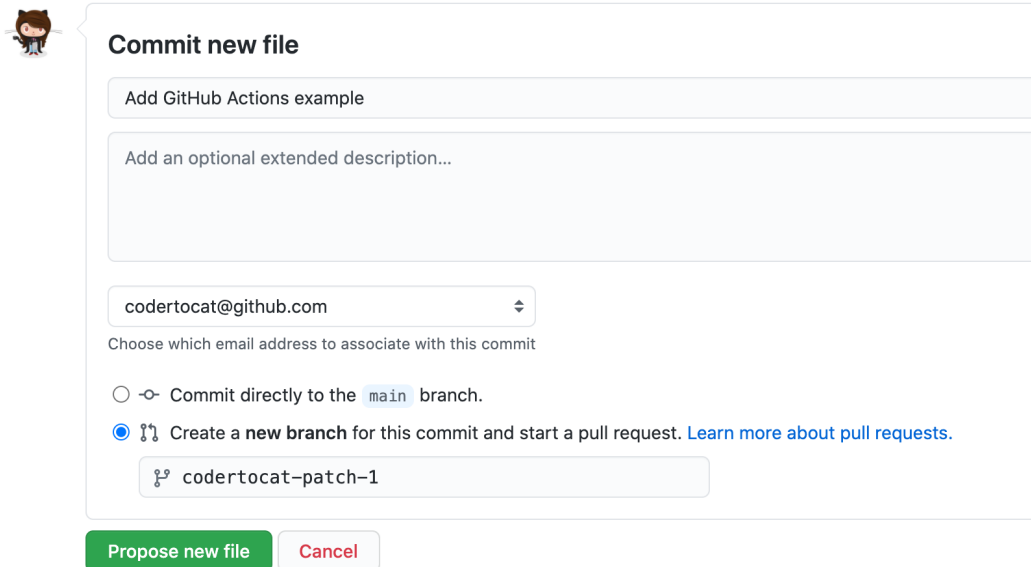


- In the **Name your file...** field, enter `.github/workflows/github-actions-demo.yml`.
3. Copy the following YAML contents into the `github-actions-demo.yml` file:

```
name: GitHub Actions Demo
run-name: ${ github.actor } is testing out GitHub Actions 🚀
on: [push]
jobs:
  Explore-GitHub-Actions:
    runs-on: ubuntu-latest
    steps:
      - run: echo "🎉 The job was automatically triggered by a ${ github.event_name } event."
      - run: echo "👤 This job is now running on a ${ runner.os } server hosted by GitHub!"
      - run: echo "🔍 The name of your branch is ${ github.ref } and your repository is ${ github.repository }."
      - name: Check out repository code
        uses: actions/checkout@v3
      - run: echo "💡 The ${ github.repository } repository has been cloned to the runner."
      - run: echo "🖨 The workflow is now ready to test your code on the runner."
```

```
- name: List files in the repository
  run: |
    ls ${ github.workspace }
- run: echo "🐱 This job's status is ${ job.status }."
```

4. Scroll to the bottom of the page and select **Create a new branch for this commit and start a pull request**. Then, to create a pull request, click **Propose new file**.



**Commit new file**

Add GitHub Actions example

Add an optional extended description...

codertocat@github.com

Choose which email address to associate with this commit

☐ Commit directly to the `main` branch.

☒ Create a new branch for this commit and start a pull request. [Learn more about pull requests.](#)

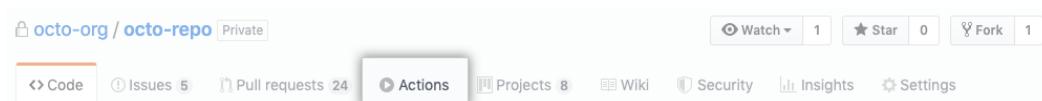
`codertocat-patch-1`

Propose new file Cancel

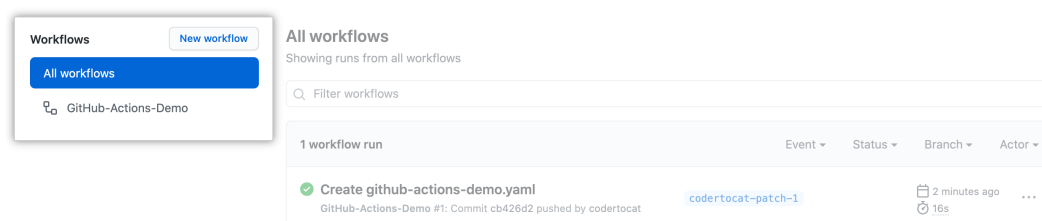
Committing the workflow file to a branch in your repository triggers the `push` event and runs your workflow.

## Viewing your workflow results

1. On GitHub.com, navigate to the main page of the repository.
2. Under your repository name, click **Actions**.



3. In the left sidebar, click the workflow you want to see.



**Workflows** [New workflow](#)

All workflows

GitHub-Actions-Demo

**All workflows**

Showing runs from all workflows

Filter workflows

1 workflow run	Event	Status	Branch	Actor
<div> <div>✓</div> <div>Create github-actions-demo.yaml</div> </div> <div>GitHub-Actions-Demo #1: Commit cb426d2 pushed by codertocat</div>			codertocat-patch-1	2 minutes ago 16s

4. From the list of workflow runs, click the name of the run you want to see.

Workflows New workflow

All workflows

GitHub-Actions-Demo

Filter workflows

1 workflow run

Event Status Branch Actor

Create github-actions-demo.yml  
 GitHub-Actions-Demo #1: Commit cb426d2 pushed by codertocat  
 codertocat-patch-1  
 36 minutes ago  
 16s

5. Under **Jobs**, click the **Explore-GitHub-Actions** job.

Summary

Jobs

Explore-GitHub-Actions

Triggered via push 11 minutes ago

codertocat pushed -> cb426d2 [codertocat-patch-1](#)

Status: Success

Total duration: 19s

Billable time: 3s

github-actions-demo.yml

on: push

Explore-GitHub-Actions 4s

6. The log shows you how each of the steps was processed. Expand any of the steps to view its details.

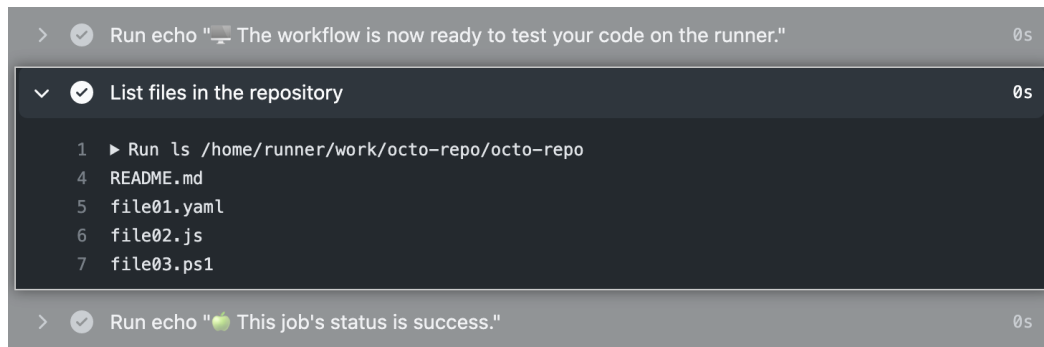
Explore-GitHub-Actions

succeeded 2 minutes ago in 6s

Search logs

- > Set up job 3s
- > Run echo "🔥 The job was automatically triggered by a push event." 0s
- > Run echo "🌐 This job is now running on a Linux server hosted by GitHub!" 0s
- > Run echo "📁 The name of your branch is refs/heads/codertocat-patch-1 and your repository is codertocat/octo-repo." 0s
- > Check out repository code 3s
- > Run echo "💡 The codertocat/octo-repo repository has been cloned to the runner." 0s
- > Run echo "🚀 The workflow is now ready to test your code on the runner." 0s
- > List files in the repository 0s
- > Run echo "🍏 This job's status is success." 0s
- > Post Check out repository code 0s
- > Complete job 0s

For example, you can see the list of files in your repository:



The screenshot shows a GitHub Actions workflow run interface. At the top, a step titled 'Run echo "🚀 The workflow is now ready to test your code on the runner."' is completed, with a duration of 0s. Below it, a step titled 'List files in the repository' is also completed, with a duration of 0s. The output of this step is displayed in a dark-themed terminal window, showing the command 'Run ls /home/runner/work/octo-repo/octo-repo' and its output: 'README.md', 'file01.yaml', 'file02.js', and 'file03.ps1'. At the bottom, another step titled 'Run echo "🍏 This job's status is success."' is shown, also with a duration of 0s.

```
> ✓ Run echo "🚀 The workflow is now ready to test your code on the runner." 0s

✓ List files in the repository 0s

1 ▶ Run ls /home/runner/work/octo-repo/octo-repo
4 README.md
5 file01.yaml
6 file02.js
7 file03.ps1

> ✓ Run echo "🍏 This job's status is success." 0s
```

The example workflow you just added is triggered each time code is pushed to the branch, and shows you how GitHub Actions can work with the contents of your repository.