

**This version of GitHub Enterprise was discontinued on 2023-03-15.** No patch releases will be made, even for critical security issues. For better performance, improved security, and new features, [upgrade to the latest version of GitHub Enterprise](#). For help with the upgrade, [contact GitHub Enterprise support](#).

# Adding labels to issues

## In this article

- Introduction
- Creating the workflow
- Testing the workflow
- Next steps

You can use GitHub Actions to automatically label issues.

**Note:** GitHub-hosted runners are not currently supported on GitHub Enterprise Server. You can see more information about planned future support on the [GitHub public roadmap](#).

## Introduction

This tutorial demonstrates how to use the `actions/github-script` [action](#) in a workflow to label newly opened or reopened issues. For example, you can add the `triage` label every time an issue is opened or reopened. Then, you can see all issues that need to be triaged by filtering for issues with the `triage` label.

The `actions/github-script` action allows you to easily use the GitHub API in a workflow.

In the tutorial, you will first make a workflow file that uses the `actions/github-script` [action](#). Then, you will customize the workflow to suit your needs.

## Creating the workflow

- 1 Choose a repository where you want to apply this project management workflow. You can use an existing repository that you have write access to, or you can create a new repository. For more information about creating a repository, see "[Creating a new repository](#)."
- 2 In your repository, create a file called `.github/workflows/YOUR_WORKFLOW.yml`, replacing `YOUR_WORKFLOW` with a name of your choice. This is a workflow file. For more information about creating new files on GitHub, see "[Creating new files](#)."
- 3 Copy the following YAML contents into your workflow file.

YAML



```
name: Label issues
on:
  issues:
```

```

types:
  - reopened
  - opened
jobs:
  label_issues:
    runs-on: ubuntu-latest
    permissions:
      issues: write
    steps:
      - uses: actions/github-script@v5
        with:
          script: |
            github.rest.issues.addLabels({
              issue_number: context.issue.number,
              owner: context.repo.owner,
              repo: context.repo.repo,
              labels: ["triage"]
            })

```

#### 4 Customize the `script` parameter in your workflow file:

- The `issue_number`, `owner`, and `repo` values are automatically set using the `context` object. You do not need to change these.
- Change the value for `labels` to the list of labels that you want to add to the issue. Separate multiple labels with commas. For example, `["help wanted", "good first issue"]`. For more information about labels, see "[Managing labels](#)."

#### 5 Commit your workflow file to the default branch of your repository. For more information, see "[Creating new files](#)."

## Testing the workflow

Every time an issue in your repository is opened or reopened, this workflow will add the labels that you specified to the issue.

Test out your workflow by creating an issue in your repository.

- 1 Create an issue in your repository. For more information, see "[Creating an issue](#)."
- 2 To see the workflow run that was triggered by creating the issue, view the history of your workflow runs. For more information, see "[Viewing workflow run history](#)."
- 3 When the workflow completes, the issue that you created should have the specified labels added.

## Next steps

- To learn more about additional things you can do with the `actions/github-script` action, see the [actions/github-script action documentation](#).
- To learn more about different events that can trigger your workflow, see "[Events that trigger workflows](#)."
- [Search GitHub](#) for examples of workflows using this action.

### Legal

