

Skipping workflow runs

You can skip workflow runs triggered by the `push` and `pull_request` events by including a command in your commit message.

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](#).

Note: If a workflow is skipped due to [path filtering](#), [branch filtering](#) or a commit message (see below), then checks associated with that workflow will remain in a "Pending" state. A pull request that requires those checks to be successful will be blocked from merging.

Workflows that would otherwise be triggered using `on: push` or `on: pull_request` won't be triggered if you add any of the following strings to the commit message in a push, or the HEAD commit of a pull request:

- `[skip ci]`
- `[ci skip]`
- `[no ci]`
- `[skip actions]`
- `[actions skip]`

Alternatively, you can end the commit message with two empty lines followed by either:

- `skip-checks:true`
- `skip-checks: true`

You won't be able to merge the pull request if your repository is configured to require specific checks to pass first. To allow the pull request to be merged you can push a new commit to the pull request without the skip instruction in the commit message.

Note: Skip instructions only apply to the `push` and `pull_request` events. For example, adding `[skip ci]` to a commit message won't stop a workflow that's triggered `on: pull_request_target` from running.

Skip instructions only apply to the workflow run(s) that would be triggered by the commit that contains the skip instructions. You can also disable a workflow from running. For more information, see "[Disabling and enabling a workflow](#)."

Legal