

Setting exit codes for actions

Create an action
7 of 7 in learning path

More guides →

In this article

- About exit codes
- Setting a failure exit code in a JavaScript action
- Setting a failure exit code in a Docker container action

You can use exit codes to set the status of an action. GitHub displays statuses to indicate passing or failing actions.

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](#).

About exit codes [↗](#)

GitHub uses the exit code to set the action's check run status, which can be `success` or `failure`.

Exit status	Check run status	Description
0	<code>success</code>	The action completed successfully and other tasks that depend on it can begin.
Nonzero value (any integer but 0)	<code>failure</code>	Any other exit code indicates the action failed. When an action fails, all concurrent actions are canceled and future actions are skipped. The check run and check suite both get a <code>failure</code> status.

Setting a failure exit code in a JavaScript action [↗](#)

If you are creating a JavaScript action, you can use the actions toolkit `@actions/core` package to log a message and set a failure exit code. For example:

```
try {
  // something
} catch (error) {
  core.setFailed(error.message);
}
```

For more information, see "[Creating a JavaScript action](#)."

Setting a failure exit code in a Docker container action

If you are creating a Docker container action, you can set a failure exit code in your `entrypoint.sh` script. For example:

```
if <condition> ; then
  echo "Game over!"
  exit 1
fi
```

For more information, see "[Creating a Docker container action](#)."

Previous

[Dockerfile support for GitHub Actions](#)

Legal

© 2023 GitHub, Inc. [Terms](#) [Privacy](#) [Status](#) [Pricing](#) [Expert services](#) [Blog](#)