GitHub Packages documentation

Learn to safely publish and consume packages, store your packages alongside your code, and share your packages privately with your team or publicly with the open source community. You can also automate your packages with GitHub Actions.

[Quickstart] [Reference]

Start here

Learn GitHub Packages

You can find out more about using packages in GitHub, including publishing new packages to GitHub Packages, view and install existing packages, and, in special circumstances, delete existing packages.

Managing GitHub packages using GitHub Actions workflows

You can safely publish and consume packages by building custom workflows that can also build, test, and deploy your code.

Installing a package

You can install a package from GitHub Packages and use the package as a dependency in your own project.

Popular

Working with the npm registry

You can configure npm to publish packages to GitHub Packages and to use packages stored on GitHub Packages as dependencies in an npm project.

Migrating to the Container registry from the Docker registry

An enterprise owner can migrate Docker images previously stored in the Docker registry on your GitHub Enterprise Server instance to the Container registry.

Learn GitHub Packages

You can find out more about using packages in GitHub, including publishing new packages to GitHub Packages, view and install existing packages, and, in special circumstances, delete existing packages.

Working with the Apache Maven registry

You can configure Apache Maven to publish packages to GitHub Packages and to use packages stored on GitHub Packages as dependencies in a Java project.

What's new   View all →

Changes to token permission on packages
September 26

Fine-grained permissions and organization-level publishing are now available for the GitHub Packages RubyGems registry
January 10

Fine-grained permissions and organization-level publishing are now available for the GitHub Packages NuGet registry
December 09

Guides 🔗

Migrating to the Container registry from the Docker registry

An enterprise owner can migrate Docker images previously stored in the Docker registry on your GitHub Enterprise Server instance to the Container registry.

@GitHub

Working with the Container registry

You can store and manage Docker and OCI images in the Container registry, which uses the package namespace https://containers.HOSTNAME.

@GitHub

Working with the RubyGems registry

You can configure RubyGems to publish a package to GitHub Packages and to use packages stored on GitHub Packages as dependencies in a Ruby project with Bundler.

@GitHub

All GitHub Packages docs 🔗

Learn GitHub Packages

[Introduction to GitHub Packages](#)

[About permissions for GitHub Packages](#)

[Configuring a package's access control and visibility](#)

[Connecting a repository to a package](#)

[Publishing a package](#)

[Viewing packages](#)

[Installing a package](#)

[Deleting and restoring a package](#)

Working with a GitHub Packages registry

[Working with the Container registry](#)

[Working with the Docker registry](#)

[Working with the RubyGems registry](#)

[Working with the npm registry](#)

[Working with the Apache Maven registry](#)

[Working with the Gradle registry](#)

[Working with the NuGet registry](#)

[Migrating to the Container registry from the Docker registry](#)

Managing GitHub packages using GitHub Actions workflows

[Publishing and installing a package with GitHub Actions](#)

[Example workflows for publishing a package](#)

[About GitHub Packages and GitHub Actions](#)