

Coverage Analytics

Codecov analyzes your [coverage reports](#) to help you identify untested code and improve test effectiveness. Before integrating with Codecov, ensure your project generates coverage reports, as Codecov relies on these reports for coverage analysis.

Select a setup option

Using GitHub Actions

Using Circle CI

Using Codecov's CLI

Step 1: Output a Coverage report file in your CI

Select your language below to generate your coverage reports. If your language isn't listed, visit our [supported languages doc](#) for example repositories. Codecov generally supports XML and JSON formats.

Pytest

Install requirements in your terminal:

pip install pytest pytest-cov

In a GitHub Action, run tests and generate a coverage report:

pytest --cov --cov-branch --cov-report=xml

Step 2: Select an upload token to add as a secret on GitHub

Repository token

Use it for uploading coverage reports in the csfrace-scrape-back repository.

Step 3: add token as repository secret

Codecov requires a token to authenticate uploading your coverage reports. **Organization admin** required to access organization settings > secrets and variables > actions

CODECOV_TOKEN

e5843ec9-45a7-476d-a447-378f82d75bcc

Your organization secret in GitHub should look like this:

Step 4: Add Codecov to your GitHub Actions workflow yaml file

After tests run, this will upload your coverage report to Codecov:

- name: Upload coverage reports to Codecov
 uses: codecov/codecov-action@v5
 with:
 token: \${{ secrets.CODECOV_TOKEN }}
 slug: zachatkinson/csfrace-scrape-back

Your final GitHub Actions workflow for a project using Pytest could look something like this:

name: Run tests and upload coverage

on:
 push

jobs:
 test:
 name: Run tests and collect coverage
 runs-on: ubuntu-latest
 steps:
 - name: Checkout
 uses: actions/checkout@v4
 with:
 fetch-depth: 2

 - name: Set up Python
 uses: actions/setup-python@v4

 - name: Install dependencies
 run: pip install pytest pytest-cov

 - name: Run tests
 run: pytest --cov --cov-branch --cov-report=xml

 - name: Upload results to Codecov
 uses: codecov/codecov-action@v5
 with:
 token: \${{ secrets.CODECOV_TOKEN }}
 slug: zachatkinson/csfrace-scrape-back

[Learn more](#) about generating coverage reports with Pytest.

Step 5: merge to main or your preferred feature branch

Once merged to your default branch, subsequent pull requests will have Codecov checks and comments. Additionally, you'll find your repo coverage dashboard here. If you have merged, try reloading the page.

How was your setup experience? Let us know in [this issue](#)

Visit our guide to [learn more](#) about coverage integration.