

How to integrate unit tests in my Code to be integrated into the Build

Using GitHub Actions

GitHub Actions is considered a good CI/CD (Continuous Integration/Continuous Deployment) tool because it is tightly integrated into the GitHub platform. This integration makes it easy to set up and configure CI/CD workflows directly within my GitHub repository, without the need to use external CI tools.

1. Create Unit Tests:

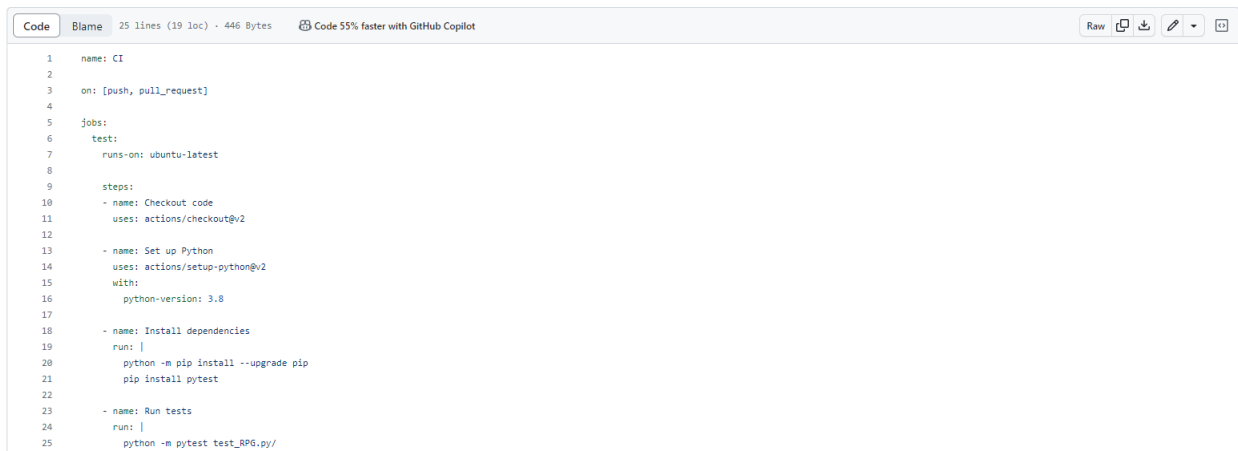
Writing unit tests for my code, and ensure that my unit tests are placed in a separate directory.

2. Configure GitHub Actions:

In my GitHub repository, inside the `.github/workflows` directory create a YAML file to define my GitHub Actions workflow. (`ci.yml`)

3. Define Workflow:

In the YAML file, define the workflow to run my unit tests.



```
1 name: CI
2
3 on: [push, pull_request]
4
5 jobs:
6   test:
7     runs-on: ubuntu-latest
8
9     steps:
10      - name: Checkout code
11        uses: actions/checkout@v2
12
13      - name: Set up Python
14        uses: actions/setup-python@v2
15        with:
16          python-version: 3.8
17
18      - name: Install dependencies
19        run: |
20          python -m pip install --upgrade pip
21          pip install pytest
22
23      - name: Run tests
24        run: |
25          python -m pytest test_RPG.py/
```

The above workflow triggers on pushes and pull requests, checks out the code, sets up Python, installs dependencies, and runs the tests using pytest.

By this, my unit tests are automatically executed whenever changes are pushed to my GitHub repository. This ensures that my tests are an integral part of the build and continuous integration process.

The screenshot displays the GitHub Actions interface for a repository named 'RandomPass' by user 'Mahshad89'. The 'Actions' tab is selected, showing a list of workflow runs for the 'Update ci.yml' workflow. The interface includes a sidebar with navigation options like 'Code', 'Issues', 'Pull requests', 'Actions', 'Projects', 'Wiki', 'Security', 'Insights', and 'Settings'. The main content area shows a table of workflow runs with columns for Event, Status, Branch, and Actor. The table lists four runs, all triggered by 'push' events on the 'main' branch. The first two runs are successful, while the third is failed and the fourth is successful. Below the table, there is a section for the selected workflow run, 'Update ci.yml #7', which shows a summary of the run, including the trigger event, status, total duration, and artifacts. The run details section shows a single job named 'test' with a duration of 9s.

Actions

New workflow

All workflows

Showing runs from all workflows

Filter workflow runs

Help us improve GitHub Actions

Give feedback

80 workflow runs

Event	Status	Branch	Actor
Update ci.yml	Success	main	Mahshad89
Update ci.yml	Success	main	Mahshad89
Update ci.yml	Failed	main	Mahshad89
Update ci.yml	Success	main	Mahshad89

Update ci.yml #7

Summary

Jobs

test

Run details

Usage

Workflow file

Triggered via push 20 minutes ago

Mahshad89 pushed · 8a36ae5 · main

Status: Success

Total duration: 18s

Artifacts: -

ci.yml

on: push

test 9s

Re-run all jobs