

**Министерство науки и высшего образования
Российской Федерации**

**Федеральное государственное автономное
образовательное учреждение высшего образования**

**«Национальный исследовательский университет
ИТМО»**

**Факультет информационных технологий и
программирования**

Цифровая Культура
Лабораторная работа №4

GitHub Actions. Linter's subsystem. Интеграция CI/CD.

Выполнил студент группы № М3111

Акберов Рустам Ханкишиевич

Подпись:



Санкт-Петербург
2021

Требования к лабораторной работе.

На основе лабораторных работ по программированию - с помощью GitHub Actions Произвести настройку CI CD для вашего проекта - необходимо произвести автоматизацию сборки, прогона тестов и упаковки в exe вашего проекта по событию коммита.

Отчет.

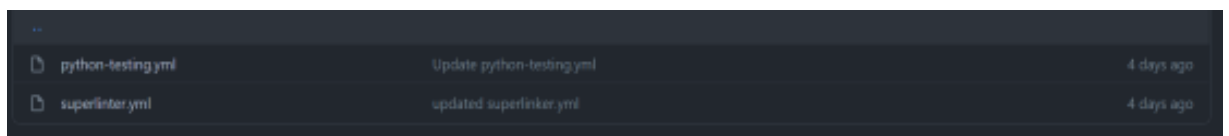
Будем работать с репозиторием, который был специально создан для этой лабораторной работы. Удаленный репозиторий содержит проект на python, тесты к нему через библиотеку unittest

Чтобы продемонстрировать работу Линтера, необходимо создать файл с расширением .yaml

Что такое Линтер?

Если говорить простым языком, то он подстраивает код под стандарты, заданные пользователем, и приводя его в читабельный вид. Правила, которые задает пользователь, описывают отступы, названия создаваемых сущностей, скобки, математические операции, длину строк и множество других аспектов. Каждое отдельное правило кажется не очень важным, но соблюдение их всех – основа хорошего кода.

- 1) Создаем директорию в репозиторий “/github/workflows”



- 2) Файл python-testing.yml будет основным для автоматизации работы с тестами. Здесь прописываем сценарий: при каждом git push запускать файлы тестов с unittest библиотекой. Если все тесты прошли, линтер завершает свою проверку. Смысл в том, чтобы всегда после внесения изменений в тело основной программы или в дерево зависимостей проекта оставлять за собой репозиторий с рабочим кодом на известных тестах.
Линтер запускается при каждом обновлении содержимого репозитория, кроме пуша тэгов. Проверка осуществляется по всем веткам. Система отладки и запуска на серверах Github – Ubuntu (последняя доступная версия). В разделе steps перечисляем, что необходимо сделать: в данном случае установить python и все компоненты requirements.txt для работы проекта (docker v4.1.0, Flask v1.1.1, coverage v4.5.4)

```

name: Run tests on any Push event
on:
  push:
  tags:
    - '!refs/tags/*'
  branches:
    - '*'
jobs:
  run_tests:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@master
      - uses: actions/setup-python@v1
      with:
        python-version: '3.8'
        architecture: 'x64'
      - name: Install requirements
        run: pip install -r requirements.txt
      - name: Run tests
        run: coverage mrunit unittesting_console_calculator.py
      - name: Tests report
        run: coverage report

```

3) Напишем тест, который завалит проверку. Случай обработался правильно.

The screenshot shows a GitHub Actions workflow run for the repository 'Update unittesting_console_calculator.py'. The workflow is titled 'Run tests on any Push event #10'. The left sidebar shows the 'Summary' tab with a list of jobs, including 'run_tests'. The main panel displays the 'run_tests' job details, which failed. The job steps are: 'Set up job', 'Run actions/checkout@master', 'Run actions/setup-python@v1', 'Install requirements', 'Run tests', 'Test report', 'Post Run actions/checkout@master', and 'Complete job'. The 'Run tests' step is highlighted with a red error icon. The step's command is: `coverage mrunit unittesting_console_calculator.py`. The output shows the command being executed and the resulting error: `AssertionError: 88101211 != 8811211`. The job status is 'Failed' and the workflow status is 'Failed'.

4) GitHub Actions:

The screenshot displays the GitHub Actions interface for a workflow named 'Delete python-app.yml'. The workflow is triggered by a 'Push event' and is currently in a 'succeeded' state. The interface shows a summary of the workflow run, including the job name 'run_tests' and its duration 'succeeded 4 days ago in 17s'. A list of steps is provided, showing the sequence of actions performed: 'Set up job', 'Run actions/checkout@master', 'Run actions/setup-python@v1', 'Install requirements', 'Run tests', 'Tests report', 'Post Run actions/checkout@master', and 'Complete job'. Below the summary, the 'Set up job' step is expanded, showing the environment details: 'Current runner version: 2.285.0', 'Operating System: Ubuntu 20.04.3 LTS', 'Virtual Environment: Environment: ubuntu-20.04, Version: 20211129.1', 'Included Software: https://github.com/actions/virtual-environments/blob/ubuntu20/20211129.1/images/linux/Ubuntu2004-README.md', 'Image Release: https://github.com/actions/virtual-environments/releases/tag/ubuntu2004/20211129.1', 'Virtual Environment Provisioner: 1.0.0.0-master-20211123-1', 'GITHUB_TOKEN Permissions: Actions: write, Checks: write, Contents: write, Deployments: write, Discussions: write, Issues: write, Metadata: read, Packages: write, Pages: write, PullRequests: write, RepositoryProjects: write, SecurityEvents: write, Statuses: write, Secret source: Actions, Prepare workflow directory, Prepare all required actions, Getting action download info, Download action repository 'actions/checkout@master' (SHA: 238611db0e0520a1c1f4f7bcb00e5a339f18b7), and Download action repository 'actions/setup-python@v1' (SHA: 152b47c0dd0521b8e9c93f72d362c00b6c4f20).

Delete python-app.yml Run tests on any Push event #7

Summary

Jobs

- run_tests

run_tests
succeeded 4 days ago in 17s

- > Set up job
- > Run actions/checkout@master
- > Run actions/setup-python@v1
- > Install requirements
- > Run tests
- > Tests report
- > Post Run actions/checkout@master
- > Complete job

Set up job

```
1 Current runner version: '2.285.0'
2 Operating System
3 Ubuntu
4 20.04.3
5 LTS
6 Virtual Environment
7 Environment: ubuntu-20.04
8 Version: 20211129.1
9 Included Software: https://github.com/actions/virtual-environments/blob/ubuntu20/20211129.1/images/linux/Ubuntu2004-README.md
10 Image Release: https://github.com/actions/virtual-environments/releases/tag/ubuntu2004/20211129.1
11 Virtual Environment Provisioner
12 1.0.0.0-master-20211123-1
13 GITHUB_TOKEN Permissions
14 Actions: write
15 Checks: write
16 Contents: write
17 Deployments: write
18 Discussions: write
19 Issues: write
20 Metadata: read
21 Packages: write
22 Pages: write
23 PullRequests: write
24 RepositoryProjects: write
25 SecurityEvents: write
26 Statuses: write
27 Secret source: Actions
28 Prepare workflow directory
29 Prepare all required actions
30 Getting action download info
31 Download action repository 'actions/checkout@master' (SHA: 238611db0e0520a1c1f4f7bcb00e5a339f18b7)
32 Download action repository 'actions/setup-python@v1' (SHA: 152b47c0dd0521b8e9c93f72d362c00b6c4f20)
```

✓ Tests report

```
1 ▼ Run coverage report
2 coverage report
3 shell: /usr/bin/bash -e {0}
4 env:
5 pythonLocation: /opt/hostedtoolcache/Python/3.8.12/x64
6 Name                               Stmts  Miss  Cover
7 -----
8 console_calculator.py             20     13   35%
9 unittesting_console_calculator.py  18      1   94%
10 -----
11 TOTAL                             38     14   63%
```