Министерство науки и высшего образования РФ

Федеральное государственное автономное образовательное учреждение высшего образования «Национальный исследовательский университет ИТМО»

Факультет информационных технологий и программирования

Лабораторная работа №4

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

Выполнил студент группы № M3102

Лопатенко Георгий Валентинович

Подпись:

Проверил:

Приискалов Роман Андреевич

Санкт-Петербург

2021

Требования к выполнению лабораторной работы

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.

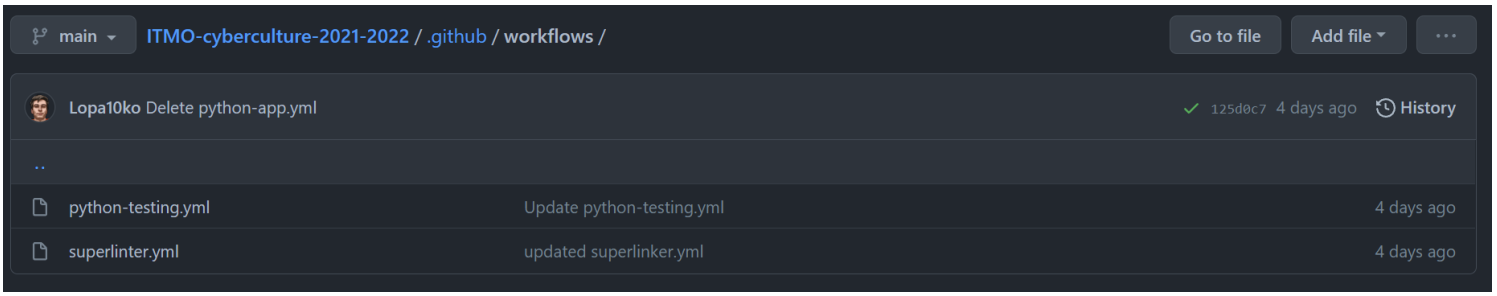
Отчет

Будем работать с репозиторием из второй лабораторной работы в курсе Цифровая культура. Удаленный репозиторий содержит в себе простой проект на Python и тесты к нему через библиотеку unittest.
Чтобы продемонстрировать работу линтера, необходимо создать файл .yaml (Yet Another Markup Language), но сначала разберемся, что из себя представляет линтер и как он работает на платформе Github:

Линтер – это программа, которая проверяет код на соответствие стандартам в соответствии с определенным набором правил. Правила 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. (https://github.com/marketplace/actions/super-linter)

Главная задача линтера – сделать код единообразным, удобным для восприятия и самим программистом, и другими людьми, которые будут читать код. В разных командах могут использоваться разные линтеры и разные наборы правил для них, но главное – уметь работать с линтером в принципе, а привыкнуть писать по определенным правилам будет несложно. (https://github.com/github/super-linter)

1. Создадим директорию в репо “./github/workflows”. (!)Другое название директории не будет триггерить работу линтера.

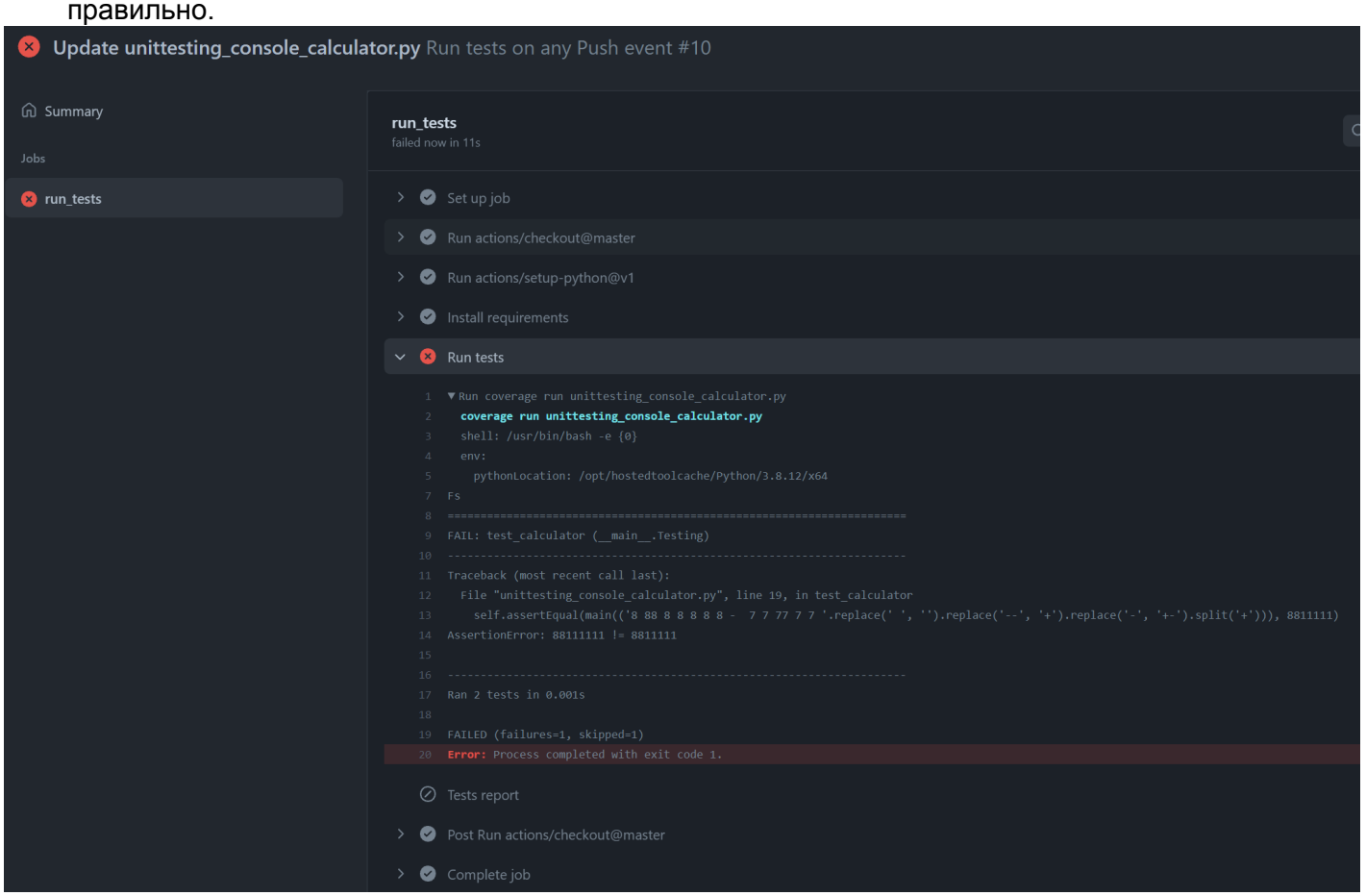


2. Файл python-testing.yml будет основным для автоматизации работы с тестами. Здесь прописываем сценарий: при каждом git push запускай файлы тестов с unittest библиотекой. Если все тесты прошли, линтер завершает свою проверку. Смысл в том, чтобы всегда после внесения изменений в тело основной программы или в дерево зависимостей проекта оставлять за собой репозиторий с рабочим кодом на известных тестах.

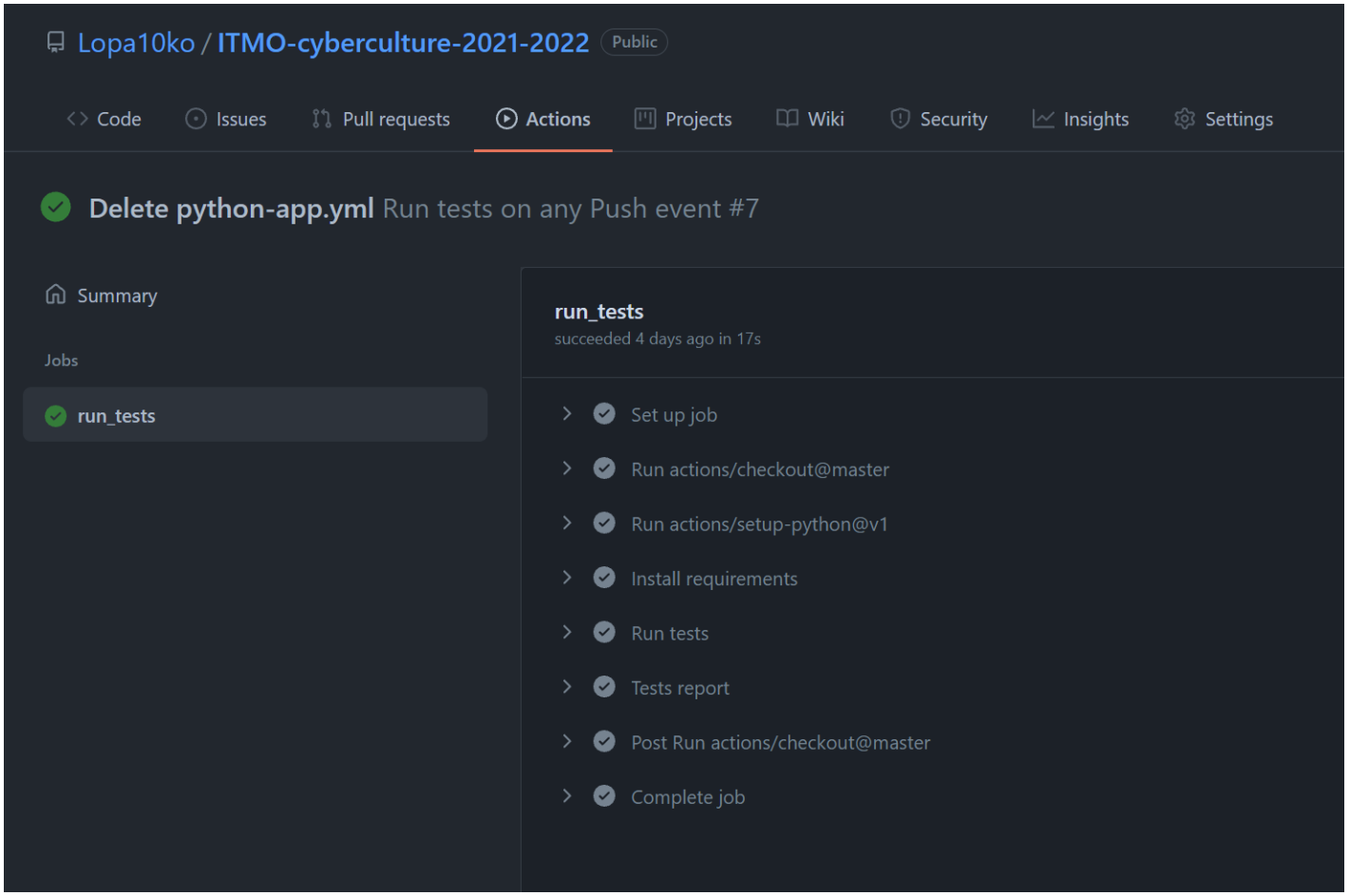
   Линтер запускается при каждом обновлении содержимого репозитория, кроме пуша тэгов. Проверка осуществляется по всем веткам. Система отладки и запуска на серверах Github – Ubuntu (последняя доступная версия). В разделе steps перечисляем, что необходимо сделать: в данном случае установить python и все компоненты requirements.txt для работы проекта (docker v4.1.0, Flask v1.1.1, coverage v4.5.4)

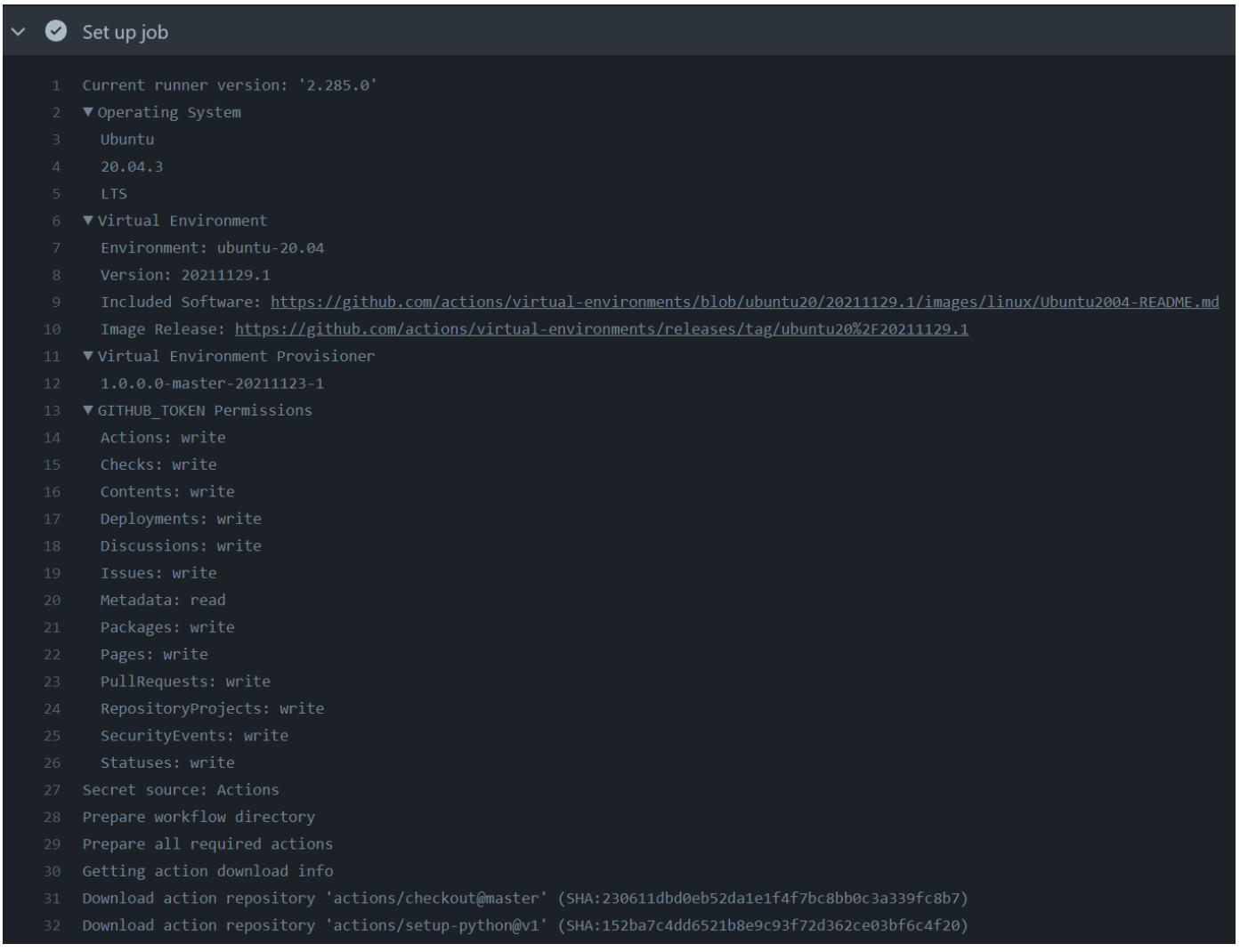
```yaml
name: Run tests on any Push event
on:
  push:
    tags:
      - '!refs/tags/*'
    branches:
      - '*'
jobs:
  run_tests:
    runs-on: ubuntu-latest
    steps:
      - uses: actions/checkout@master
      - uses: actions/setup-python@v1
        with:
          python-version: '3.8'
        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

3. Умышленно напишем тест, который валит проверку. Как видно, случай обрабатывается правильно.



4. Посмотрим, какую информацию можно получить во вкладке Github Actions:





Tests report

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