Выполнение работы удобно разбить на следующие этапы:

1. Создание репозитория на сервисе Github
2. Написание тестовой программы (взять из предыдущей лабы)
3. Разобраться с тем, как создавать список установленных модулей requrements.txt и устанавливать модули по списку командой

**pip freeze > requirements.txt –** создать список установленных модулей в requirements.txt

**pip install –r requirements.txt –** установить модули из списка requirements.txt

1. Разобраться с тем, как автоматизировать запуск тестов и генерацию отчета по ним (в виде 1 файла html)

**pytest --html=report.html --self-contained-html**

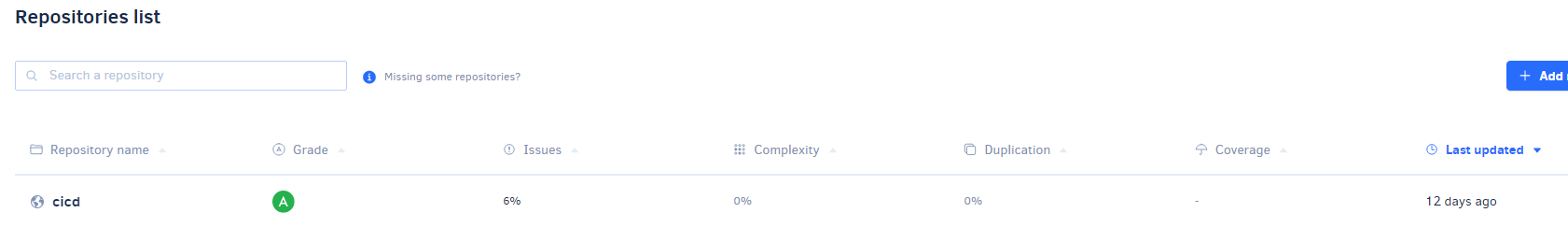
1. Разобраться с тем, как проводить анализ покрытия тестами кода и генерировать отчеты в виде html и xml

**coverage run –m pytest --html=report.html --self-contained-html –** сделать отчет и сохранить его в виде .coverage

**coverage html --directory coverage\_result –** сгенерировать отчет в формате html из .coverage

**coverage xml -o coverage\_result.xml -** сгенерировать отчет в формате xml из .coverage

1. Интеграция анализаторов кода Codacy в репозиторий



1. Разобраться с тем, как выгружать xml отчет покрытия кода тестами на сервис Codacy

bash <(curl -Ls https://coverage.codacy.com/get.sh) report -r <coverage report file name>

1. Реализовать шаги по тестированию кода в Github Actions



1. Разобраться с тем, как запускать в Github Actions Job, если предыдущая Job завершилась без ошибок



1. Реализовать выгрузку кода в виде артефакта при успешном прохождении тестов

