**Задание 5.4. (для самопроверки)**

Дана маленькая библиотека, вычисляющая некоторые тригонометрические функции. Директория проекта имеет следующую структуру:

mat\_lib

├─docs

│ └─math\_lib\_docs.txt

│

└─pyfiles

└─factorial.py

└─test.py

└─trigonometry.py

Скачать файлы можно [здесь](https://lms-cdn.skillfactory.ru/assets/courseware/v1/d6574cf88aacf84749d9c461dfa5a132/asset-v1:SkillFactory+DSPRMGU+2023_FEB+type@asset+block/mat_lib.zip).

Назначение каждого из файлов:

* 1. *math\_lib\_docs.txt* — краткая документация по функционалу библиотеки.
  2. *factorial.py* — модуль с функцией factorial(), которая вычисляет факториал переданного числа.
  3. *trigonometry.py* — модуль с тригонометрическими функциями, например sin().
  4. *test.py* — модуль для проверки функционала тригонометрических функций для разных значений углов.

**Задания:**

* 1. Инициализируйте репозиторий в папке *mat\_lib*.
  2. Откройте папку *.git/*:
     + - Узнайте, на что после инициализации репозитория указывает *HEAD*. Открыть файл *HEAD* можно с помощью текстового редактора.
       - Найдите в папке *./git* файл, на который указывает *HEAD* (в этом пункте есть подвох).
  3. Добавьте все файлы в индекс.
  4. Сделайте первый коммит.
     + - С помощью команды show посмотрите на данные о совершённом коммите.
       - Проверьте, на что сейчас указывает *HEAD*.
       - Просмотрите файл, на который указывает *HEAD*.
  5. C помощью команды на *Python* ./pyfiles/test.py или средствами *VS Code* запустите файл *test.py*. Посмотрите на состояние рабочей директории, чтобы убедиться, что появились файлы кэша (папка *\_\_pycache\_\_*).
  6. Сделайте второй коммит, предварительно добавив новые файлы в индекс.
     + - Посмотрите на информацию о коммите.
       - Посмотрите, на что сейчас указывает *HEAD*.
       - Проверьте файл, на который указывает *HEAD*.
  7. Отметьте изменения второго коммита. Убедитесь, что файлы с кэшем пропали из вашей директории. Вновь посмотрите на содержимое файла *HEAD*.

Решение

* 1. Заходим в папку *./git* и открываем файл *HEAD*. В ней будет находиться ссылка *(ref)* на файл *refs/heads/master*. Однако если мы зайдём в папку *refs/heads*, файла *master* там не окажется, так как не было совершено ни одного коммита.
  2. Добавляем файлы в индекс:
  3. git **add** .

Смотрим статус:

git status

* 1. Делаем первый коммит:
  2. git **commit** -m 'first commit'

Смотрим на информацию о коммите (ваш хеш может отличаться):

git **show** 5302eac

Смотрим на содержимое файла *HEAD*. Он будет указывать на тот же файл *refs/heads/master*. Если мы откроем папку *refs/heads*, то увидим, что файл *master* появился и его содержимое совпадает с хешем только что сделанного коммита.

* 1. Выполняем код файла *test.py*:
  2. python ./pyfiles/test.py

После выполнения файла в директории *pyfiles* появится папка *\_\_pycache\_\_*, которая будет содержать кэш.

* 1. Добавляем файлы в индекс:
  2. git **add** .

Делаем второй коммит:

git **commit** -m 'second commit'

Смотрим на информацию о коммите (ваш хеш может отличаться):

git **show** 0417301

Смотрим содержимое файла *HEAD*. Он всё так же указывает на файл *refs/heads/master*. Открываем его содержимое — теперь там будет указан хеш нашего последнего коммита.

* 1. Откатываем последний коммит:

git revert 0417301