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

**Структура директории mat\_lib**

mat\_lib

    ├─docs

    │   └─math\_lib\_docs.txt

    │

    └─pyfiles

 └─factorial.py

 └─test.py

  └─trigonometry.py

Назначение каждого из файлов:

1. **[ math\_lib\_docs.txt ]**Короткая 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

Далее приводим содержимое перечисленных модулей:

**Содержимое файла math\_lib\_docs.txt**

This **is** the library which designation **is** to implement some math functions

**Содержимое файла factorial.py**

**def** **factorial**(x):

    ans = 1

**if** x < 0:

**raise** ValueError('x must be greater than 0')

**for** i **in** range(1, x+1):

        ans \*= i

**return** ans

**Содержимое файла trigonometry.py**

**from** factorial **import** factorial **as** fct

**def** **sin**(x):

    sin = 1 - (x\*\*2/fct(2)) + (x\*\*4/fct(4)) - (x\*\*6/fct(6)) + (x\*\*8/fct(8)) - (x\*\*10/fct(10))

**return** round(sin, 5)

**Содержимое файла test.py**

**from** trigonometry **import** sin

**import** math

pi = math.pi

print('pi:', pi)

**for** alpha **in** [0, pi, pi/2, pi/3, pi/4, pi/6]:

    print(f'For angle: {0 if alpha == 0 else "pi/"+str(int(pi/alpha))}, Sine  is ~ {sin(alpha)}')

**Условие**

1) Инициализируйте репозиторий в папке mat\_lib

2) Задайте имя пользователя и email глобально

1. Проверьте, что изменения внесены в файл глобальных настроек .gitconfig

2. Проверьте, что файл локальных настроек .git/config не был изменен

3) Изучите содержимое папки .git/

1. Узнайте, на что сейчас указывает HEAD

2. Просмотрите файл, на который указывает HEAD (в этом пункте есть подвох)

4) Добавьте все файлы в индекс

5) Сделайте первый коммит

1. Просмотрите объект коммита, найдите хэш объекта-дерева корня репозитория

2. Просмотрите объект дерева корня репозитория

3. Проверьте, на что указывает HEAD сейчас

4. Просмотрите файл, на который указывает HEAD

5. Ответьте на вопрос: на что указывает текущая ветка? Для этого просмотрите на объект, на который указывает ветка.

6) Выполните файл test.py

1. Просмотрите статус файлов, чтобы обнаружить, что появились файлы кэша

7) Сделайте второй коммит

1. Просмотрите объект коммита, найдите хэш родительского коммита

2. Посмотрите, на что сейчас указывает HEAD

3. Проверьте файл, на который указывает HEAD

4. Узнайте, на что указывает текущая ветка. Для этого просмотрите на объект, на который указывает ветка.