**Задание 1: Поставить NodeJs и позапускать \*.js-файлы из консоли**

Поставим NodeJs.

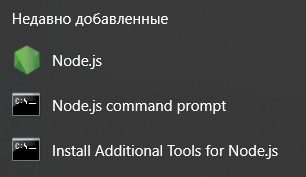


Рисунок 1 – node js поставлен

Создадим какой-нибудь файл \*.js.

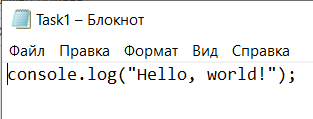


Рисунок 2 – файл Task1.js

Запустим его через консоль.



Рисунок 3 – запуск файла и результат

**Задание 2: Написать скрипт, который считывает текстовый файл с диска и записывает его в обратном порядке ('абв' -> 'вба') в тот же файл.**

Создадим файл, который будет считываться.

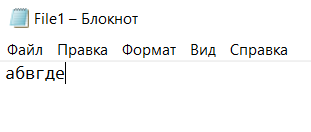


Рисунок 4 – File1.txt, который будет считываться

Напишем скрипт, который считывает наш текстовый файл и записывает его в обратном порядке ('абв' -> 'вба') в тот же файл.

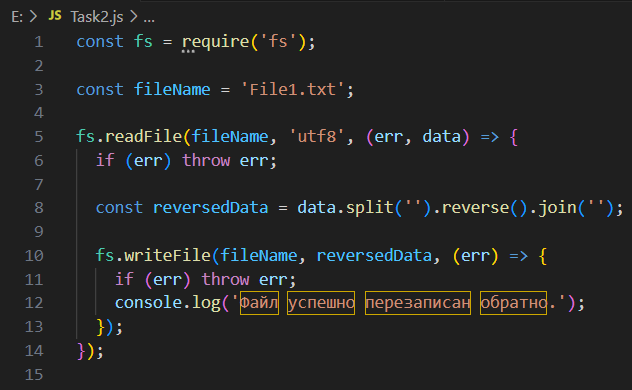


Рисунок 5 – Скрипт для задания 2

Запустим Task2.js файл через консоль.



Рисунок 6 – Запуск скрипта для задания 2

Скрипт успешно выполнился, появилась надпись, что файл перезаписан. Откроем файл, чтобы в этом убедиться. Результат показан на рисунке 7

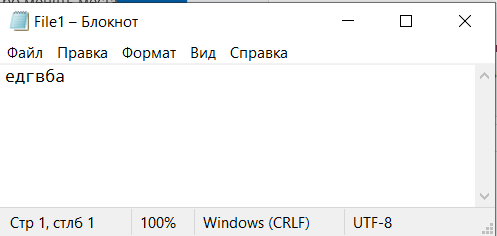


Рисунок 7 – Результат выполнения скрипта

**Задание 3: Написать сервер, который при запросе GET / отправляет содержимое некоего файла на диске (см. предыдущую задачу). POST / добавляет в файл тело запроса (body).**

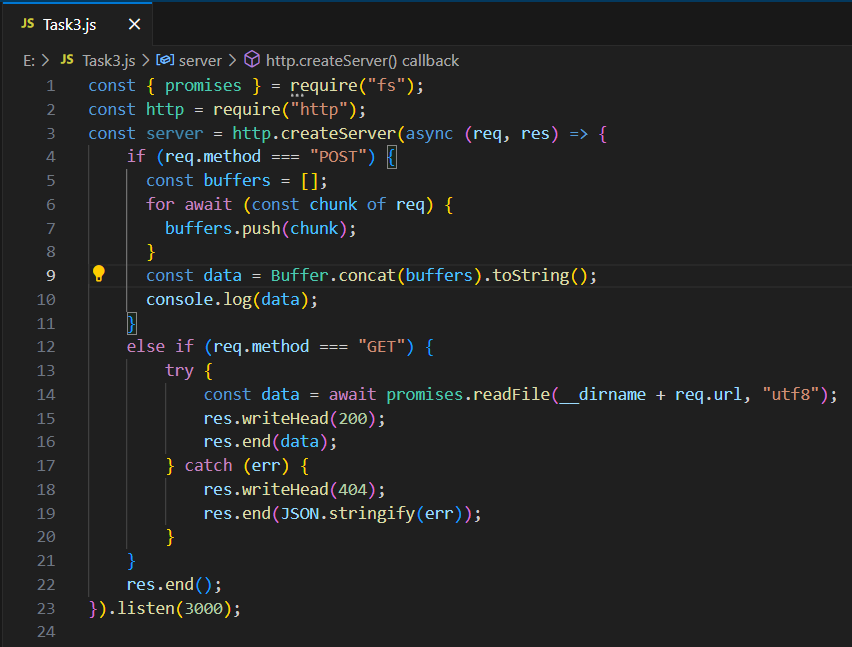
****

Рисунок 8 – Скрипт для задания 3

Сервер получает запрос вида GET /filename, ищет файл в текущем каталоге и отдаёт содержимое, при POST-запросе – получаем body запроса в консоль

Запустим Task3.js файл через консоль.



Рисунок 9 – Запуск скрипта для задания 3

Сервер запущен на порту 3000. Проверим его. Устанавливаем Postman для отправки запросов.

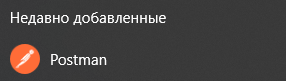


Рисунок 10 – Postman установлен

Пишем GET запрос, в качестве файла выберем текстовый документ File1.txt из предыдущего задания, отправляем. В ответе видим содержимое файла. Все исправно работает. Это показано на рисунке 11

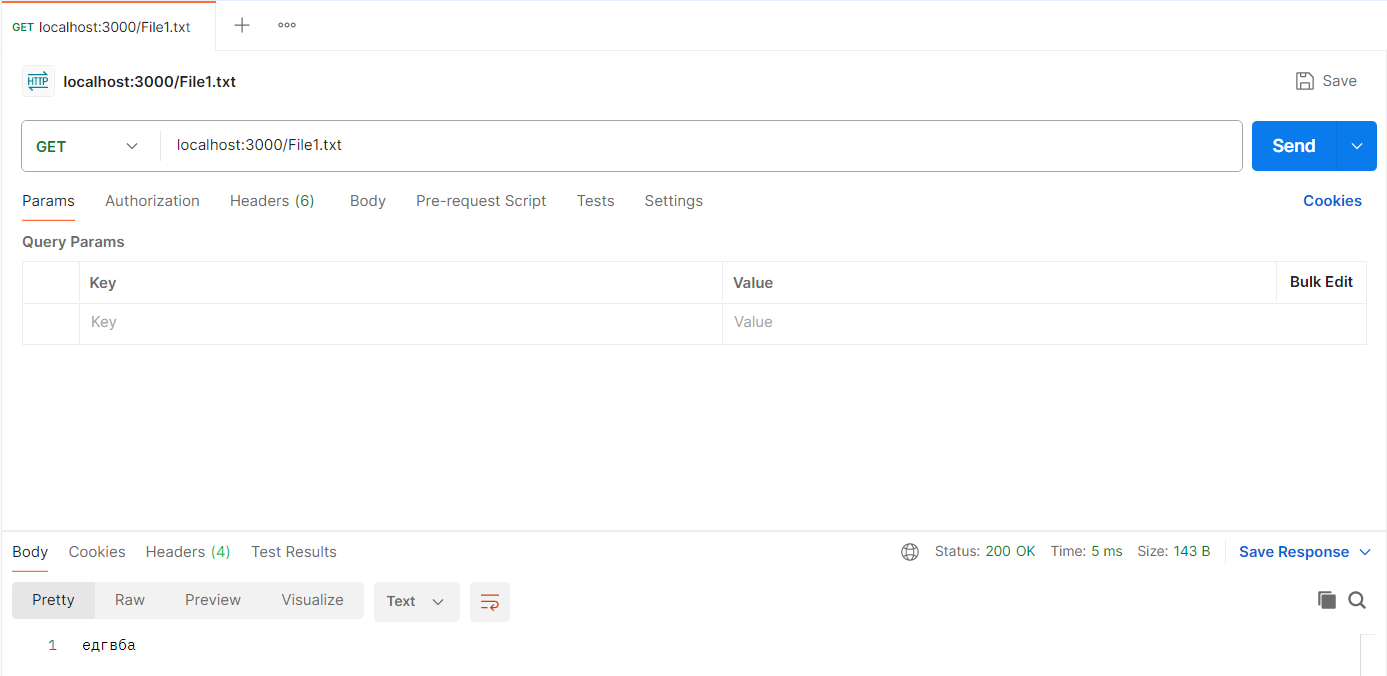
****

Рисунок 11 – Ответ на GET запрос

Теперь протестируем POST запрос. Пишем в body какой-нибудь текст, отправляем запрос.

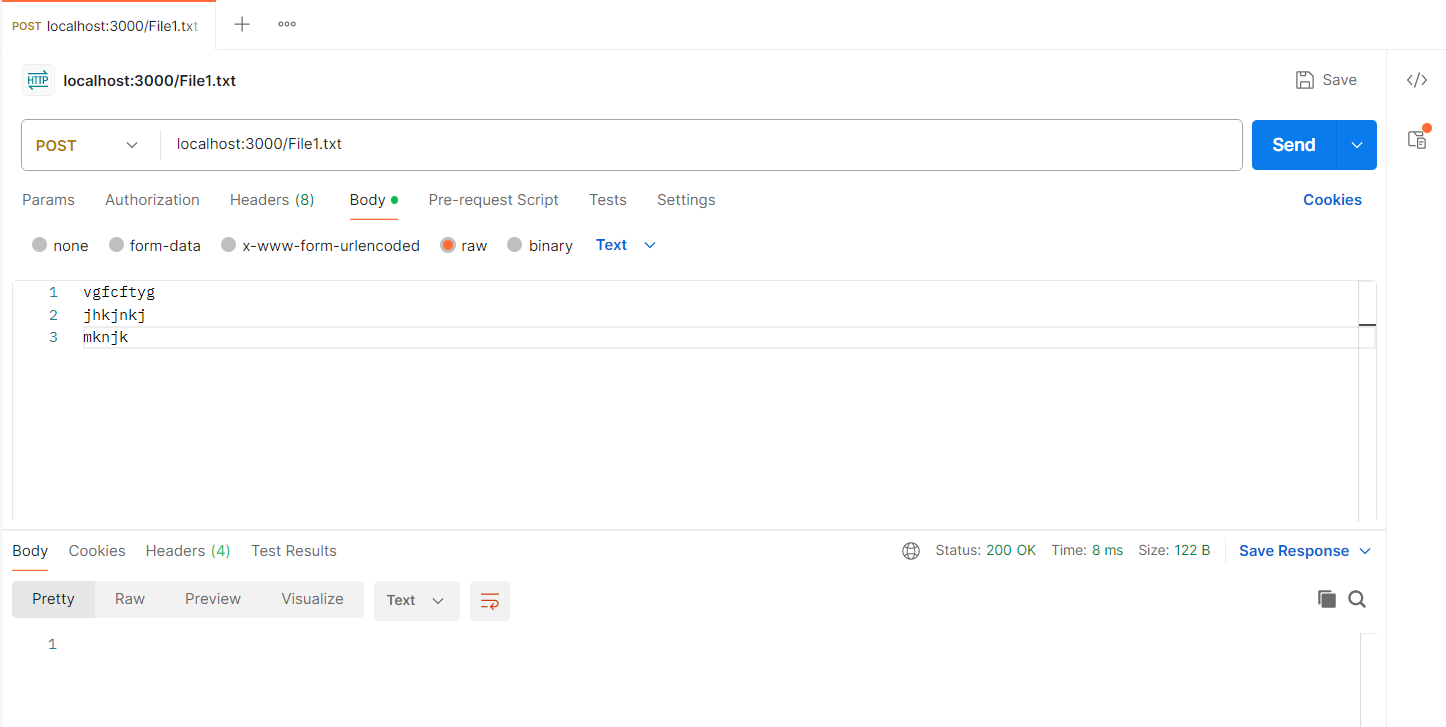
****

Рисунок 12 – Ответ на POST запрос

Body запроса отправился в консоль, как и было задумано. Скрипт работает исправно. Как это выглядит показано на рисунке 13.

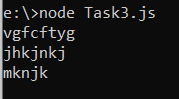
****

Рисунок 13 – body запроса в консоли

Останавливаем работу сервера сочетанием Ctrl+C