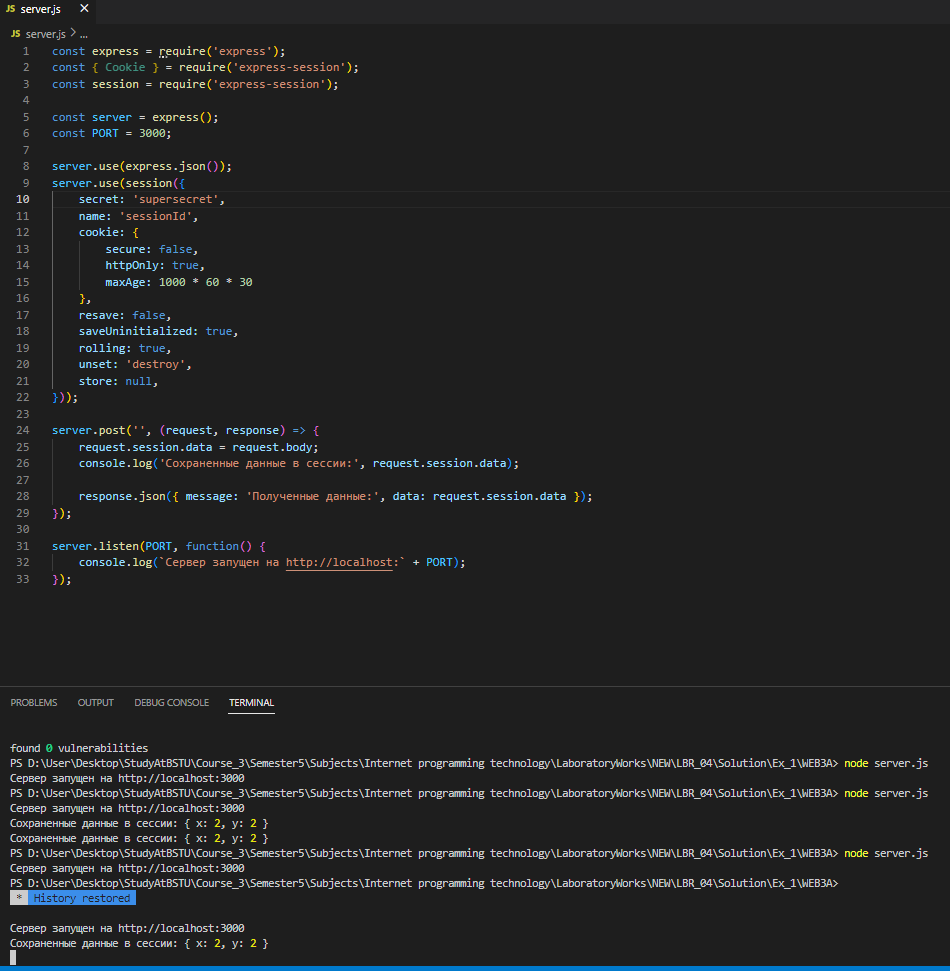
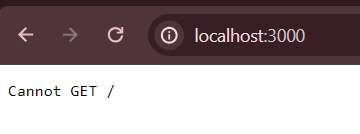
1. *Разработайте web-приложение* ***WEB3A****, содержащее HTTP-обработчик post-запроса.*

Код:

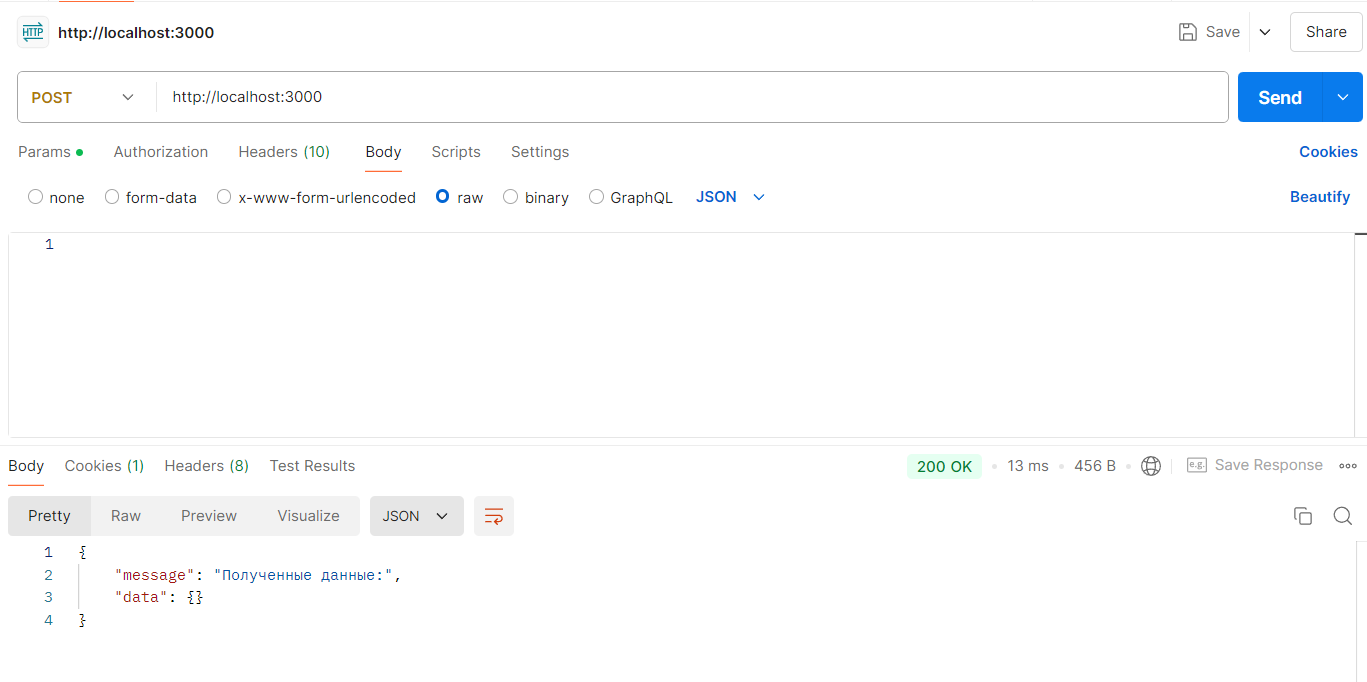


Get-запрос на сервер:

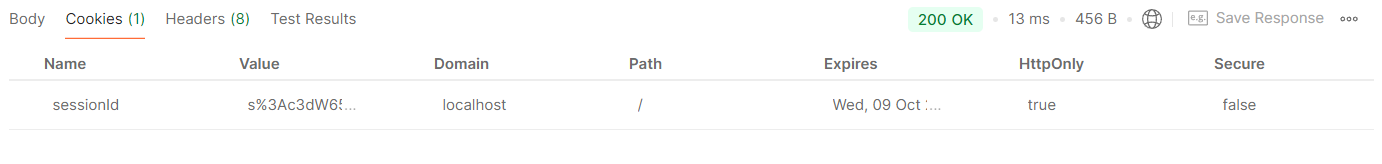


1. *В качестве клиента используйте POSTMAN.*

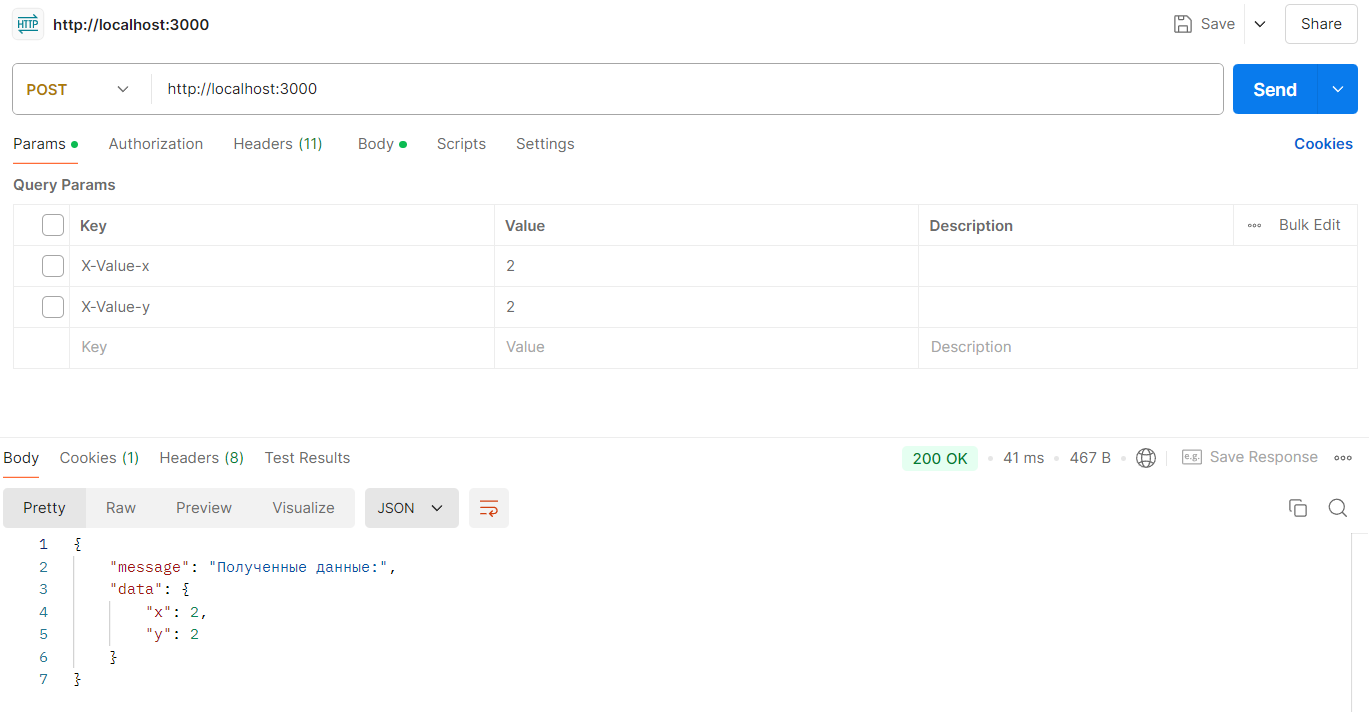
Post-запрос на сервер, ничего не передаем:



Содержимое куки:



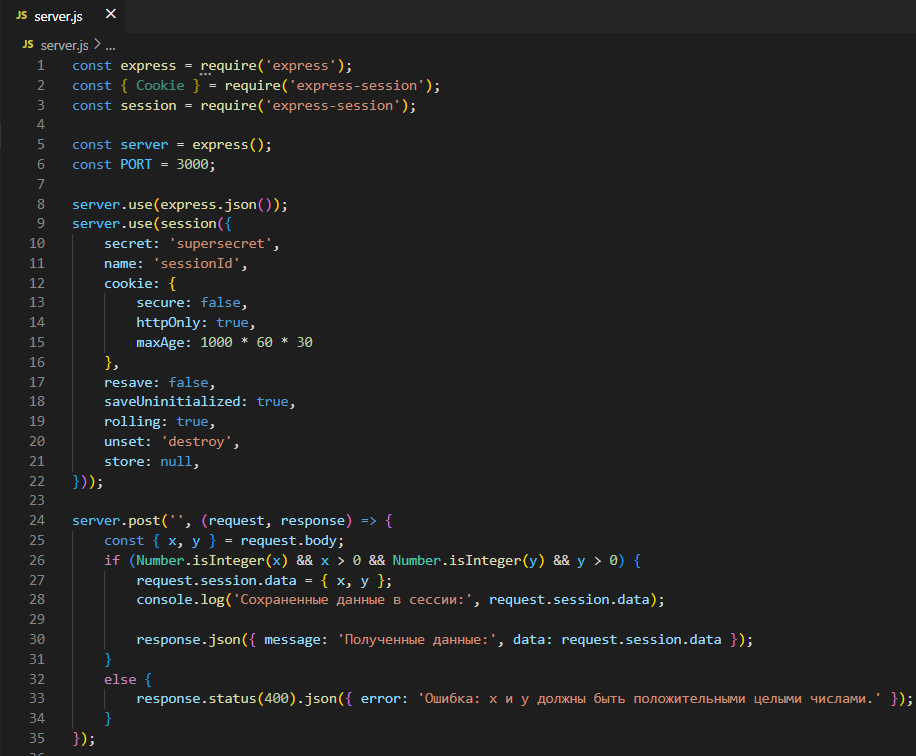
Post-запрос на сервер, передаем 2 и 2 :

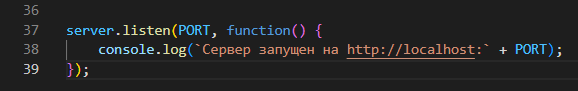


Содержимое куки: 

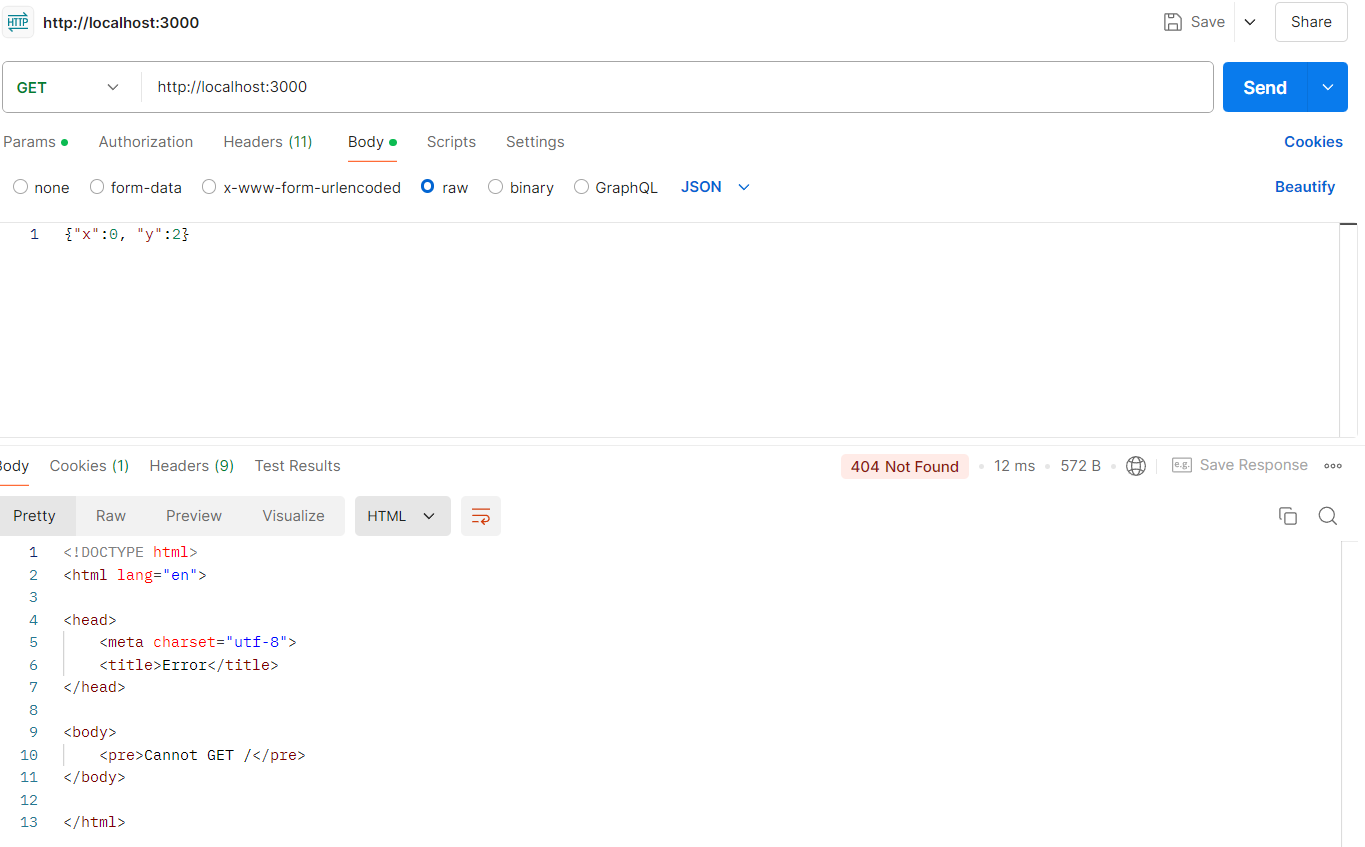
1. *HTTP-обработчик должен принимать запрос, содержащий json-сообщение {****"x":N1, "y":N2}****, где* ***N1*** *и N2 положительные целочисленные значения.*

Код:

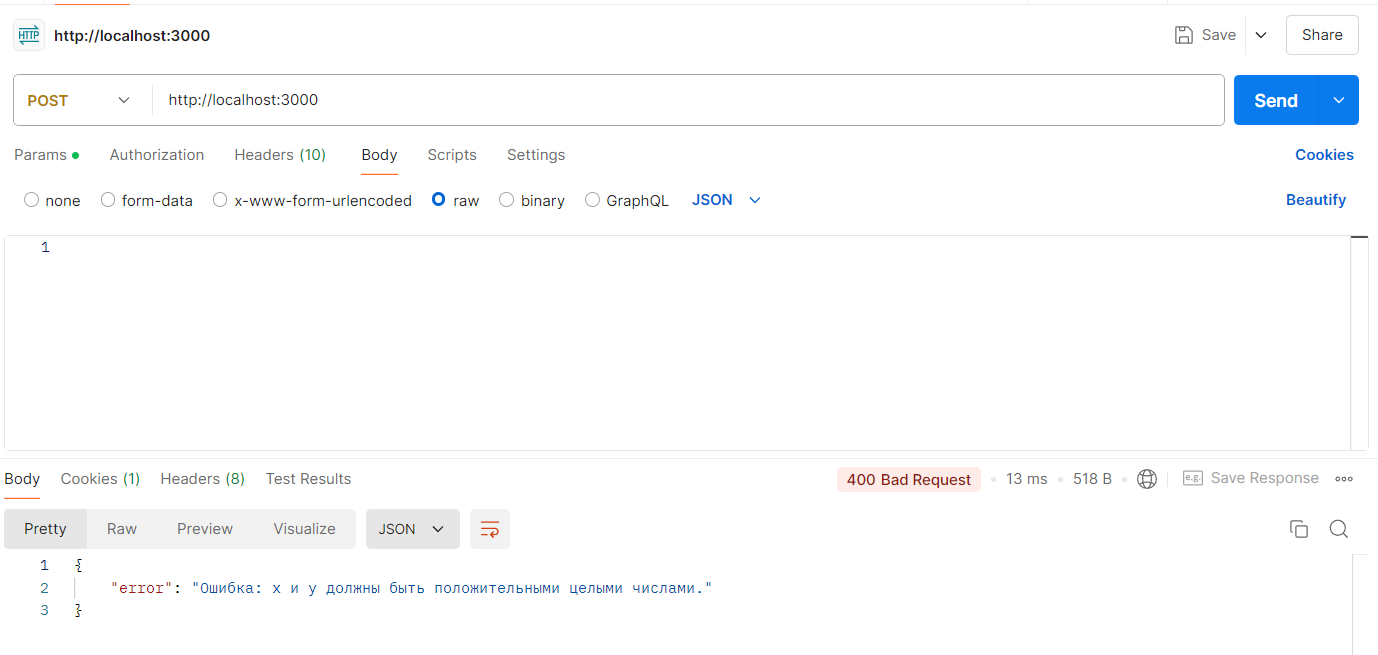




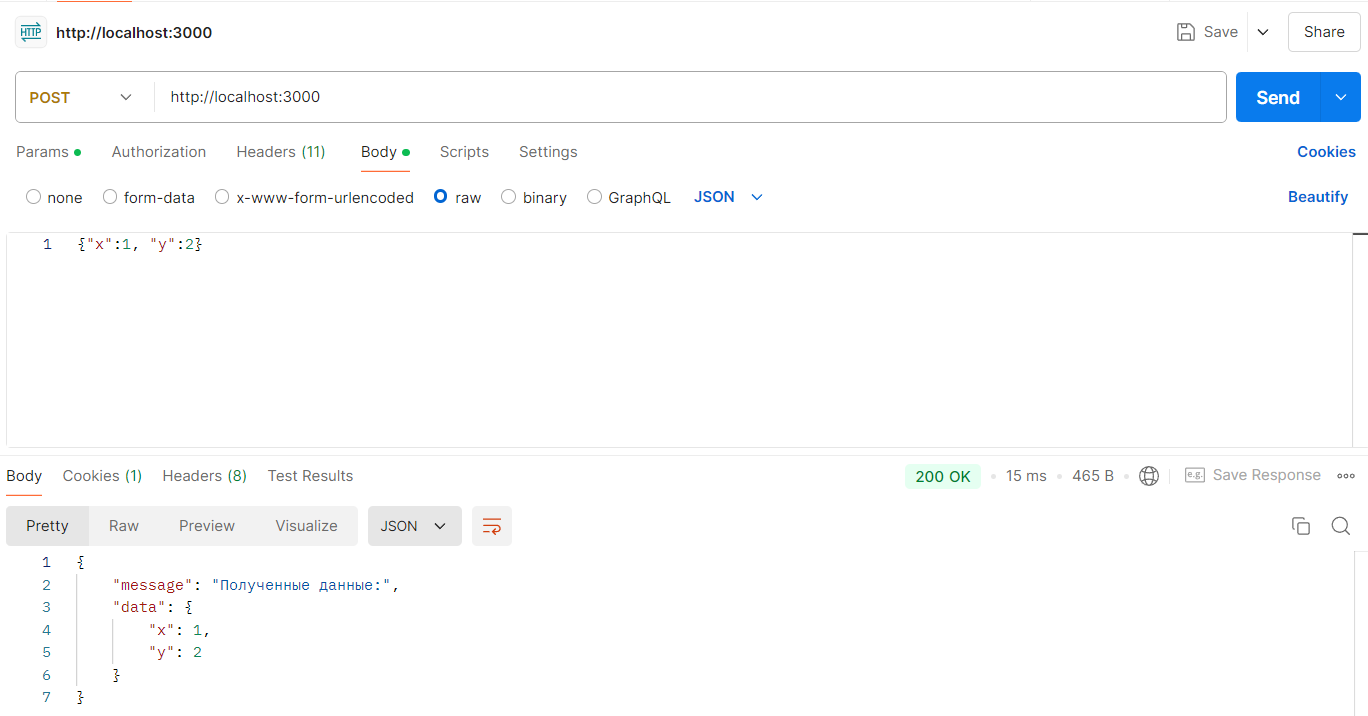
Get-запрос на сервер без ввода данных:



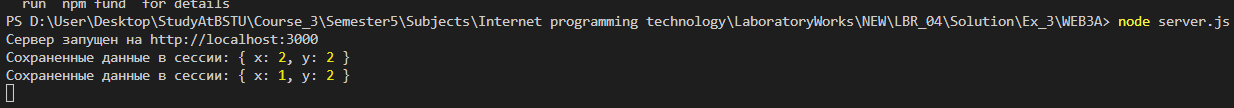
Post-запрос на сервер без ввода данных:



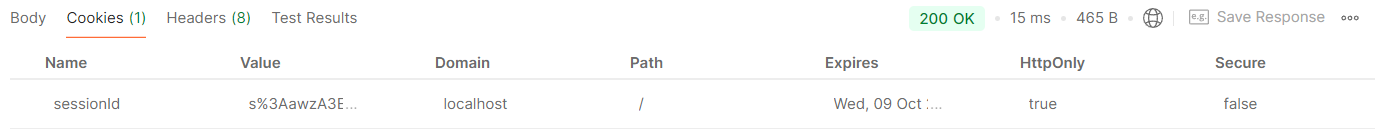
Post-запрос на сервер с вводом данных:



Результаты в консоли:

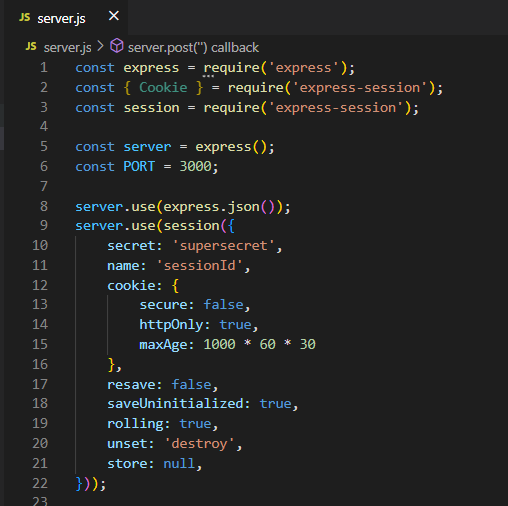


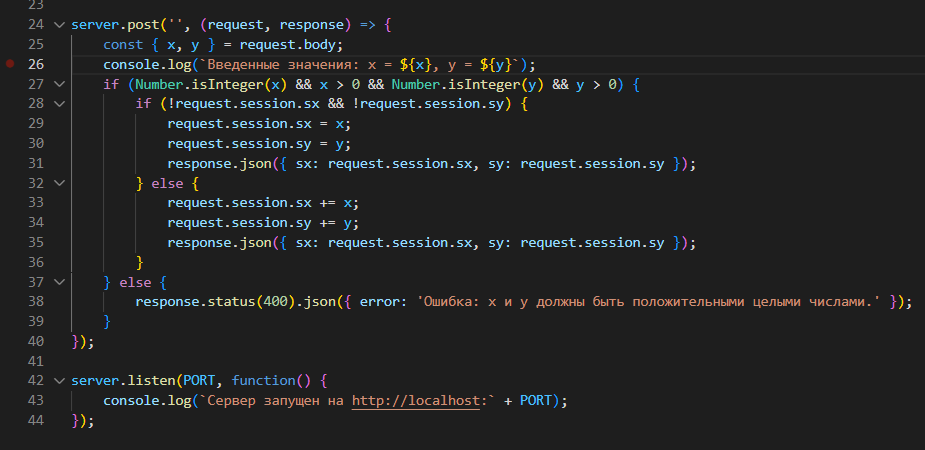
Куки:



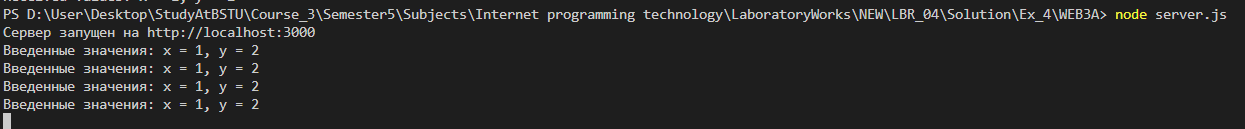
*4. На первый post-запрос HTTP-обработчик должен отправит ответ с json-сообщением {****"sx":N1, "sy":N2}*** *(значения* ***N1*** *и* ***N2*** *из первого запроса). На все последующие post-запросы HTTP-обработчик отвечает json-сообщением* ***{"sx":sN1, "sy":sN2}****, где* ***sN1*** *и* ***sN2*** *накопленные суммы.*

Код:

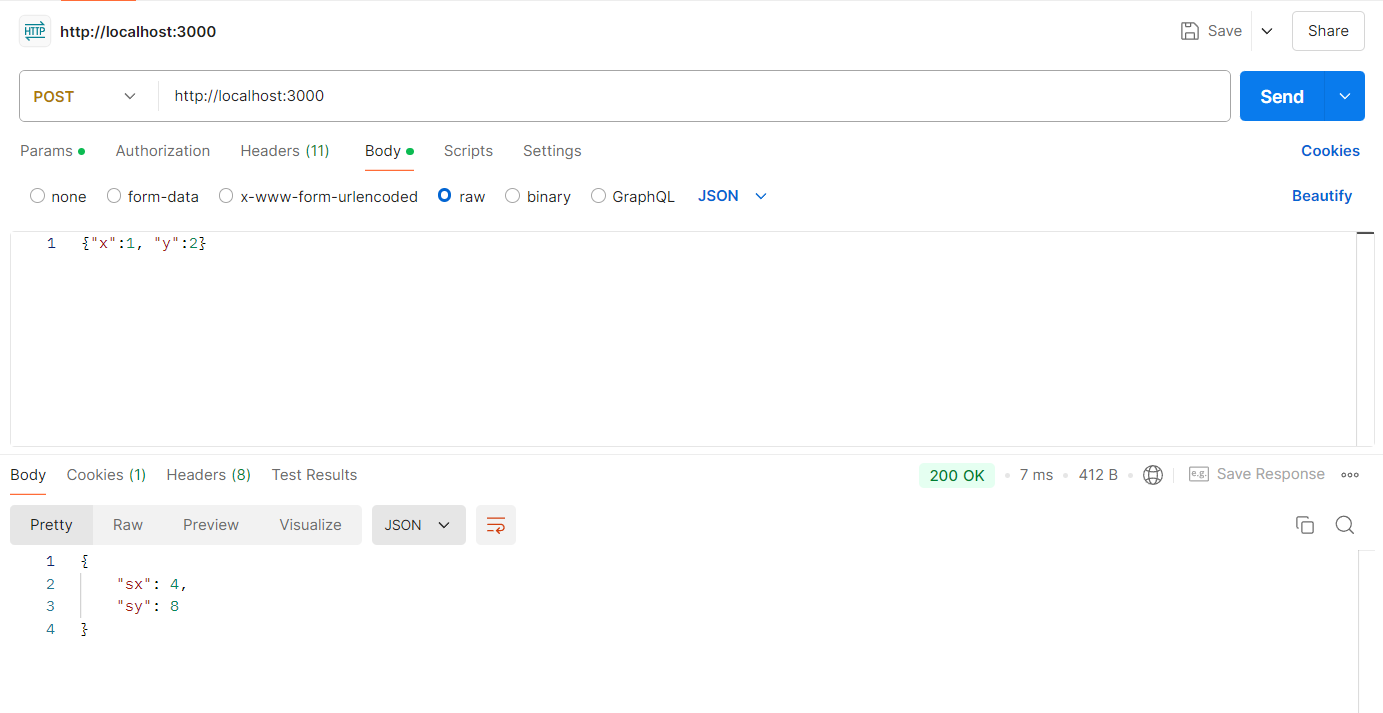




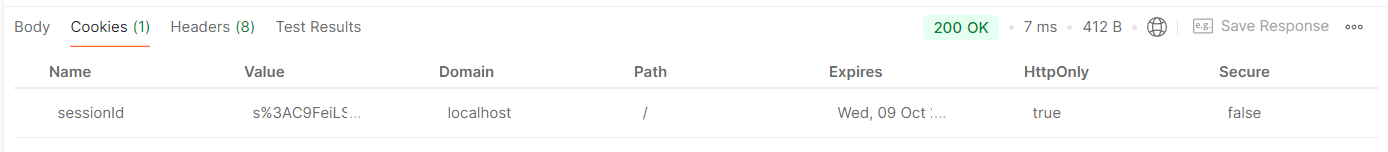
Консоль:



Post-запрос на сервер с вводом данных:



Куки:

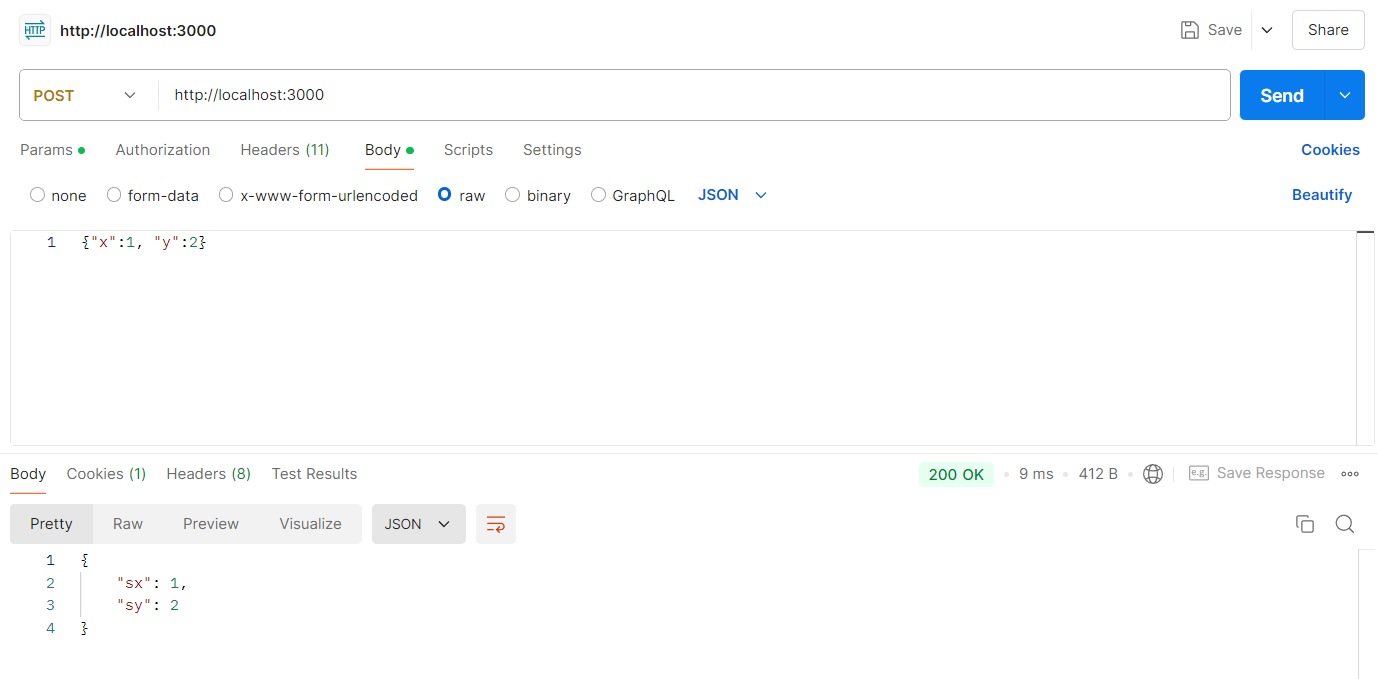


*5. Каждый запрос с номером кратным 5, возвращает накопленную сумму и начинает новое накопление с 0 (как при первом запросе).*

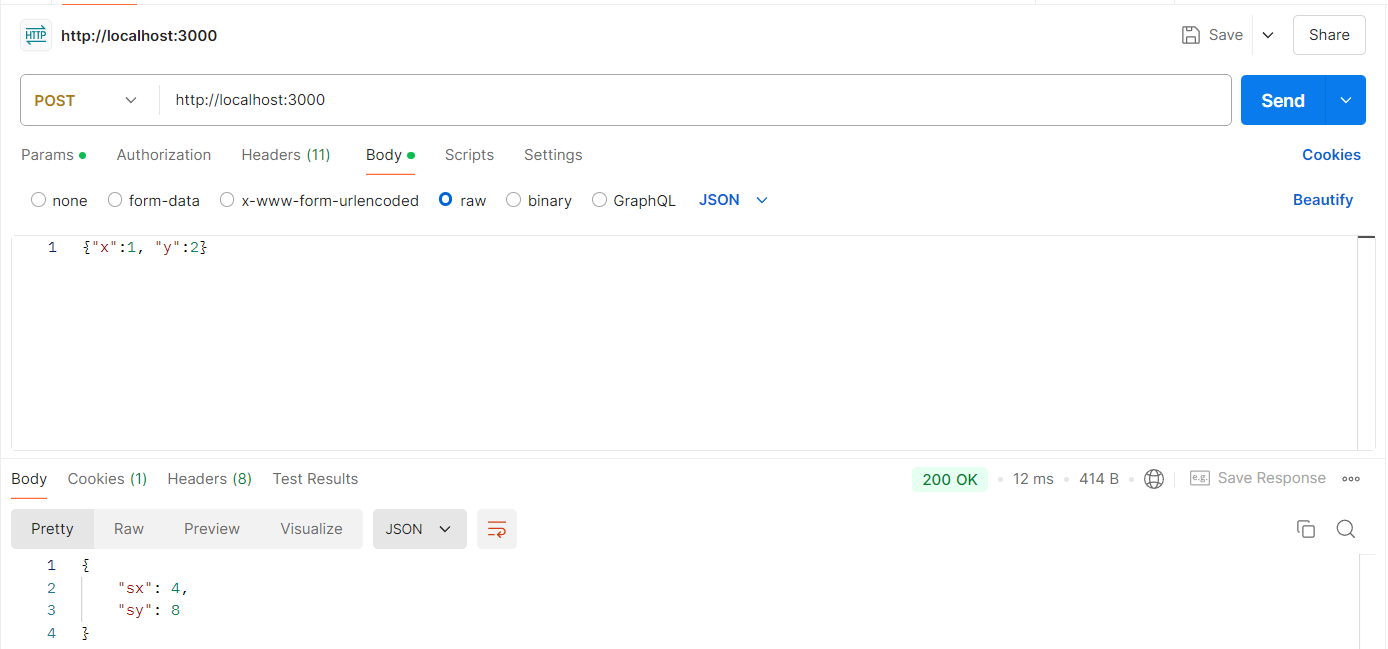
Post-запрос с данными.

История Post-запросов из консоли:

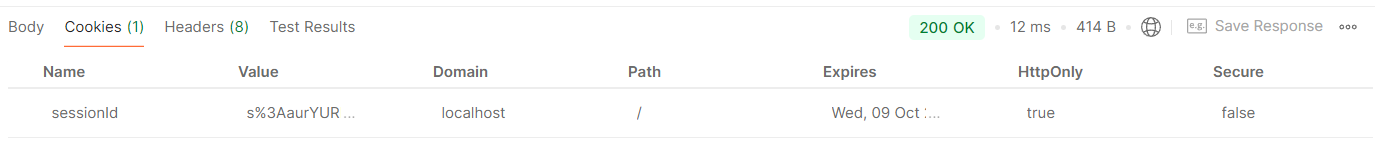
1-ый post-запрос:



5-ый post-запрос:



Файлы куки:



1. *Разработайте в POSTMAN тестовый скрипт, с запросами из примера, демонстрирующий работу HTTP-обработчика.*



