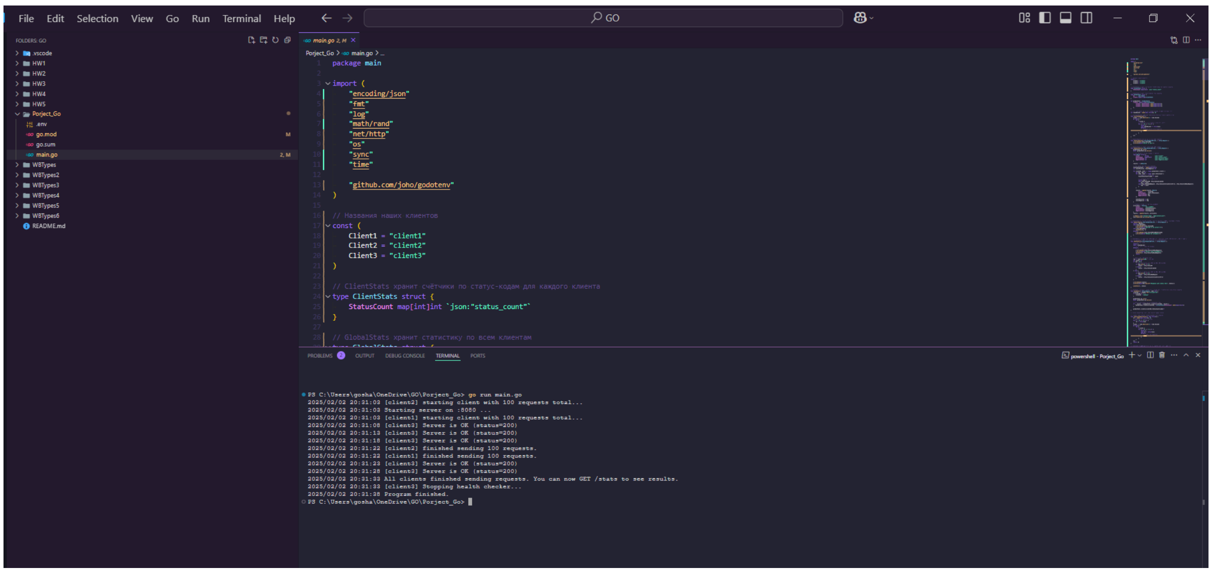
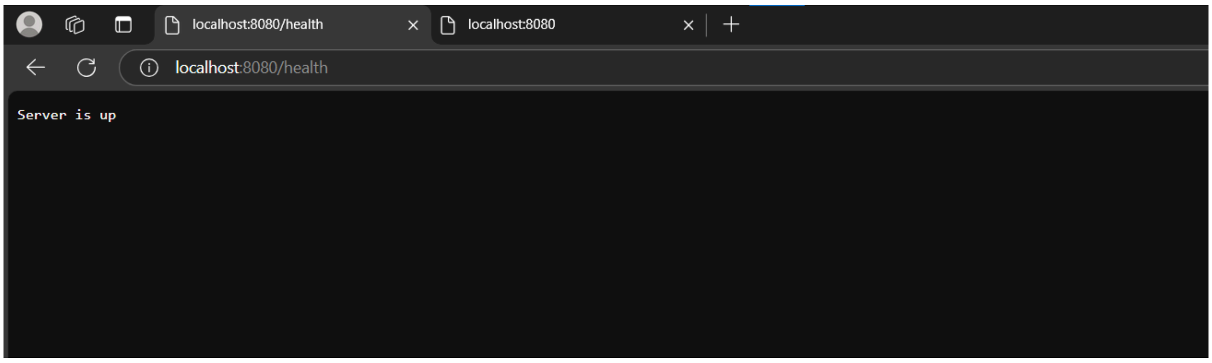
После запуска go run main.go появляются логи в консоли о запуске сервера и работе клиентов



```go
package main

import (
	"encoding/json"
	"fmt"
	"log"
	"math/rand"
	"net/http"
	"os"
	"sync"
	"time"

	"github.com/joho/godotenv"
)

// Названия наших клиентов
const (
	Client1 = "client1"
	Client2 = "client2"
	Client3 = "client3"
)

// ClientStats хранит счётчики по статус-кодам для каждого клиента
type ClientStats struct {
	StatusCount map[int]int `json:"status_count"`
}

// GlobalStats хранит статистику по всем клиентам
```

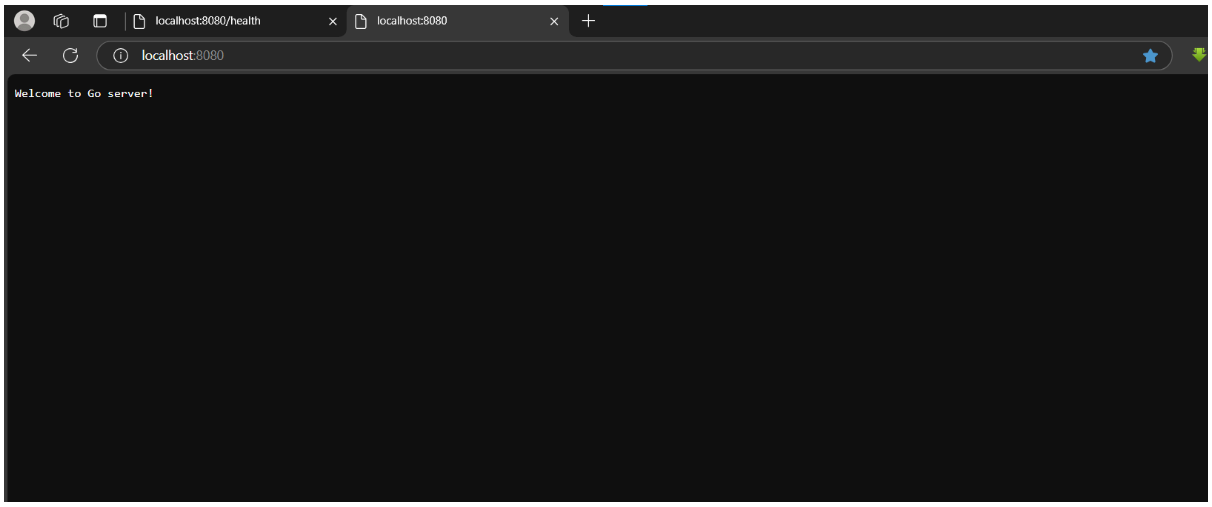
```
PS C:\Users\gosha\OneDrive\GO\Porject_Go> go run main.go
2025/02/02 20:31:03 [client2] starting client with 100 requests total...
2025/02/02 20:31:03 Starting server on :8080 ...
2025/02/02 20:31:03 [client1] starting client with 100 requests total...
2025/02/02 20:31:08 [client3] Server is OK (status=200)
2025/02/02 20:31:13 [client3] Server is OK (status=200)
2025/02/02 20:31:18 [client3] Server is OK (status=200)
2025/02/02 20:31:22 [client2] finished sending 100 requests.
2025/02/02 20:31:22 [client1] finished sending 100 requests.
2025/02/02 20:31:23 [client3] Server is OK (status=200)
2025/02/02 20:31:28 [client3] Server is OK (status=200)
2025/02/02 20:31:33 All clients finished sending requests. You can now GET /stats to see results.
2025/02/02 20:31:33 [client3] Stopping health checker...
2025/02/02 20:31:38 Program finished.
PS C:\Users\gosha\OneDrive\GO\Porject_Go>
```

http://localhost:8080/health возвращает Server is up



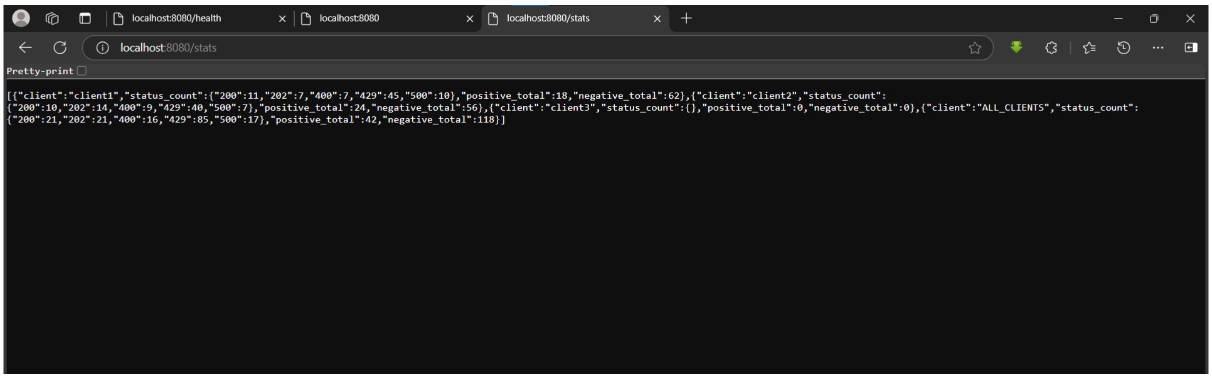
```
Server is up
```

http://localhost:8080/ возвращает Welcome to Go server!



```
Welcome to Go server!
```

http://localhost:8080/stats получаем JSON отчёт по всем клиентам



```
[{"client":"client1","status_count":{"200":11,"202":7,"400":7,"429":45,"500":10},"positive_total":18,"negative_total":62},{"client":"client2","status_count":
{"200":10,"202":14,"400":9,"429":40,"500":7},"positive_total":24,"negative_total":56},{"client":"client3","status_count":{},"positive_total":0,"negative_total":0},{"client":"ALL_CLIENTS","status_count":
{"200":21,"202":21,"400":16,"429":85,"500":17},"positive_total":42,"negative_total":118}]
```

Итого

- Сервер работает на порте, указанном в .env (:8080).

Client1 и Client2 автоматически делают по 100 POST запросов, используя 2 горутины и rate limit 5 req/sec

- Client3 каждые 5 сек проверяет /health
- После ~30 сек отправки, можно GET /stats и увидеть количество положительных/отрицательных ответов
- Сервер тоже ограничен 5 req/sec — если поток запросов превышает, возвращает 429