



Министерство науки и высшего образования Российской Федерации  
Федеральное государственное бюджетное образовательное учреждение  
высшего образования  
«Московский государственный технический университет  
имени Н.Э. Баумана  
(национальный исследовательский университет)»  
(МГТУ им. Н.Э. Баумана)

---

ФАКУЛЬТЕТ ИНФОРМАТИКА И СИСТЕМЫ УПРАВЛЕНИЯ

КАФЕДРА КОМПЬЮТЕРНЫЕ СИСТЕМЫ И СЕТИ (ИУ6)

НАПРАВЛЕНИЕ ПОДГОТОВКИ 09.03.01 Компьютерные системы и сети

**О Т Ч Е Т**

**по лабораторной работе № 6**

**Название:** Основы Back-End разработки на Golang

**Дисциплина:** Языки Интернет-Программирования

Студент ИУ6-31Б  
(Группа)

\_\_\_\_\_  
(Подпись, дата) К.Д. Коротаев  
(И.О. Фамилия)

Преподаватель \_\_\_\_\_

\_\_\_\_\_  
(Подпись, дата) И.О. Фамилия  
(И.О. Фамилия)

Москва, 2024

**Цель работы:** изучение основ сетевого взаимодействия и серверной разработки с использованием языка Golang.

В рамках данной лабораторной работы предлагается продолжить изучение Golang и познакомиться с набором стандартных библиотек, используемых для организации сетевого взаимодействия и разработки серверных приложений.

**Задание 1:** Необходимо написать веб-сервер, который по пути ``/get`` отдает текст ``"Hello, web!"``. Порт должен быть ``:8080``.

**Задание 2:** Необходимо написать веб-сервер, который по пути ``/api/user`` приветствует пользователя. Сервер по этому пути должен принимать и парсить параметр ``name``, после этого отвечая в формате: ``"Hello,<name>!"``. Пример url: ``/api/user?name=Golang``

**Задание 3:** Напиши веб сервер (порт **:3333**) - счетчик который будет обрабатывать GET (`/count`) и POST (`/count`) запросы:

**GET:** возвращает счетчик

**POST:** увеличивает ваш счетчик на значение (с ключом "count") которое вы получаете из формы, но если пришло НЕ число то нужно ответить клиенту: "это не число" со статусом `http.StatusBadRequest (400)`.

## Ход работы:

### Задание 1

Код main.go

```
package main
import (
    "net/http"
)
func main() {
    http.HandleFunc("/get", func(w http.ResponseWriter, r *http.Request) {
        w.Write([]byte("Hello, web!"))
    })
    http.ListenAndServe(":8080", nil)
}
```

PS C:\Users\admin> cd .ssh	PS C:\Users\admin> curl.exe http://localhost:8080/get
PS C:\Users\admin\.ssh> cd web-6/projects	Hello, web!
PS C:\Users\admin\.ssh\web-6\projects> cd 1_hello	PS C:\Users\admin>
PS C:\Users\admin\.ssh\web-6\projects\1_hello> go run main.go	

### Задание 2

Код main.go

```
package main
import (
    "net/http"
)
func main() {
    http.HandleFunc("/api/user", func(w http.ResponseWriter, r *http.Request) {
        name := r.URL.Query().Get("name")
        w.Write([]byte("Hello, "))
        w.Write([]byte(name))
        w.Write([]byte("!"))
    })
    http.ListenAndServe(":9000", nil)
}
```

PS C:\Users\admin> cd .ssh/web-6/projects/2_query	PS C:\Users\admin> curl.exe http://localhost:9000/api/user?name=Golang
PS C:\Users\admin\.ssh\web-6\projects\2_query> go run main.go	Hello, Golang!
	PS C:\Users\admin>

### Задание 3

#### Код main.go

```
package main
import (
    "net/http"
    "strconv"
)
func main() {
    var counter int = 0
    http.HandleFunc("/count", func(w http.ResponseWriter, r *http.Request) {
        switch r.Method {
        case http.MethodGet:
            w.Write([]byte(strconv.Itoa(counter)))
        case http.MethodPost:
            r.ParseForm()
            st := r.Form.Get("count")
            i, wrong := strconv.Atoi(st)
            if wrong != nil {
                w.WriteHeader(http.StatusBadRequest)
                w.Write([]byte("это не число"))
            } else {
                counter = counter + i
            }
        }
    })
    http.ListenAndServe(":3333", nil)
}
```

```
PS C:\Users\admin> cd .ssh/web-6/projects/3_count
PS C:\Users\admin\.ssh\web-6\projects\3_count> go run main.go
```

```
PS C:\Users\admin> curl.exe --data "count=3" http://localhost:3333/count
PS C:\Users\admin> curl.exe http://localhost:3333/count
3
PS C:\Users\admin> curl.exe --data "count=abab" http://localhost:3333/count
это не число
PS C:\Users\admin> curl.exe http://localhost:3333/count
3
PS C:\Users\admin> curl.exe --data "count=-1" http://localhost:3333/count
PS C:\Users\admin> curl.exe http://localhost:3333/count
2
PS C:\Users\admin> |
```

**Заключение:** мы изучили основы сетевого взаимодействия и серверной разработки с использованием языка Golang.

**Список использованных источников:**

1. <https://stepik.org/lesson/1101590/>