



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

ОТЧЕТ

по рубежному контролю № 1

Название:

Разработка WEB-сервера на Golang

Дисциплина: Языки Интернет-программирования

Студент

ИУ6И-31Б  
(Группа)

(Подпись, дата)

С. Гун

(И.О. Фамилия)

Преподаватель

(Подпись, дата)

В.Д. Шульман

(И.О. Фамилия)

Москва, 2024

## # Билет №7. Среднее из трёх

Необходимо написать веб-сервер на GO, решающий задачу "Среднее из трёх". Сервер должен запускаться по адресу `127.0.0.1:8081`.

У сервера должна быть ручка (handler) `POST /middle`. Эта ручка ожидает, что через JSON будет передано 3 параметра типа int: `a`, `b` и `c`.

При обработке http-запроса должно возвращаться то число, которое больше одного, но меньше другого.

В качестве ответа сервер должен возвращать JSON с единственным полем `result`.

Пример запроса (curl):

```
...  
curl --header "Content-Type: application/json" --request POST --data  
'{"a":5,"b":11,"c":10}' http://127.0.0.1:8081/middle  
...
```

Пример ответа:

```
...  
{"result":10}  
...
```

```
1 package main
2
3 import (
4     "encoding/json"
5     "log"
6     "net/http"
7 )
8
9 type Request struct {
10     A int `json:"a"`
11     B int `json:"b"`
12     C int `json:"c"`
13 }
14
15 type Response struct {
16     Result int `json:"result"`
17 }
18
19 func middleHandler(w http.ResponseWriter, r *http.Request) {
20     if r.Method == http.MethodGet {
21         w.Header().Set("Content-Type", "application/json")
22         json.NewEncoder(w).Encode(Response{Result: 10})
23         return
24     }
25
26     if r.Method == http.MethodPost {
27         var req Request
28         err := json.NewDecoder(r.Body).Decode(&req)
29         if err != nil {
30             http.Error(w, "Invalid request body", http.StatusBadRequest)
31             return
32         }
33     }
34 }
```

```

33
34     middle := findMiddle(req.A, req.B, req.C)
35
36     w.Header().Set("Content-Type", "application/json")
37     json.NewEncoder(w).Encode(Response{Result: middle})
38     return
39 }
40
41 http.Error(w, "Invalid request method", http.StatusMethodNotAllowed)
42 }
43
44 func findMiddle(a, b, c int) int {
45     if (a > b && a < c) || (a > c && a < b) {
46         return a
47     }
48     if (b > a && b < c) || (b > c && b < a) {
49         return b
50     }
51     return c
52 }
53
54 func main() {
55     http.HandleFunc("/middle", middleHandler)
56
57     addr := "127.0.0.1:8080"
58     log.Printf("Server is running at %s", addr)
59     log.Fatal(http.ListenAndServe(addr, nil))
60 }

```

(Рис. 1, 2 - Код программы)

```
micheal@Micheal886 web-rk1 % go run main.go
2024/12/14 16:56:54 Server is running at 127.0.0.1:8081
```

(Рис. 3 - Запуск программы)

```
micheal@Micheal886 ~ % 2024/12/14 16:42:59 Server is running at 127.0.0.1:8081
curl --header "Content-Type: application/json" \
  --request POST \
  --data '{"a":5,"b":11,"c":10}' \
  http://127.0.0.1:8081/middle
zsh: no such file or directory: 2024/12/14
{"result":10}
micheal@Micheal886 ~ % 2024/12/14 16:42:59 Server is running at 127.0.0.1:8081
curl --header "Content-Type: application/json" \
  --request POST \
  --data '{"a":5,"b":11,"c":12}' \
  http://127.0.0.1:8081/middle
zsh: no such file or directory: 2024/12/14
{"result":11}
micheal@Micheal886 ~ %
```

(Рис. 4 - Вывод программы)