Midterm Report

Hello and welcome to the report of our online shop project that is named All-in-One. Our website is about the online shop that includes buying products. There is a website that also has registration and authorization that has full-fledged authentication.

Members:

Tlepbergen Abdussamad 200103400: Responsible for the entire database using MySQL database for the website.

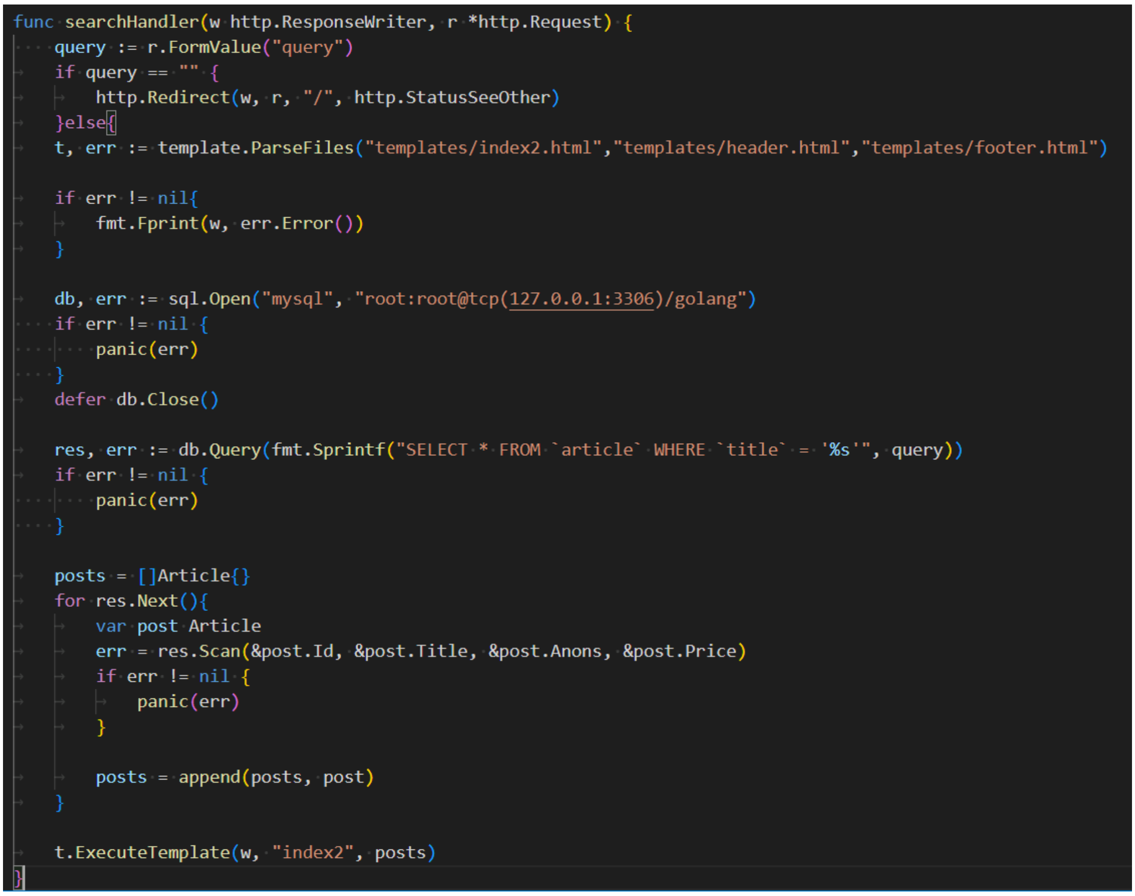
Moldabayev Almat 200103324: Responsible for LogIn and Registration windows for the website.

Tabyldiyev Rauan 200103412: Responsible for scripts that include transitions between authorization and registration windows for the website.

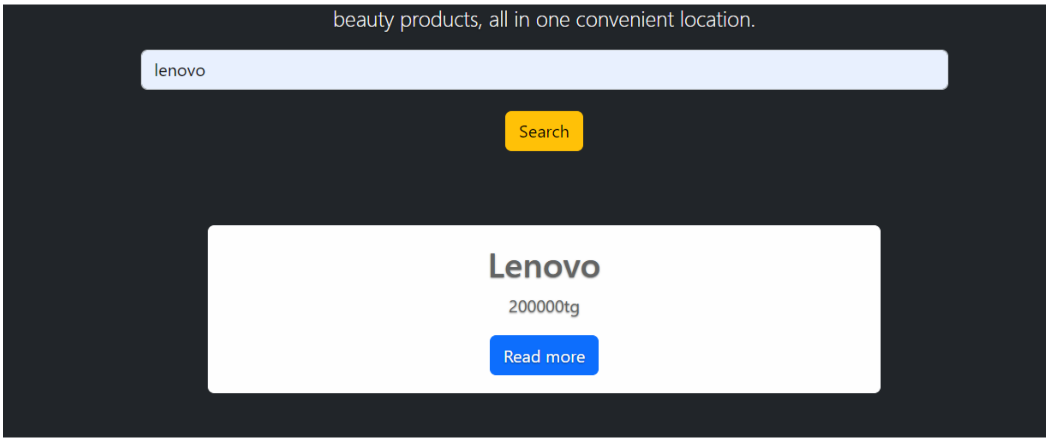
Bulatov Maxat 200103033: Responsible for the Main Page and Footer using HTML and CSS for the website.

How to run code: First you have to run MAMP, for the database. Create in phpMyAdmin a structure called "golang". Inside this structure create two tables: "article" and "login". Inside "article" add such rows as "id", "title", "anons", "price". Inside "login" add such rows as "id", "name", "email", "password". Then open our code and run main.go with the command "go run main.go". Then through the browser we open "localhost:8080"

Search items by name



```go
func searchHandler(w http.ResponseWriter, r *http.Request) {
    query := r.FormValue("query")
    if query == "" {
        http.Redirect(w, r, "/", http.StatusSeeOther)
    }else{
    t, err := template.ParseFiles("templates/index2.html","templates/header.html","templates/footer.html")

    if err != nil{
        fmt.Fprint(w, err.Error())
    }

    db, err := sql.Open("mysql", "root:root@tcp(127.0.0.1:3306)/golang")
    if err != nil {
        panic(err)
    }
    defer db.Close()

    res, err := db.Query(fmt.Sprintf("SELECT * FROM `article` WHERE `title` = '%s'", query))
    if err != nil {
        panic(err)
    }

    posts = []Article{}
    for res.Next(){
        var post Article
        err = res.Scan(&post.Id, &post.Title, &post.Anons, &post.Price)
        if err != nil {
            panic(err)
        }

        posts = append(posts, post)
    }

    t.ExecuteTemplate(w, "index2", posts)
}
```



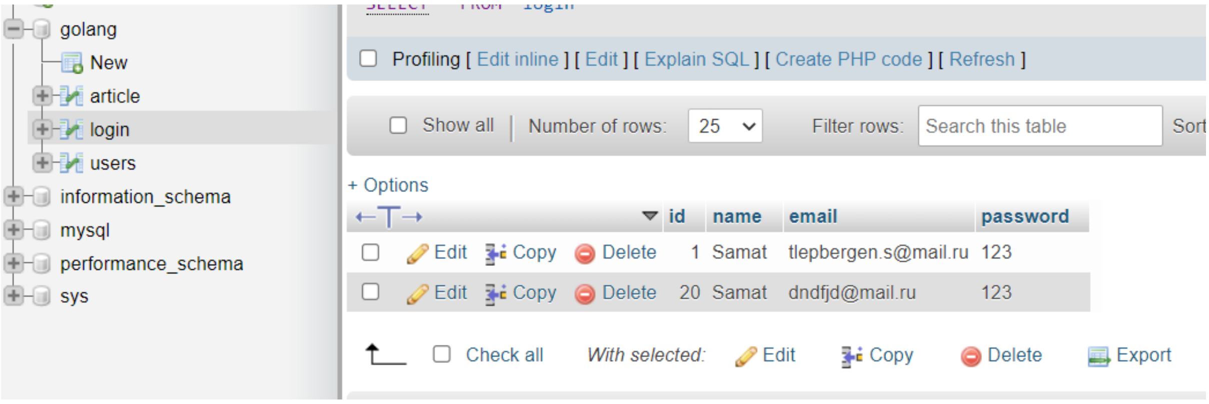
beauty products, all in one convenient location.

lenovo

Lenovo

200000tg

Login



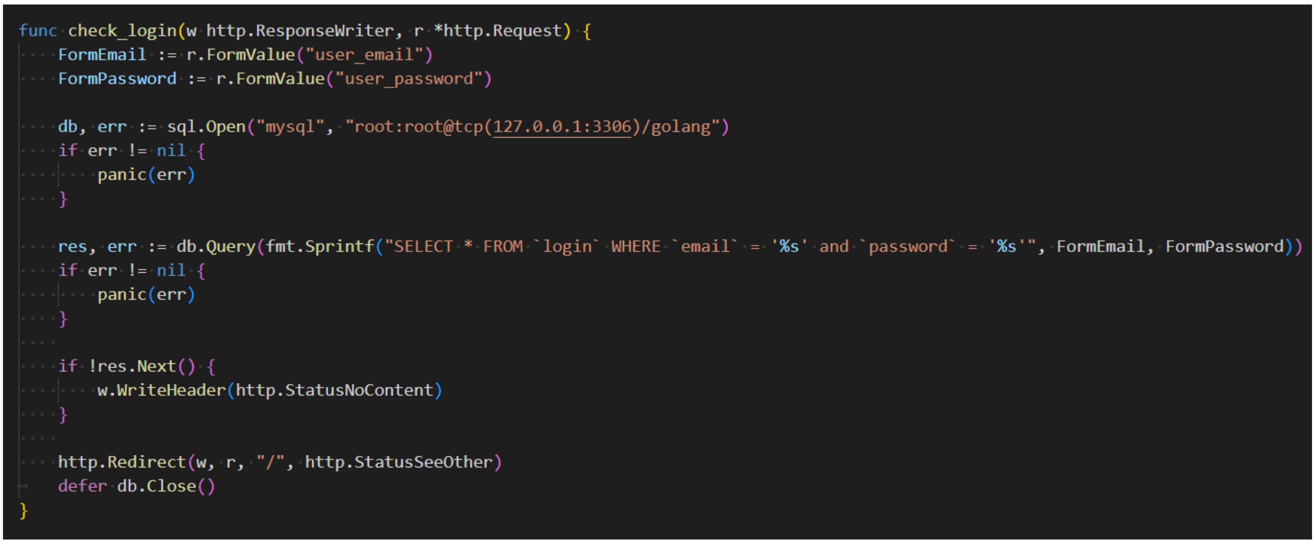
□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort

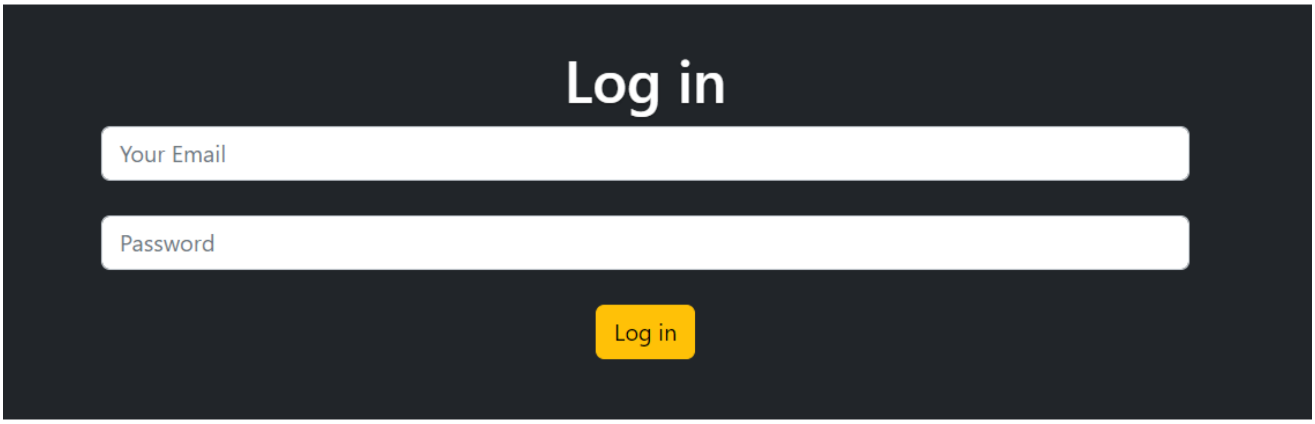
+ Options

				id	name	email	password
□	Edit	Copy	Delete	1	Samat	tlepbergen.s@mail.ru	123
□	Edit	Copy	Delete	20	Samat	dndfjd@mail.ru	123

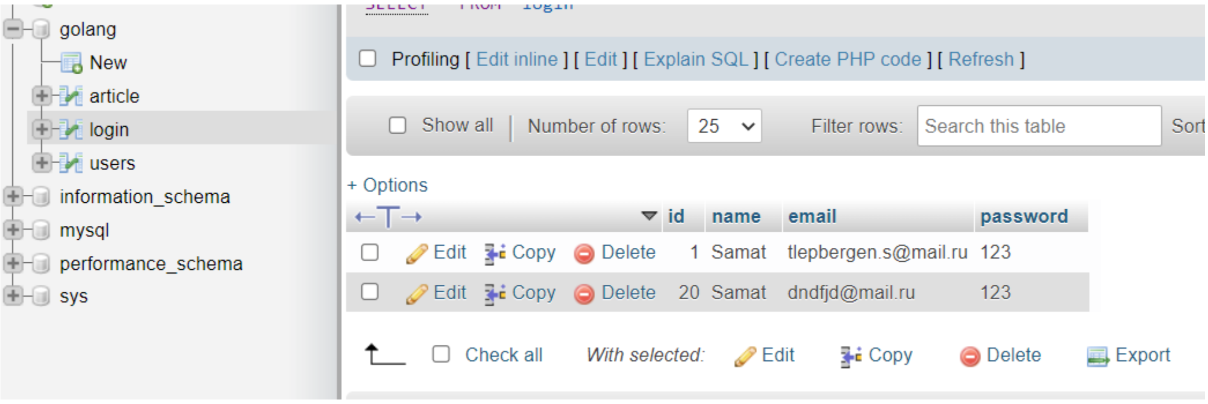
□ Check all *With selected:* Edit Copy Delete Export



```go
func check_login(w http.ResponseWriter, r *http.Request) {
    FormEmail := r.FormValue("user_email")
    FormPassword := r.FormValue("user_password")

    db, err := sql.Open("mysql", "root:root@tcp(127.0.0.1:3306)/golang")
    if err != nil {
        panic(err)
    }

    res, err := db.Query(fmt.Sprintf("SELECT * FROM `login` WHERE `email` = '%s' and `password` = '%s'", FormEmail, FormPassword))
    if err != nil {
        panic(err)
    }

    if !res.Next() {
        w.WriteHeader(http.StatusNoContent)
    }

    http.Redirect(w, r, "/", http.StatusSeeOther)
    defer db.Close()
}
```



Registration



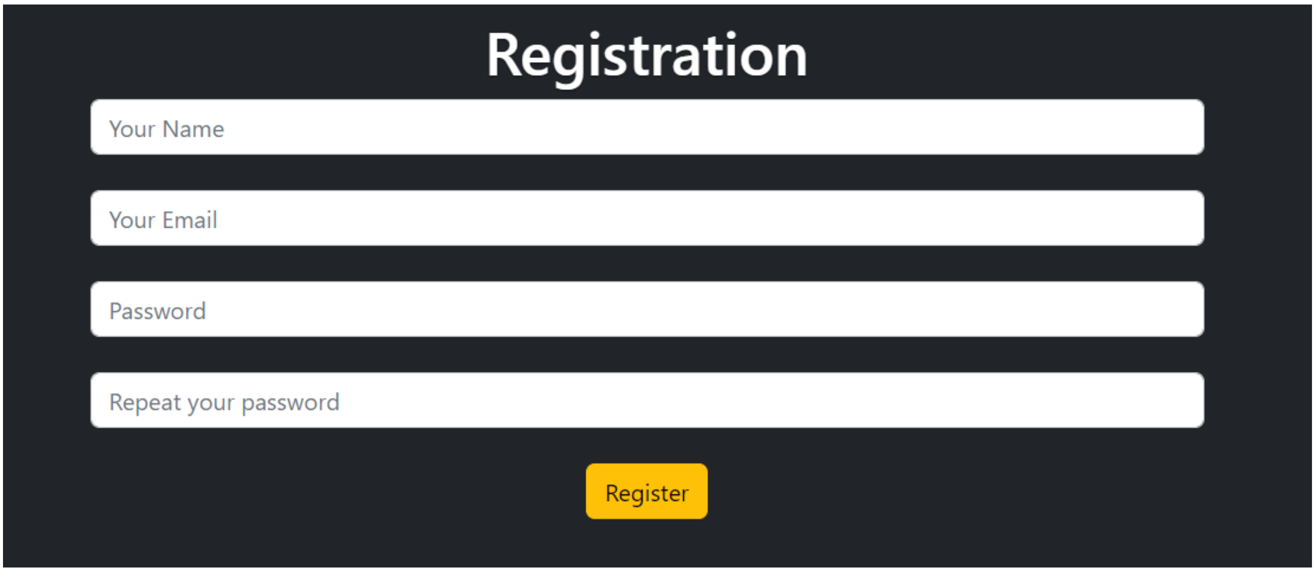
☐ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

☐ Show all | Number of rows: 25 Filter rows: Search this table Sort

+ Options

←T→	id	name	email	password
☐ Edit Copy Delete	1	Samat	tlepbergen.s@mail.ru	123
☐ Edit Copy Delete	20	Samat	dndfjd@mail.ru	123

☐ Check all *With selected:* Edit Copy Delete Export



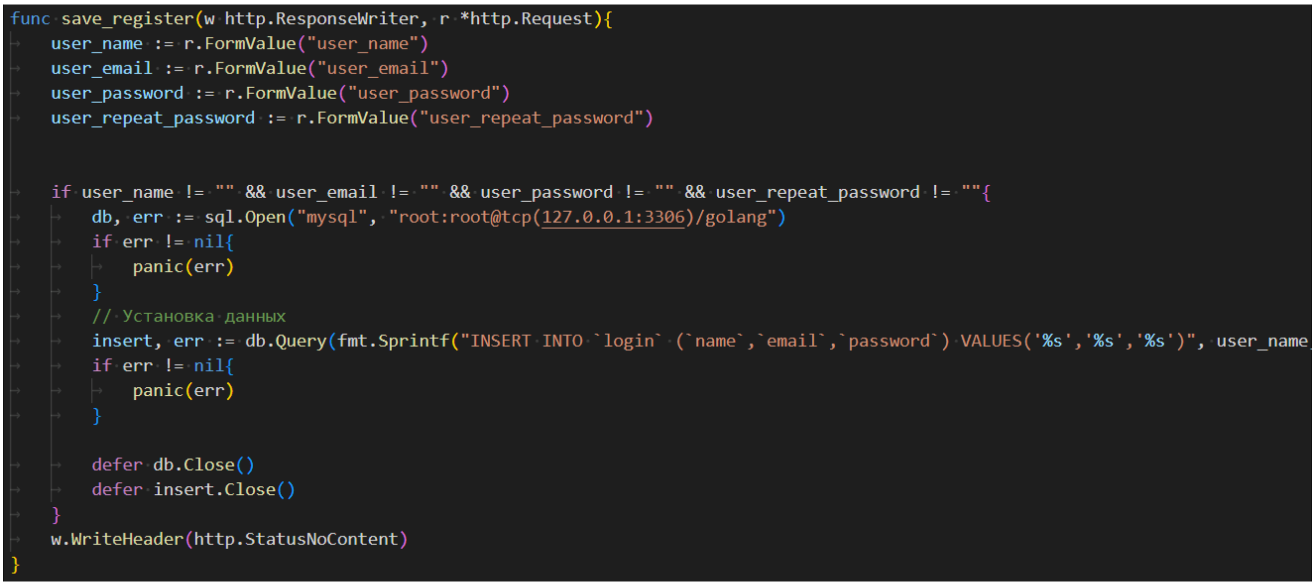
Registration

Your Name

Your Email

Password

Repeat your password



```go
func save_register(w http.ResponseWriter, r *http.Request){
    user_name := r.FormValue("user_name")
    user_email := r.FormValue("user_email")
    user_password := r.FormValue("user_password")
    user_repeat_password := r.FormValue("user_repeat_password")


    if user_name != "" && user_email != "" && user_password != "" && user_repeat_password != ""{
        db, err := sql.Open("mysql", "root:root@tcp(127.0.0.1:3306)/golang")
        if err != nil{
            panic(err)
        }
        // Установка данных
        insert, err := db.Query(fmt.Sprintf("INSERT INTO `login` (`name`,`email`,`password`) VALUES('%s','%s','%s')", user_name
        if err != nil{
            panic(err)
        }

        defer db.Close()
        defer insert.Close()
    }
    w.WriteHeader(http.StatusNoContent)
}
```