

# DOKUMENTACIJA

Praktikum iz web programiranja PHP

Mateja Rastoder 82/21

Smer: Internet tehnologije

Modul: Web programiranje

Predmet: Praktikum Web Programiranje PHP

Kredencijali

admin: admin@gmail.com ; Admin123

User: korisnik@gmail.com ; Korisnik123

## Contents

1 Uvod .....	4
1.1 programski jezici .....	4
1.2 Opis funkcionalnosti .....	4
1.3 Template .....	4
2 Organizacija .....	5
3 Izgled i opis stranica .....	6
3.1 index.php .....	6
3.2 shop .....	7
3.3 about .....	8
3.4 contact .....	9
4. Baza podataka .....	13
5. Kod .....	27
Index.php .....	27
.env .....	28
Config.php .....	28
Connection.php .....	29
Head.php .....	29
Header.php .....	31
heroArea.php .....	32
Scripts.php .....	33
Navigation.php .....	34
About.php .....	35
Adminpanel.php .....	36
Cart.php .....	39
Contact.php .....	39
Home.php .....	41
Login.php .....	44
Product.php .....	46
Register.php .....	46
Shop.php .....	49
Models .....	51
Brisanje u admin panelu .....	51
getAll funkcije .....	54
Helper.php .....	63
Insert funkcije .....	63

Login check .....	71
Logout.....	72
registration insert.....	72
Verify registration .....	75
Css .....	76
Main.js.....	77
Main js iz visual studio code (prethodni je menjan direktno na 000webhost pa je prekopiran plain text) .....	120

# 1 Uvod

## 1.1 programski jezici

Za izradu sajta su korišćeni PHP, HTML, CSS, Javascript, Jquery biblioteka javascripta i Bootstrap.

## 1.2 Opis funkcionalnosti

- Dinamicko ispisivanje headera i footera zajedno sa navigacionim menijem unutar headera
- Dinamicko ispisivanje proizvoda iz baze podataka.
- Sortiranje i filtriranje proizvoda po odredjenim kriterijumima koje bira korisnik
- "Scroll to top" dugme na dnu svake stranice koje vraća korisnika na vrh
- Uz bootstrap su svi elementi stranica prilagođeni veličini uređaja
- Hvatanje grešaka pri komunikaciji sa bazom
- Provera unosa na klijentskoj i serverskoj strani za register i login stranicu.
- Provera unosa na admin panelu za sve stavke.
- Navigacija se pretvara u toggle dugme na manjim veličinama ekrana.

## 1.3 Template

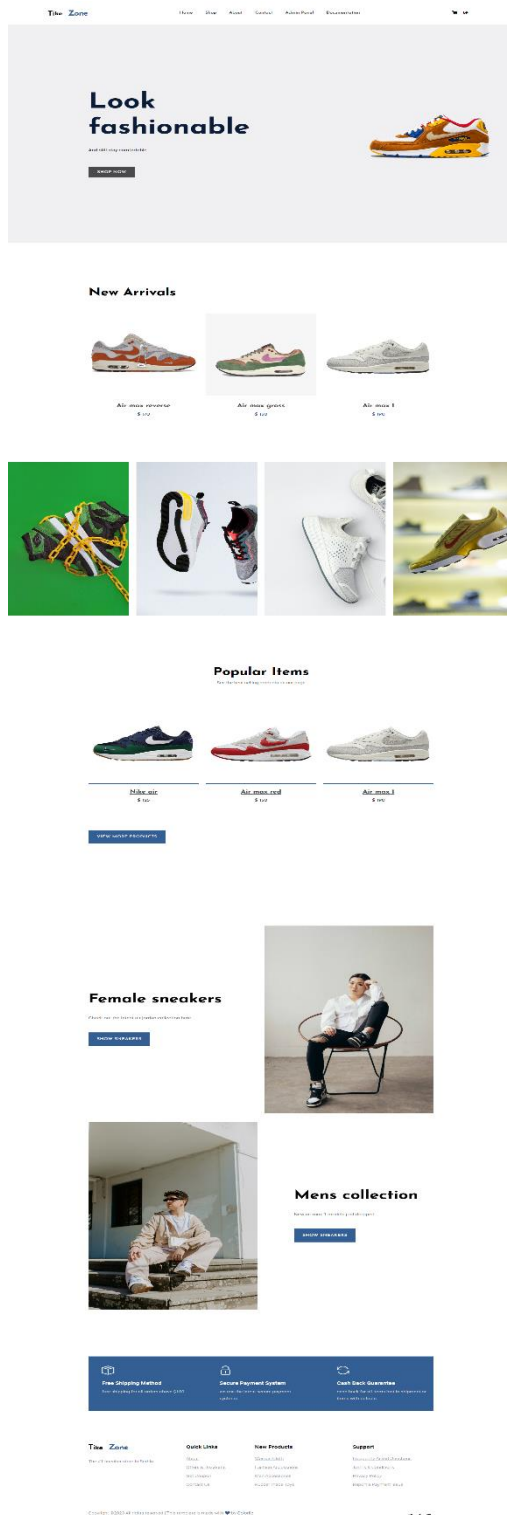
Za izgled sajta je koriscen colorlib TimeZone template [WordPress Themes & Website Templates - Colorlib](#)

## 2 Organizacija

▼ assets	●
> css	●
> fonts	
> img	●
> js	●
> scss	
> webfonts	
▼ config	●
⚙ .env	U
🐘 config.php	U
🐘 connection.php	U
▼ data	●
≡ log.txt	U
> models	●
▼ views	●
▼ fixed	●
🐘 footer.php	U
🐘 head.php	U
🐘 header.php	U
🐘 heroArea.php	U
🐘 scripts.php	U
▼ pages	●
🐘 about.php	U
🐘 adminpanel.php	U
🐘 cart.php	U
🐘 contact.php	U
🐘 home.php	U
🐘 login.php	U
🐘 product.php	U
🐘 register.php	U
🐘 shop.php	U
📄 documentation.pdf	U
🐘 index.php	M

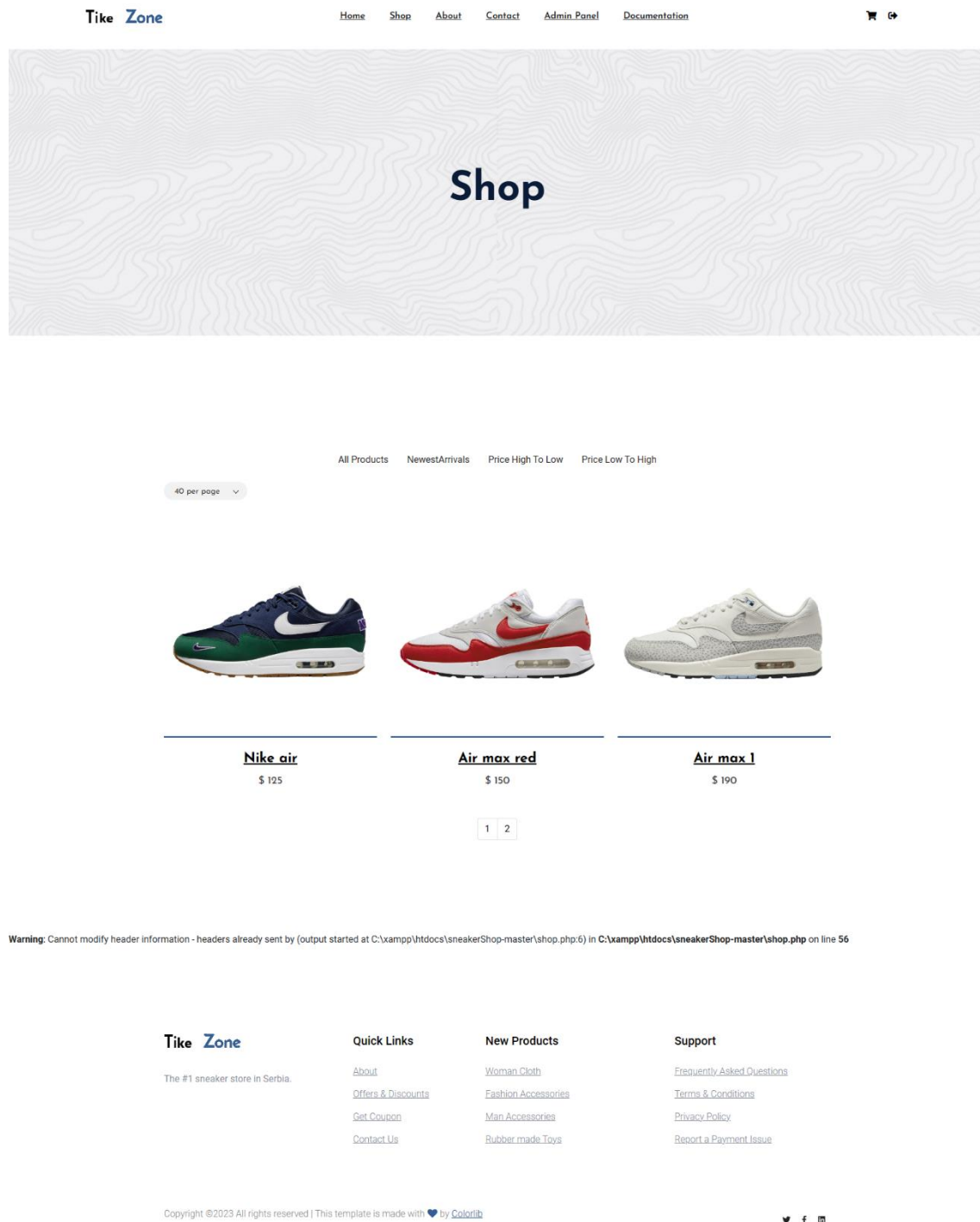
### 3 Izgled i opis stranica

#### 3.1 index.php



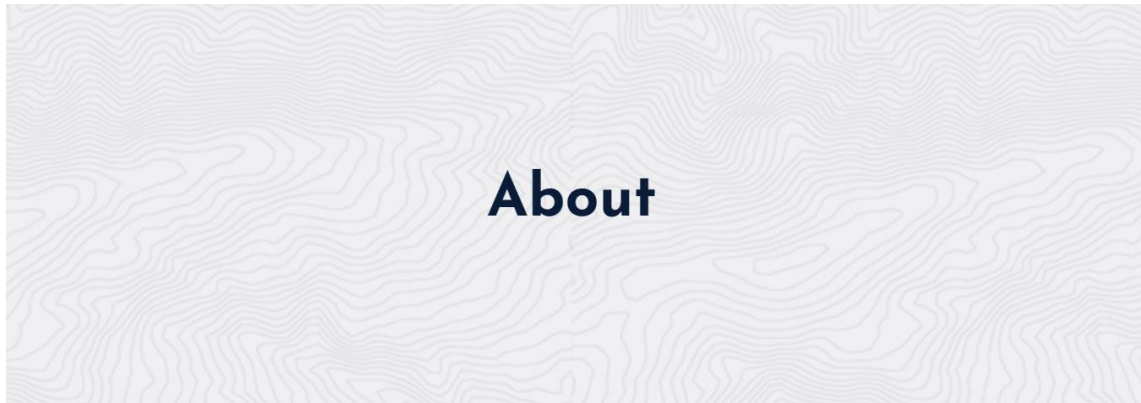
Na pocetnoj strani korisnike čeka baner koji menja slike banera kolekcija na određeni vremenski period. New arrivals sekcija prikazuje 3 najnovija proizvoda i tekođe postoji sekcija koja prikazuje najprodavanije proizvode koji su određeni brojem narudžbina iz tabele orders u bazi.

## 3.2 shop



Na ovoj stranici se prikazuju svi proizvodi i straničenje se obavlja u grupama po 3 ili 6 u zavisnosti sta korisnik izabere. Takođe se uz straničenje obavlja i sortiranje po prikazanim kriterijumima.

### 3.3 about



My name is Mateja Rastoder. I'm a 20 year old programmer from Serbia. I'm currently enrolled in the ICT college located in Belgade. This website is a project made for one of my subjects.

Jednostavna about stranica sa dinamickim banerom koji se ispisuje na svakoj stranici osim home.



### 3.4 contact

#### Get in Touch


Enter Message


Enter your name


Email

Enter Subject

SEND

  
**Buttonwood, California.**  
Rosemead, CA 91770

  
**+1 253 565 2365**  
Mon to Fri 9am to 6pm

  
**support@colorlib.com**  
Send us your query anytime!

Contact stranica sa proverom unosa na klijentskoj strani.

3.5 login

New to our Shop?

There are advances being made in science and technology everyday, and a good example of this is the

CREATE AN ACCOUNT

Welcome Back !  
Please Sign in now

admin@gmail.com

Enter valid email.

\*\*\*\*\*

Incorrect password.

LOG IN

3.6  
Register

Welcome !  
Please Register Now

Name

Lastname





Email

Password

Re-enter password

REGISTER

3.7 cart

Product	Image	Qty	Price	Total	Remove
Nike air		1	\$125	\$125	<button>REMOVE</button>
Air max red		2	\$150	\$300	<button>REMOVE</button>
Air max 1		2	\$190	\$380	<button>REMOVE</button>
Air max reverse		2	\$170	\$340	<button>REMOVE</button>
total: \$1145					
<button>COMPLETE PURCHASE</button>					

Discoversoft Inc. 

Izracunate cene proizvoda i ukupna cena korpe. Mogu se izbacivati ili dodavati pojedinačni proizvodi a na complete purchase se simulira završena kupovina.

### 3.8 admin panel

Welcome Admin

Last 24h of user activity

Home page

7.14%

Documentation page

3.97%

Admin panel page

7.14%

Shop page

19.05%

About page







3.17%

Contact page

2.3%

Prikazuje se aktivnost korisnika iz log fajla

ADD NEW PRODUCTS

id	name	price	old_price	img	description	gender	brand	dateofadd	Edit	Delete
1	Nike air	125	225		Opis nekili	man	Nike	2023-05-04 14:33:26		
15	Air max red	150	200		Air max 1	man	Nike	2023-05-11 21:40:20		

i stranica za upravljanje svim delovima sajta.

#### 4. Baza podataka

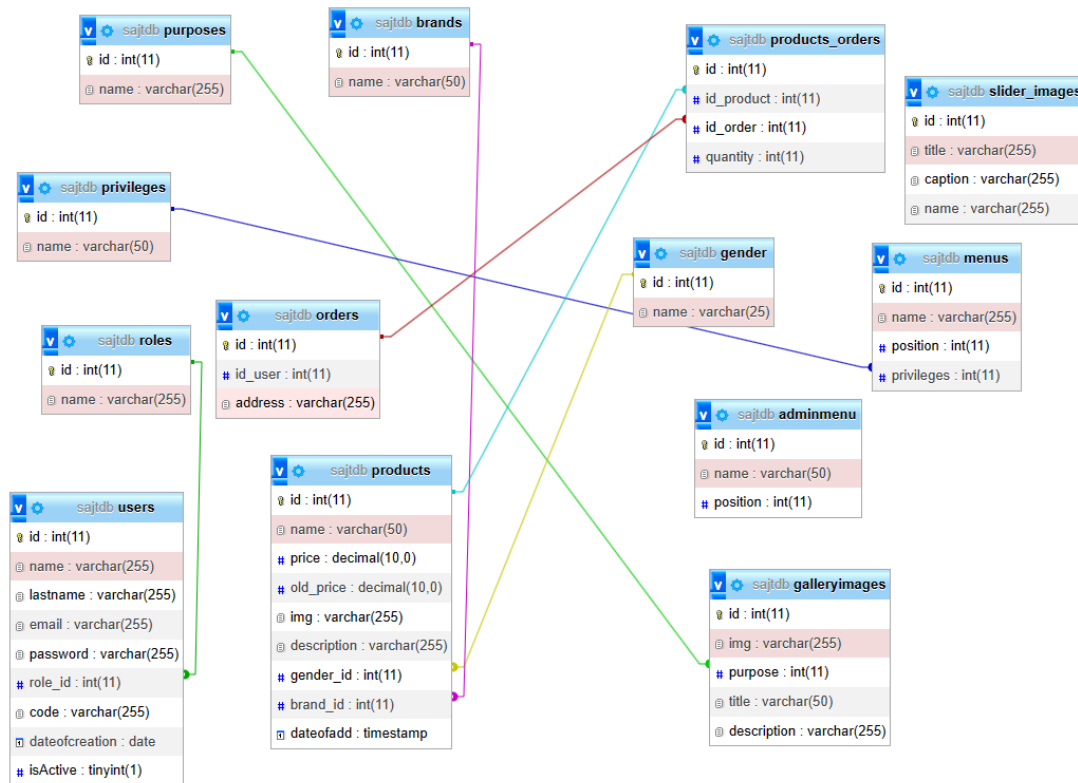


Tabela users sadrži korisnike koji mogu biti administrator ili korisnici. User mora biti aktivan da bi mogao da se uloguje na sajt(row active da bude 1).

U tabeli roles se nalaze uloge korisnika.

Tabela products sadrži sve informacije o proizvodima i ima pomoćne tabele brands i products orders koje sadrže nazive brendova i broj narudžbina proizvoda.

Preko admin panela možemo dodati ili obrisati nove korisnike, proizvode, stranice, brendove i polove ako npr. bude potrebno za dodavanje unisex modela proizvoda.

Tabele galleryimages i slider\_images su dodate kako bi se slike za slajder i galeriju promocija učitala iz baze.

```
-- phpMyAdmin SQL Dump
-- version 5.2.0
-- https://www.phpmyadmin.net/
--
-- Host: 127.0.0.1
-- Generation Time: May 12, 2023 at 03:28 PM
-- Server version: 10.4.27-MariaDB
-- PHP Version: 8.2.0
```

```
SET SQL_MODE = "NO_AUTO_VALUE_ON_ZERO";
```

```

START TRANSACTION;
SET time_zone = "+00:00";

/*!40101 SET @OLD_CHARACTER_SET_CLIENT=@@CHARACTER_SET_CLIENT */;
/*!40101 SET @OLD_CHARACTER_SET_RESULTS=@@CHARACTER_SET_RESULTS */;
/*!40101 SET @OLD_COLLATION_CONNECTION=@@COLLATION_CONNECTION */;
/*!40101 SET NAMES utf8mb4 */;

--
-- Database: `sajtdb`
--

--
-- -----
--
-- Table structure for table `adminmenu`
--

CREATE TABLE `adminmenu` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `position` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `adminmenu`
--

INSERT INTO `adminmenu` (`id`, `name`, `position`) VALUES
(1, 'products', 1),
(2, 'users', 2),
(3, 'brands', 3),
(4, 'genders', 4),
(5, 'menus', 5);

--
-- -----
--
-- Table structure for table `brands`
--

CREATE TABLE `brands` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `brands`
--

```

```
--

INSERT INTO `brands` (`id`, `name`) VALUES
(1, 'Adidas'),
(2, 'Nike');

-- -----

--
-- Table structure for table `galleryimages`
--

CREATE TABLE `galleryimages` (
  `id` int(11) NOT NULL,
  `img` varchar(255) NOT NULL,
  `purpose` int(11) NOT NULL,
  `title` varchar(50) DEFAULT NULL,
  `description` varchar(255) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `galleryimages`
--

INSERT INTO `galleryimages` (`id`, `img`, `purpose`, `title`,
`description`) VALUES
(1, 'promo1.png', 1, 'Female sneakers', 'Check out the latest air jordan
collection here'),
(2, 'promo2.png', 1, 'Mens collection', 'New air max 1 models just
dropped'),
(3, 'gallery1.png', 2, NULL, NULL),
(4, 'gallery2.png', 2, NULL, NULL),
(5, 'gallery5edit.png', 2, NULL, NULL),
(6, 'gallery6edit.png', 2, NULL, NULL);

-- -----

--
-- Table structure for table `gender`
--

CREATE TABLE `gender` (
  `id` int(11) NOT NULL,
  `name` varchar(25) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `gender`
--
```

```

INSERT INTO `gender` (`id`, `name`) VALUES
(1, 'man'),
(2, 'woman');

-- -----

--
-- Table structure for table `menus`
--

CREATE TABLE `menus` (
  `id` int(11) NOT NULL,
  `name` varchar(255) NOT NULL,
  `position` int(11) NOT NULL,
  `privileges` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `menus`
--

INSERT INTO `menus` (`id`, `name`, `position`, `privileges`) VALUES
(1, 'home', 1, 1),
(2, 'shop', 2, 2),
(3, 'about', 3, 1),
(4, 'contact', 4, 1),
(5, 'admin Panel', 5, 3),
(8, 'Documentation', 6, 1);

-- -----

--
-- Table structure for table `orders`
--

CREATE TABLE `orders` (
  `id` int(11) NOT NULL,
  `id_user` int(11) NOT NULL,
  `address` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `orders`
--

INSERT INTO `orders` (`id`, `id_user`, `address`) VALUES
(1, 2, 'Adresa 1'),
(2, 3, 'adresa 3'),

```



```

(3, 5, 'adresa2');

-- -----

--
-- Table structure for table `paymentoffers`
--

CREATE TABLE `paymentoffers` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `description` varchar(255) NOT NULL,
  `img` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `paymentoffers`
--

INSERT INTO `paymentoffers` (`id`, `name`, `description`, `img`) VALUES
(1, 'Free Shipping Method', 'free shipping for all orders above $100',
'ti-package'),
(2, 'Secure Payment System', 'we use the latest secure payment systems ',
'ti-unlock'),
(3, 'Cash Back Guarantee', 'cash back for all items lost in shipment or
items with defects\r\n', 'ti-reload');

-- -----

--
-- Table structure for table `privileges`
--

CREATE TABLE `privileges` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `privileges`
--

INSERT INTO `privileges` (`id`, `name`) VALUES
(1, 'unauthorized user'),
(2, 'authorized user'),
(3, 'admin');

-- -----

```

```

--
-- Table structure for table `products`
--

CREATE TABLE `products` (
  `id` int(11) NOT NULL,
  `name` varchar(50) NOT NULL,
  `price` decimal(10,0) NOT NULL,
  `old_price` decimal(10,0) DEFAULT NULL,
  `img` varchar(255) NOT NULL,
  `description` varchar(255) NOT NULL,
  `gender_id` int(11) NOT NULL,
  `brand_id` int(11) NOT NULL,
  `dateofadd` timestamp NOT NULL DEFAULT current_timestamp()
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `products`
--

INSERT INTO `products` (`id`, `name`, `price`, `old_price`, `img`,
`description`, `gender_id`, `brand_id`, `dateofadd`) VALUES
(1, 'Nike air ', '125', '225', 'nike1.jpg', 'Opis nekiii', 1, 2, '2023-
05-04 14:33:26'),
(15, 'Air max red', '150', '200', 'nike2.jpg', 'Air max 1', 1, 2, '2023-
05-11 21:40:20'),
(16, 'Air max 1', '190', '190', 'nike4.jpg', 'White air max 1s', 2, 2,
'2023-05-11 21:41:06');

-----

--
-- Table structure for table `products_orders`
--

CREATE TABLE `products_orders` (
  `id` int(11) NOT NULL,
  `id_product` int(11) NOT NULL,
  `id_order` int(11) NOT NULL,
  `quantity` int(11) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `products_orders`
--

INSERT INTO `products_orders` (`id`, `id_product`, `id_order`,
`quantity`) VALUES
(1, 16, 1, 4),

```

```

(2, 15, 3, 2),
(3, 1, 2, 5);

-- -----

--
-- Table structure for table `purposes`
--

CREATE TABLE `purposes` (
  `id` int(11) NOT NULL,
  `name` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `purposes`
--

INSERT INTO `purposes` (`id`, `name`) VALUES
(1, 'sneakersOfChoice'),
(2, 'galleryArea');

-- -----

--
-- Table structure for table `roles`
--

CREATE TABLE `roles` (
  `id` int(11) NOT NULL,
  `name` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `roles`
--

INSERT INTO `roles` (`id`, `name`) VALUES
(1, 'admin'),
(2, 'user');

-- -----

--
-- Table structure for table `slider_images`
--

CREATE TABLE `slider_images` (
  `id` int(11) NOT NULL,

```

```

    `title` varchar(255) NOT NULL,
    `caption` varchar(255) NOT NULL,
    `name` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `slider_images`
--

INSERT INTO `slider_images` (`id`, `title`, `caption`, `name`) VALUES
(1, 'Select your new perfect style', 'Perfect sneakers for any
lifestyle', 'sliderImage1.png'),
(2, 'Look fashionable', 'And still stay comfortable',
'sliderImage2.png');

--
-----

--
-- Table structure for table `socials`
--

CREATE TABLE `socials` (
  `id` int(11) NOT NULL,
  `img` varchar(255) NOT NULL,
  `link` varchar(255) NOT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `socials`
--

INSERT INTO `socials` (`id`, `img`, `link`) VALUES
(1, 'fab fa-twitter', 'https://twitter.com/home'),
(2, 'fab fa-facebook-f', 'https://www.facebook.com/'),
(3, 'fab fa-linkedin', 'https://www.linkedin.com/');

--
-----

--
-- Table structure for table `users`
--

CREATE TABLE `users` (
  `id` int(11) NOT NULL,
  `name` varchar(255) NOT NULL,
  `lastname` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `role_id` int(11) NOT NULL,

```

```

    `code` varchar(255) NOT NULL,
    `dateofcreation` date NOT NULL,
    `isActive` tinyint(1) DEFAULT NULL
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4 COLLATE=utf8mb4_general_ci;

--
-- Dumping data for table `users`
--

INSERT INTO `users` (`id`, `name`, `lastname`, `email`, `password`,
`role_id`, `code`, `dateofcreation`, `isActive`) VALUES
(1, 'Mateja', 'Rastoder ', 'mateja@gmail.com',
'8aa87050051efe26091a13dbfdf901c6', 1, '', '0000-00-00', 1),
(13, 'Admin', 'Admin', 'admin@gmail.com',
'e64b78fc3bc91bcb7dc232ba8ec59e0', 1,
'4da193ac09ad17b6c202cfb377d9865396c2cf3a', '2023-05-04', 1),
(14, 'Korisnik', 'Korisnik', 'korisnik@gmail.com',
'60c5b02d7e7c682f4d56a64d99b9731a', 2,
'd5602cb351134b86e93e8626650101baab230625', '2023-05-04', 1),
(15, 'Amteja', 'Amtoder', 'amteja@gmail.com',
'f5ebb11c65ce6bf63d43b88d1129c8ca', 2,
'9f78ad6e7879b0a283ad58a9e507fa62898d041d', '2023-05-11', 0),
(16, 'Korisnik', 'Korisnikprezime', 'korisnik5@gmail.com',
'60c5b02d7e7c682f4d56a64d99b9731a', 2,
'582bc5a95a3a5d2d324595bf5c55bfd657478e6a', '2023-05-11', 0);

--
-- Indexes for dumped tables
--

--
-- Indexes for table `adminmenu`
--
ALTER TABLE `adminmenu`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `brands`
--
ALTER TABLE `brands`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `galleryimages`
--
ALTER TABLE `galleryimages`
  ADD PRIMARY KEY (`id`),
  ADD KEY `purpose` (`purpose`);

```

```
--
-- Indexes for table `gender`
--
ALTER TABLE `gender`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `menus`
--
ALTER TABLE `menus`
  ADD PRIMARY KEY (`id`),
  ADD KEY `privileges` (`privileges`);

--
-- Indexes for table `orders`
--
ALTER TABLE `orders`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `paymentoffers`
--
ALTER TABLE `paymentoffers`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `privileges`
--
ALTER TABLE `privileges`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `products`
--
ALTER TABLE `products`
  ADD PRIMARY KEY (`id`),
  ADD KEY `gender_id` (`gender_id`),
  ADD KEY `brand_id` (`brand_id`);

--
-- Indexes for table `products_orders`
--
ALTER TABLE `products_orders`
  ADD PRIMARY KEY (`id`),
  ADD KEY `id_product` (`id_product`),
  ADD KEY `id_order` (`id_order`);

--
-- Indexes for table `purposes`
```

```
--
ALTER TABLE `purposes`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `roles`
--
ALTER TABLE `roles`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `slider_images`
--
ALTER TABLE `slider_images`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `socials`
--
ALTER TABLE `socials`
  ADD PRIMARY KEY (`id`);

--
-- Indexes for table `users`
--
ALTER TABLE `users`
  ADD PRIMARY KEY (`id`),
  ADD KEY `role_id` (`role_id`);

--
-- AUTO_INCREMENT for dumped tables
--

--
-- AUTO_INCREMENT for table `adminmenu`
--
ALTER TABLE `adminmenu`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;

--
-- AUTO_INCREMENT for table `brands`
--
ALTER TABLE `brands`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;

--
-- AUTO_INCREMENT for table `galleryimages`
--
ALTER TABLE `galleryimages`
```

```
MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=7;

--
-- AUTO_INCREMENT for table `gender`
--
ALTER TABLE `gender`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=6;

--
-- AUTO_INCREMENT for table `menus`
--
ALTER TABLE `menus`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=9;

--
-- AUTO_INCREMENT for table `orders`
--
ALTER TABLE `orders`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT for table `paymentoffers`
--
ALTER TABLE `paymentoffers`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT for table `privileges`
--
ALTER TABLE `privileges`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT for table `products`
--
ALTER TABLE `products`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=17;

--
-- AUTO_INCREMENT for table `products_orders`
--
ALTER TABLE `products_orders`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=11;

--
-- AUTO_INCREMENT for table `purposes`
--
ALTER TABLE `purposes`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;
```



```

--
-- AUTO_INCREMENT for table `roles`
--
ALTER TABLE `roles`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=3;

--
-- AUTO_INCREMENT for table `slider_images`
--
ALTER TABLE `slider_images`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT for table `socials`
--
ALTER TABLE `socials`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=4;

--
-- AUTO_INCREMENT for table `users`
--
ALTER TABLE `users`
  MODIFY `id` int(11) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=17;

--
-- Constraints for dumped tables
--

--
-- Constraints for table `galleryimages`
--
ALTER TABLE `galleryimages`
  ADD CONSTRAINT `galleryimages_ibfk_1` FOREIGN KEY (`purpose`)
REFERENCES `purposes` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `menus`
--
ALTER TABLE `menus`
  ADD CONSTRAINT `menus_ibfk_1` FOREIGN KEY (`privileges`) REFERENCES
`privileges` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `products`
--
ALTER TABLE `products`
  ADD CONSTRAINT `products_ibfk_1` FOREIGN KEY (`gender_id`) REFERENCES
`gender` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,

```

```
    ADD CONSTRAINT `products_ibfk_2` FOREIGN KEY (`brand_id`) REFERENCES
`brands` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `products_orders`
--
ALTER TABLE `products_orders`
  ADD CONSTRAINT `products_orders_ibfk_1` FOREIGN KEY (`id_product`)
REFERENCES `products` (`id`) ON DELETE CASCADE ON UPDATE CASCADE,
  ADD CONSTRAINT `products_orders_ibfk_2` FOREIGN KEY (`id_order`)
REFERENCES `orders` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;

--
-- Constraints for table `users`
--
ALTER TABLE `users`
  ADD CONSTRAINT `users_ibfk_1` FOREIGN KEY (`role_id`) REFERENCES
`roles` (`id`) ON DELETE CASCADE ON UPDATE CASCADE;
COMMIT;

/*!40101 SET CHARACTER_SET_CLIENT=@OLD_CHARACTER_SET_CLIENT */;
/*!40101 SET CHARACTER_SET_RESULTS=@OLD_CHARACTER_SET_RESULTS */;
/*!40101 SET COLLATION_CONNECTION=@OLD_COLLATION_CONNECTION */;
```

## 5. Kod

Index.php

```
<?php
session_start();

require_once "views/fixed/head.php";

if (isset($_SESSION['user'])) {
    $user = $_SESSION['user'];
}

require_once "views/fixed/header.php";

$page = "";
$product = "";

if (isset($_GET['page'])) {
    $page = $_GET['page'];
}

if(isset($_GET['product'])){
    $product = $_GET['product'];
}

switch ($page) {

    case "home":
        include 'views/pages/home.php';
        break;
    case "shop":
        include 'views/pages/shop.php';
        break;
    case "about":
        include 'views/pages/about.php';
        break;
    case "contact":
        include 'views/pages/contact.php';
        break;
    case "admin":
        include 'views/pages/admin.php';
        break;
    case "login":
        include 'views/pages/login.php';
        break;
```

```

        case "register":
            include 'views/pages/register.php';
            break;
        case "logout":
            include 'models/logout.php';
            break;
        case "admin Panel":
            include 'views/pages/adminpanel.php';
            break;
        case "Documentation":
            header("location: documentation.pdf");
            break;
        case "product":
            include 'views/pages/product.php';
            break;
        case "cart":
            include 'views/pages/cart.php';
            break;
        default:
            include 'views/pages/home.php';
    }

```

```
require_once "views/fixed/footer.php";
```

```
require_once "views/fixed/scripts.php";
```

.env

```

SERVER=localhost
DATABASE=sajtdb
USERNAME=root
PASSWORD=

```

Config.php

```

<?php

define("MYSQL_HOST", env("SERVER"));
define("MYSQL_DBNAME", env("DATABASE"));
define("MYSQL_USERNAME", env("USERNAME"));
define("MYSQL_PASSWORD", env("PASSWORD"));

```

```

function env($marker){
    $baseUrl=$_SERVER['DOCUMENT_ROOT'].'/sneakerShop-master\';
    $arr=file($baseUrl."config/.env");
    $reqValue= "";

    foreach($arr as $row){
        $row=trim($row);

        list($id,$value) = explode("=", $row);

        if($id==$marker){
            $reqValue=$value;
            break;
        }
    }

    return $reqValue;
}
?>

```

## Connection.php

```

<?php
require_once "config.php";

try {

    $connection = new PDO("mysql:host=" . MYSQL_HOST . ";dbname=" .
MYSQL_DBNAME . ";charset=utf8", MYSQL_USERNAME, MYSQL_PASSWORD);

    $connection->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);

    //$connection->setAttribute(PDO::ATTR_DEFAULT_FETCH_MODE,
PDO::FETCH_OBJ);
}
catch (PDOException $e) {
    echo $e->getMessage();
}
}

```

## Head.php

```

<!doctype html>

```

```

<html class="no-js" lang="zxx">

<head>
    <meta charset="utf-8">
    <meta http-equiv="x-ua-compatible" content="ie=edge">
    <title>Footwear shop | eCommerce</title>
    <meta name="description" content="">
    <meta name="viewport" content="width=device-width, initial-scale=1">
    <link rel="shortcut icon" type="image/x-icon"
href="assets/img/favicon.ico">

    <!-- css fajlovi template-a -->
    <link rel="stylesheet" href="./assets/css/mojCSS/moj.css">
    <link
href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min
.css" rel="stylesheet" integrity="sha384-
EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWFspD3yD65VohhpUuCOmLASjC"
crossorigin="anonymous">    <link rel="stylesheet"
href="./assets/css/flaticon.css">
    <link rel="stylesheet" href="./assets/css/slicknav.css">
    <link rel="stylesheet" href="./assets/css/animate.min.css">
    <link rel="stylesheet" href="./assets/css/magnific-popup.css">
    <link rel="stylesheet" href="./assets/css/fontawesome-all.min.css">
    <link rel="stylesheet" href="./assets/css/themify-icons.css">
    <link rel="stylesheet" href="./assets/css/slick.css">
    <link rel="stylesheet" href="./assets/css/nice-select.css">
    <link rel="stylesheet" href="./assets/css/style.css">
    <link rel="stylesheet"
href="https://www.w3schools.com/w3css/4/w3.css">
</head>

<?php
    define("BASE_URL", $_SERVER['DOCUMENT_ROOT'].'/sneakerShop-
master/');
    define("LOG_FILE", BASE_URL."data/log.txt");
    $page="";
    if (isset($_GET['page'])) {
        $page = $_GET['page'];
    }
    logPage($page);

    function logPage($page){
        $visitedPage = $_SERVER['PHP_SELF'];
        $dateTime = date("d. m. Y. h:i:s");
        $ip = $_SERVER['REMOTE_ADDR'];

        if($page != ""){
            $content =
$page."__".$visitedPage."__".$dateTime."__".time()."__".$ip."\n";

```

```

    }
    else{
        $content =
"home__".$visitedPage."__".$dateTime."__".time()."__".$ip."\n";
    }

    $logFile = file_put_contents(LOG_FILE, $content, FILE_APPEND);
}

?>

```

## Header.php

```

<body>
    <!--? Preloader Start -->
    <div id="preloader-active">
        <div class="preloader d-flex align-items-center justify-content-center">
            <div class="preloader-inner position-relative">
                <div class="preloader-circle"></div>
                <div class="preloader-img pere-text">
                    
                </div>
            </div>
        </div>
    </div>
    <!-- Preloader Start -->
    <header>
        <!-- Header Start -->
        <div class="header-area">
            <div class="main-header header-sticky">
                <div class="container-fluid">
                    <div class="menu-wrapper">
                        <!-- Logo -->
                        <div class="logo">
                            <a href="index.php"></a>
                        </div>
                        <!-- Main-menu -->
                        <div class="main-menu d-none d-lg-block">
                            <nav>
                                <ul id="navigation">

                                    <!-- ispis iz js-a-->
                                </ul>
                            </nav>
                        </div>
                    </div>
                </div>
            </div>
        </div>
    </header>

```





```

switch ($page) {

    case "shop":
        echo "Shop";
        break;
    case "Documentation":
        echo "Documentation";
        break;
    case "about":
        echo "About";
        break;
    case "contact":
        echo "Contact";
        break;
    case "login":
        echo "Login";
        break;
    case "register":
        echo "Register now and become our
member";

        break;
    default:
        echo 'nesto';

}
?>
</h2>
</div>
</div>
</div>
</div>
</div>
</div>
<!-- Hero Area End-->

```

## Scripts.php

```

<!-- JS fajlovi template-a, uglavnom animacije -->

<script src="./assets/js/vendor/modernizr-3.5.0.min.js"></script>
<!-- JQuery, Popper, Bootstrap -->
<script src="./assets/js/vendor/jquery-1.12.4.min.js"></script>
<script src="./assets/js/popper.min.js"></script>
<script src="./assets/js/bootstrap.min.js"></script>

```

```

<!-- JQuery Mobile Menu -->
<script src="./assets/js/jquery.slicknav.min.js"></script>

<!-- JQuery Slick , Owl-Carousel Plugins -->
<script src="./assets/js/owl.carousel.min.js"></script>
<script src="./assets/js/slick.min.js"></script>

<!-- One Page, Animated-HeadLin -->
<script src="./assets/js/wow.min.js"></script>
<script src="./assets/js/animated.headline.js"></script>
<script src="./assets/js/jquery.magnific-popup.js"></script>

<!-- Scrollup, nice-select, sticky -->
<script src="./assets/js/jquery.scrollUp.min.js"></script>
<script src="./assets/js/jquery.nice-select.min.js"></script>
<script src="./assets/js/jquery.sticky.js"></script>

<!-- contact js -->
<script src="./assets/js/contact.js"></script>
<script src="./assets/js/jquery.form.js"></script>
<script src="./assets/js/jquery.validate.min.js"></script>
<script src="./assets/js/mail-script.js"></script>
<script src="./assets/js/jquery.ajaxchimp.min.js"></script>

<!-- JQuery Plugins, main JQuery -->
<script src="./assets/js/plugins.js"></script>
<script src="./assets/js/main.js"></script>
<script src="./assets/js/mojJS/main.js"></script>

</body>

</html>

```

Navigation.php

```

<?php
session_start();
include "config/connection.php";

if (isset($_SESSION['user'])) {
    $user = $_SESSION['user'];

    if ($user['role_id'] == 2) {
        $privilege = 2;
    } else {

```

```

        $privilege = 3;
    }
} else {
    $privilege = 1;
}

$query = "SELECT * FROM menus WHERE privileges=1";

if ($privilege == 2) {
    $query .= " OR privileges=2";
} else if ($privilege == 3) {
    $query .= " OR privileges=2 OR privileges=3";
}

$query .= " ORDER BY position";

try {
    $data = $connection->query($query)->fetchAll();
    $rez = $data;
    //var_dump($data);
    echo json_encode($rez);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

About.php

```

<main>
    <?php
    require_once 'views/fixed/heroArea.php';
    ?>
    <!-- About Details Start -->
    <div class="about-details section-padding30">
        <div class="container">
            <div class="row">
                <div class="col-lg-6 col-md-6 col-sm-12 d-flex align-items-center justify-content-center mb-5 mb-md-0 mb-lg-0">
                    
                </div>
                <div class="col-lg-6 col-md-6 col-sm-12 text-center d-flex align-items-center justify-content-center">
                    <p><b>My name is Mateja Rastoder. I'm a 20 year old programmer from Serbia.</b>

```

I'm currently enrolled in the ICT college located in Belgade. This website is a project made for one of my subjects.

```
</b>
</p>
</div>
</div>
</div>
<!-- About Details End -->
</main>
```

Adminpanel.php

```
<?php
if (isset($_SESSION['user'])) {
    $user = $_SESSION['user'];
    if ($user['role_id'] == 1) {
        require_once "config/connection.php";

        echo "<div class='container'>
            <div class='row'>
                <div class='col-12'>
                    <h3 class='mb-5 mt-5 text-center'>Welcome " .
$user['name'] . "</h3>
                </div>
            </div>
        </div>";

        //ovde ubaciti statistiku pristupa stranicama
        $log=file('data/log.txt');
        //var_dump($log);
        $brojPristupa=count($log);
        //var_dump($brojPristupa);
        $procenat = 100 / $brojPristupa;
        $HOME=0;
        $DOC=0;
        $AP=0;
        $SHOP=0;
        $ABOUT=0;
        $CONTACT=0;
        foreach($log as $linija){
            $stranica=explode("__",$linija)[0];
            $vreme=explode("__",$linija)[3];
            if($vreme>(time()-86400)){
                switch ($stranica) {
```



```

        </div>
        <p class="fs-5">Admin panel page</p>
        <div class="w3-border">
            <div class="w3-blue"
style="height:24px;width:<?=$AP?>%"><?=number_format($AP, 2)."%";?></div>
            </div>
            <p class="fs-5">Shop page</p>
            <div class="w3-border">
                <div class="w3-blue"
style="height:24px;width:<?=$SHOP?>%"><?=number_format($SHOP,
2)."%";?></div>
                </div>
                <p class="fs-5">About page</p>
                <div class="w3-border">
                    <div class="w3-blue"
style="height:24px;width:<?=$ABOUT?>%"><?=number_format($ABOUT,
2)."%";?></div>
                    </div>
                    <p class="fs-5">Contact page</p>
                    <div class="w3-border">
                        <div class="w3-blue"
style="height:24px;width:<?=$CONTACT?>%"><?=number_format($CONTACT,
2)."%";?></div>
                        </div>
                    </div>
                    </div>
                    </div>
                    </div>
                    <?php
include 'models/adminmenu.php';

echo '<div class="container">
<div class="row">
<div class="col-12" id="adminIspis"></div>
</div></div> </div>';
    }
}
?>

```

## Cart.php

```
<div class="ads-grid py-sm-5 py-4">
    <div class="container" id="products">

    </div>

    <div class="container" id="buySuccess">
        <div class="row">
            <div class="col-md-12 text-right">
                <p class="alert alert-success">Thanks for your purchase. Items will be delivered in
2-5 business days</p>
            </div>
        </div>
    </div>

    <div class="container mt-2">
        <div class="row">
            <div class="col-md-12 text-right" id="cart-sum">

            </div>
        </div>

        <div class="row">
            <div class="col-md-12 d-flex flex-row align-items-right" id="buy">
                <btn class="btn btn-secondary flex-end ml-auto" id="btnClear">Complete
purchase</btn>
            </div>
        </div>
    </div>
</div>
```

## Contact.php

```
<main>
    <?php
        require_once 'views/fixed/heroArea.php';
```

```

?>
<!-- ===== contact section start ===== -->
<section class="contact-section">
  <div class="container">
    <div class="d-none d-sm-block mb-5 pb-4">

    </div>

    <div class="row">
      <div class="col-12">
        <h2 class="contact-title">Get in Touch</h2>
      </div>
      <div class="col-lg-8">
        <form class="form-contact contact_form"
action="contact_process.php" method="post" id="contactForm"
novalidate="novalidate">
          <div class="row">
            <div class="col-12">
              <div class="form-group">
                <textarea class="form-control w-100"
name="message" id="message" cols="30" rows="9" onfocus="this.placeholder
= ''" onblur="this.placeholder = 'Enter Message'" placeholder=" Enter
Message"></textarea>
              </div>
            </div>
            <div class="col-sm-6">
              <div class="form-group">
                <input class="form-control valid"
name="name" id="name" type="text" onfocus="this.placeholder = ''"
onblur="this.placeholder = 'Enter your name'" placeholder="Enter your
name">
              </div>
            </div>
            <div class="col-sm-6">
              <div class="form-group">
                <input class="form-control valid"
name="email" id="email" type="email" onfocus="this.placeholder = ''"
onblur="this.placeholder = 'Enter email address'" placeholder="Email">
              </div>
            </div>
            <div class="col-12">
              <div class="form-group">
                <input class="form-control"
name="subject" id="subject" type="text" onfocus="this.placeholder = ''"
onblur="this.placeholder = 'Enter Subject'" placeholder="Enter Subject">
              </div>
            </div>
          </div>
        </div>
      </div>
    </div>
  </div>

```



```

        <div class="form-group mt-3">
            <button type="submit" class="button button-
contactForm boxed-btn">Send</button>
        </div>
    </form>
</div>
<div class="col-lg-3 offset-lg-1">
    <div class="media contact-info">
        <span class="contact-info__icon"><i class="ti-
home"></i></span>

        <div class="media-body">
            <h3>Buttonwood, California.</h3>
            <p>Rosemead, CA 91770</p>
        </div>
    </div>
    <div class="media contact-info">
        <span class="contact-info__icon"><i class="ti-
tablet"></i></span>

        <div class="media-body">
            <h3>+1 253 565 2365</h3>
            <p>Mon to Fri 9am to 6pm</p>
        </div>
    </div>
    <div class="media contact-info">
        <span class="contact-info__icon"><i class="ti-
email"></i></span>

        <div class="media-body">
            <h3>support@colorlib.com</h3>
            <p>Send us your query anytime!</p>
        </div>
    </div>
</div>
</div>
</div>
</section>
<!-- ===== contact section end ===== -->
</main>

```

Home.php

```

<main>
    <!--? slider Area Start -->
    <div class="slider-area">
        <div class="slider-active" id="mestoZaSlider">
            <!-- Single Slider -->

```

```

        <div class="single-slider slider-height d-flex align-items-
center slide-bg">
            <div class="container">
                <div class="row justify-content-between align-items-
center">
                    <div class="col-xl-8 col-lg-8 col-md-8 col-sm-8">
                        <div class="hero__caption">
                            <h1 data-animation="fadeInLeft" data-
delay=".4s" data-duration="200ms" id="naslovSlider">
                                <!--Ispis-->
                            </h1>
                            <p data-animation="fadeInLeft" data-
delay=".7s" data-duration="200ms" id="opisSlider">
                                <!--Ispis-->
                            </p>

                            <div class="hero__btn" data-
animation="fadeInLeft" data-delay=".8s" data-duration="200ms">
                                <a href="index.php?page=shop"
class="btn hero-btn">Shop Now</a>
                            </div>
                        </div>
                    </div>
                    <div class="col-xl-3 col-lg-3 col-md-4 col-sm-4
d-none d-sm-block">
                        <div class="hero__img">
                            <!-- ISPIS iz JS-a
                             -->
                        </div>
                    </div>
                </div>
            </div>
        </div>
        <!-- slider Area End-->
        <!-- ? New Product Start -->
        <section class="new-product-area section-padding30">
            <div class="container">
                <!-- Section tittle -->
                <div class="row">
                    <div class="col-xl-12">
                        <div class="section-tittle mb-70">
                            <h2>New Arrivals</h2>
                        </div>
                    </div>
                </div>
            </div>
            <div class="row" id="newArrivals">

```

```

        <!-- Dinamicki ispis -->
    </div>
</div>
</section>
<!-- New Product End -->
<!--? Gallery Area Start -->
<div class="gallery-area">
    <div class="container-fluid p-0 fix">
        <div class="row" id="placeForGalleryArea">
            <!-- Dinamicki ispis -->
        </div>
    </div>
</div>
<!-- Gallery Area End -->
<!--? Popular Items Start -->
<div class="popular-items section-padding30">
    <div class="container">
        <!-- Section tittle -->
        <div class="row justify-content-center">
            <div class="col-xl-7 col-lg-8 col-md-10">
                <div class="section-tittle mb-70 text-center">
                    <h2>Popular Items</h2>
                    <p>See the best selling products on our page</p>
                </div>
            </div>
        </div>
        <div class="row" id="producstArea">
            <!-- Dinamicki ispis -->
        </div>
        <!-- Button -->
        <div class="row justify-content-center">
            <div class="room-btn pt-70">
                <a href="index.php?page=shop" class="btn view-
btn1">View More Products</a>
            </div>
        </div>
    </div>
</div>
<!-- Popular Items End -->
<!--? Watch Choice Start-->
<div class="watch-area section-padding30">
    <div class="container" id="watchOfChoiceGallery">
        <!-- Dinamicki ispis -->
    </div>
</div>
<!-- Watch Choice End-->
<!--? Shop Method Start-->
<div class="shop-method-area">
    <div class="container">

```

```

        <div class="method-wrapper">
            <div class="row d-flex justify-content-between"
id="methodPaymentOffers">
                <!-- Dinamicki ispis -->
            </div>
        </div>
    </div>
</div>
<!-- Shop Method End-->
</main>

```

Login.php

```

<main>
    <?php
        require_once 'views/fixed/heroArea.php';

        if (isset($_SESSION['uspesnaRegistracija'])) {

            echo '<div class="alert alert-success text-center mt-4"
role="alert">
                Registration successful! <br> Check your email, we have sent you
activation link!
            </div>';

            unset($_SESSION['uspesnaRegistracija']);
        }

        if (isset($_SESSION['uspesnaVerifikacija'])) {
            echo '<div class="alert alert-success text-center mt-4"
role="alert">
                Activation successful! <br> You can login now!
            </div>';

            unset($_SESSION['uspesnaVerifikacija']);
        }

        if (isset($_SESSION['errors'])) {
            $errors = $_SESSION['errors'];
            echo '<div class="alert alert-warning text-center mt-4"
role="alert">';
            echo $errors;
            echo '</div>';

            unset($_SESSION['errors']);
        } elseif (isset($_SESSION['userErrors'])) {
            $errors = $_SESSION['userErrors'];

```

```

        echo '<div class="alert alert-warning text-center mt-4"
role="alert">';
        foreach ($errors as $e) {
            echo $e;
        }
        echo '</div>';

        unset($_SESSION['userErrors']);
    }
?>
<section class="login_part section_padding ">
    <div class="container">
        <div class="row align-items-center">
            <div class="col-lg-6 col-md-6">
                <div class="login_part_text text-center">
                    <div class="login_part_text_iner">
                        <h2>New to our Shop?</h2>
                        <p>There are advances being made in science
and technology
                                everyday, and a good example of this is
the</p>
                                <a href="index.php?page=register"
class="btn_3">Create an Account</a>
                            </div>
                        </div>
                    </div>
                </div>
            <div class="col-lg-6 col-md-6">
                <div class="login_part_form">
                    <div class="login_part_form_iner">
                        <h3>Welcome Back ! <br>
                            Please Sign in now</h3>
                        <form class="row contact_form"
onsubmit="return proveraLogovanje();" action="models/loginCheck.php"
method="post">
                            <div class="col-md-12 form-group p_star">
                                <input type="email" class="form-
control" id="email" name="email" value="" placeholder="Email">
                            </div>
                            <span class="col-md-12 form-group
invisible" id="emailError">
                                Enter valid email.
                            </span>
                            <div class="col-md-12 form-group p_star">
                                <input type="password" class="form-
control" id="password" name="password" value="" placeholder="Password">
                            </div>
                            <span class="col-md-12 form-group
invisible" id="passwordError">
                                Incorrect password.

```





```

                Enter valid lastname. Must contain
first letter uppercase and at least 3 characters
            </span>
            <!-- email -->
            <div class="col-md-12 form-group p_star">
                <input type="email" class="form-
control" id="email" name="email" value="" placeholder="Email">
            </div>
            <span class="col-md-12 form-group
invisible" id="emailError">
                Enter valid email.
            </span>
            <!-- password -->
            <div class="col-md-12 form-group p_star">
                <input type="password" class="form-
control" id="password" name="password" value="" placeholder="Password">
            </div>
            <span class="col-md-12 form-group
invisible" id="passwordError">
                Enter valid password. Must contain
first letter uppercase and at least 6 characters
            </span>
            <div class="col-md-12 form-group p_star">
                <input type="password" class="form-
control" id="testpassword" name="testpassword" value="" placeholder="Re-
enter password">
            </div>
            <span class="col-md-12 form-group
invisible" id="testPasswordError">
                Re-type correctly password!
            </span>
            <div class="col-md-12 form-group">
                <!-- <div class="creat_account d-flex
align-items-center">
                    <input type="checkbox" id="f-
option" name="selector">
                    <label for="f-option">Remember
me</label>
                </div> -->
                <button type="submit" value="submit"
name="btnRegister" id="btnRegister" class="btn_3">
                    Register
                </button>
                <!-- <a class="lost_pass"
href="#">forget password?</a> -->
            </div>
        </form>
    </div>
</div>

```



```

        </div>
    </div>
</div>
</section>
</main>

```

Shop.php

```

<main>
    <?php
        require_once 'views/fixed/heroArea.php';
        if (isset($_SESSION['user'])):

            if (!isset($_GET['brojProizvoda'])) {
                $results_per_page = 3;
            } else {
                $results_per_page = $_GET['brojProizvoda'];
            }

            if (!isset($_GET['sort'])) {
                $sort = "";
            } else {
                $sort = $_GET['sort'];
            }

            if (!isset($_GET['stranicaProizvoda'])) {
                $page = 1;
            } else {
                $page = $_GET['stranicaProizvoda'];
            }

        ?>
        <!-- Latest Products Start -->
        <section class="popular-items latest-padding">
            <div class="container">
                <div class="row product-btn justify-content-between mb-40">
                    <div class="properties__button">
                        <!--Nav Button -->
                        <nav>
                            <div class="nav nav-tabs" id="nav-tab"
role="tablist">
                                <a class="nav-item nav-link"
id="btnPrikazSve"
href="index.php?page=shop&brojProizvoda=3&stranicaProizvoda=<?=$page?>&sort=<?=$sort?>" name="noviProizvodi">3 per page</a>
                                <a class="nav-item nav-link"
id="btnPrikazSve"
href="index.php?page=shop&brojProizvoda=6&stranicaProizvoda=<?=$page?>&sort=<?=$sort?>" name="noviProizvodi">6 per page</a>

```

```

        <a class="nav-item nav-link"
id="btnPrikazSve"
href="index.php?page=shop&brojProizvoda=<?=$results_per_page?>&stranicaPr
oizvoda=<?=$page?>&sort=nameAsc" name="noviProizvodi">Name A-Z</a>
        <a class="nav-item nav-link"
id="btnPrikaziNove"
href="index.php?page=shop&brojProizvoda=<?=$results_per_page?>&stranicaPr
oizvoda=<?=$page?>&sort=nameDesc" name="noviProizvodi">Name Z-A</a>
        <a class="nav-item nav-link"
id="btnPriceHighLow"
href="index.php?page=shop&brojProizvoda=<?=$results_per_page?>&stranicaPr
oizvoda=<?=$page?>&sort=priceDesc" name="noviProizvodi"
name="cenaViseNize"> Price high to low</a>
        <a class="nav-item nav-link"
id="btnPriceLowHigh"
href="index.php?page=shop&brojProizvoda=<?=$results_per_page?>&stranicaPr
oizvoda=<?=$page?>&sort=priceAsc" name="cenaNizeVise"> Price low to
high</a>
    </div>
</nav>
<!--End Nav Button -->
<div id="cartPopup">
    <p class="alert alert-success text-center">Product
added to cart</p>
</div>
</div>
<!-- Grid and List view -->
<div class="grid-list-view">
</div>
<!-- Select items -->
<!-- <div class="select-this">
    <form action="#">
        <div class="select-itms">
            <select name="select" id="select1">
                <option value=""><a
href="index.php?page=shop&brojProizvoda=3">3 per page</a></option>
                <option value=""><a
href="index.php?page=shop&brojProizvoda=6">6 per page</a></option>
            </select>
        </div>
    </form>
</div> -->
</div>
<!-- Nav Card -->
<div class="tab-content" id="nav-tabContent">
    <!-- card one -->
    <div class="tab-pane fade show active" id="nav-home"
role="tabpanel" aria-labelledby="nav-home-tab">
        <div class="row" id="shopProducts">

```

```

                <!-- Dinamicki ispis -->
                <?php include "models/getProducts.php" ?>
            </div>
        </div>
    </div>
    <!-- End Nav Card -->
</div>
</section>
</main>

<?php
endif;
if (!isset($_SESSION['user'])){
    header('Location: index.php?page=login');
}
?>

```

## Models

Brisanje u admin panelu

Brands

```

<?php
require_once '../config/connection.php';

if (isset($_POST['delete'])) {
    $brandID = $_POST['poslatiID'];

    $code = 0;
    $successArray = [];

    $sql = 'DELETE FROM brands WHERE id=:id';
    $prepare = $connection->prepare($sql);
    $prepare->bindParam(':id', $brandID);

    try {
        $prepare->execute();
        $successArray['msg'] = "You deleted brand successfully!";
        $successArray['brandDeleted'] = true;
        $code = 200;
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
} else {
    $successArray["error"] = "Error! Try Again!";
    $code = 404;
}

```

```
echo json_encode($successArray);
http_response_code($code);
```

gender

```
<?php
require_once '../config/connection.php';

if (isset($_POST['delete'])) {
    $genderID = $_POST['poslatiID'];

    $code = 0;
    $successArray = [];

    $sql = 'DELETE FROM gender WHERE id=:id';
    $prepare = $connection->prepare($sql);
    $prepare->bindParam(':id', $genderID);

    try {
        $prepare->execute();
        $successArray['msg'] = "You deleted gender successfully!";
        $successArray['genderDeleted'] = true;
        $code = 200;
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
} else {
    $successArray["error"] = "Error! Try Again!";
    $code = 404;
}

echo json_encode($successArray);
http_response_code($code);
```

menu

```
<?php
require_once '../config/connection.php';

if (isset($_POST['delete'])) {
    $menuID = $_POST['poslatiID'];
```

```

$code = 0;
$successArray = [];

$sql = 'DELETE FROM menus WHERE id=:id';
$prepare = $connection->prepare($sql);
$prepare->bindParam(':id', $menuID);

try {
    $prepare->execute();
    $successArray['msg'] = "You deleted menu item successfully!";
    $successArray['menuDeleted'] = true;
    $code = 200;
} catch (PDOException $e) {
    echo $e->getMessage();
}
} else {
    $successArray["error"] = "Error! Try Again!";
    $code = 404;
}

echo json_encode($successArray);
http_response_code($code);

```

product

```

<?php
require_once '../config/connection.php';

if (isset($_POST['delete'])) {
    $productID = $_POST['poslatiID'];

    $msg = "";
    $code = 0;
    $successArray = [];

    $sql = 'DELETE FROM products WHERE id =:id';
    $prepare = $connection->prepare($sql);
    $prepare->bindParam(':id', $productID);

    try {
        $prepare->execute();
        $successArray['msg'] = "You deleted product successfully!";
        $successArray['productDeleted'] = true;
        $code = 200;
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
}

```

```

} else {
    $successArray["error"] = "Error! Try Again!";
    $code = 404;
}

echo json_encode($msg);
http_response_code($code);

```

user

```

<?php
require_once '../config/connection.php';

if (isset($_POST['delete'])) {
    $userID = $_POST['poslatiID'];

    $code = 0;
    $successArray = [];

    $sql = 'DELETE FROM users WHERE id=:id';
    $prepare = $connection->prepare($sql);
    $prepare->bindParam(':id', $userID);

    try {
        $prepare->execute();
        $successArray['msg'] = "You deleted user successfully!";
        $successArray['userDeleted'] = true;
        $code = 200;
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
} else {
    $successArray["error"] = "Error! Try Again!";
    $code = 404;
}

echo json_encode($successArray);
http_response_code($code);

```

getAll funkcije

brendovi

```

<?php

```

```

require_once '../config/connection.php';

try {
    $sql = 'SELECT * FROM brands';
    $rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC);
    if (count($rez) == 0) {
        die();
    } else {
        echo json_encode($rez);
        http_response_code(200);
    }
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

## Genders

```

<?php
require_once '../config/connection.php';

try {
    $sql = 'SELECT * FROM gender';
    $rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC);
    if (count($rez) == 0) {
        die();
    } else {
        echo json_encode($rez);
        http_response_code(200);
    }
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

## Menus

```

<?php
require_once '../config/connection.php';

try {
    $sql = 'SELECT id, name, position, (SELECT name FROM privileges WHERE
id=privileges) as privilege FROM menus';

```

```

    $rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC));
    if (count($rez) == 0) {
        die();
    } else {
        echo json_encode($rez);
        http_response_code(200);
    }
} catch (PDOException $e) {
    echo $e->getMessage();
}
}

```

## Products

```

<?php
require_once '../config/connection.php';
require_once __DIR__ . '/../config/connection.php';

try {
    $sql = 'SELECT p.id, p.name, p.price, p.old_price, p.img,
p.description, g.name as gender, b.name as brand, p.dateofadd FROM
products p INNER JOIN gender g ON p.gender_id=g.id INNER JOIN brands b ON
p.brand_id=b.id';
    $rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC));
    if (count($rez) == 0) {
        die();
    } else {
        echo json_encode($rez);
        http_response_code(200);
    }
} catch (PDOException $e) {
    echo $e->getMessage();
}
}

```

## Users

```

<?php
require_once '../config/connection.php';

try {
    $sql = 'SELECT id, name, lastname, email, (SELECT name FROM roles
WHERE id=role_id) as role, password, code,
CASE isActive
    WHEN "1" THEN "Yes"

```



```

        WHEN "0" THEN "No"
    END as "Is Active",
    dateofcreation FROM users';
$rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC));
if (count($rez) == 0) {
    die();
} else {
    echo json_encode($rez);
    http_response_code(200);
}
} catch (PDOException $e) {
    echo $e->getMessage();
}

// u.id, u.name, u.lastname, u.email, u.password, r.name, u.code,
u.dateofcreation, u.isActive

```

Brands and genders

```

<?php

require_once '../config/connection.php';

try {
    $sqlGenders = $connection->query('SELECT id,name FROM gender')->fetchAll(PDO::FETCH_ASSOC));
    $sqlBrands = $connection->query('SELECT id,name FROM brands')->fetchAll(PDO::FETCH_ASSOC));

    $data = array();
    $data['genders'] = $sqlGenders;
    $data['brands'] = $sqlBrands;

    echo json_encode($data);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

Gallery images

```

<?php
require_once '../config/connection.php';

try {

```

```

        $query = $connection->query("SELECT * FROM galleryimages g INNER JOIN
purposes p ON g.purpose=p.id ")->fetchAll();
        echo json_encode($query);
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
}

```

New arrivals

```

<?php
require_once '../config/connection.php';

try {
    $query = $connection->query('SELECT * FROM products ORDER BY
dateofadd DESC LIMIT 3')->fetchAll();
    echo json_encode($query);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

getPaymentOffers

```

<?php
require_once '../config/connection.php';

try {
    $query = $connection->query("SELECT * FROM paymentoffers")->fetchAll();
    echo json_encode($query);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

getPrivileges

```

<?php
require_once '../config/connection.php';

try {
    $sqlPrivileges = $connection->query('SELECT id,name FROM
privileges')->fetchAll(PDO::FETCH_ASSOC));
}

```

```

        $data = $sqlPrivileges;
        echo json_encode($data);
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
}

```

getProduct

```

<?php
require_once '../config/connection.php';
require_once __DIR__ . '/../config/connection.php';

try {
    $sql = 'SELECT p.id, p.name, p.price, p.old_price, p.img,
p.description, g.name as gender, b.name as brand, p.dateofadd FROM
products p INNER JOIN gender g ON p.gender_id=g.id INNER JOIN brands b ON
p.brand_id=b.id WHERE id =:id';
    $prepare = $connection->prepare($sql);
    $prepare->bindParam(':id', $product);

    try {
        $prepare->execute();
        $successArray['msg'] = "You deleted product successfully!";
        $successArray['productDeleted'] = true;
        $code = 200;
    } catch (PDOException $e) {
        echo $e->getMessage();
    }

    // $rez = $connection->query($sql)->fetchAll(PDO::FETCH_ASSOC));
    // if (count($rez) == 0) {
    //     die();
    // } else {
    //     echo json_encode($rez);
    //     http_response_code(200);
    // }
}

```

getProducts

```

<?php
//require_once '../config/connection.php';
require_once __DIR__ . '/../config/connection.php';

```

```
if (!isset($_GET['brojProizvoda'])) {
    $results_per_page = 3;
}
else {
    $results_per_page = $_GET['brojProizvoda'];
}

if (!isset($_GET['sort'])) {
    $sort = "";
}
else {
    $sort = $_GET['sort'];
}

$query = $connection->query("SELECT * FROM products")->fetchAll();

$number_of_results = count($query);
$number_of_pages = ceil($number_of_results / $results_per_page);

if (!isset($_GET['stranicaProizvoda'])) {
    $page = 1;
} else {
    $page = $_GET['stranicaProizvoda'];
}

$this_page_first_result = ($page - 1) * $results_per_page;

if($sort==""){
    $sql = 'SELECT * FROM products LIMIT ' . $this_page_first_result .
    ',' . $results_per_page;
}
else if($sort=="priceDesc"){
    $sql = 'SELECT * FROM products ORDER BY price DESC LIMIT ' .
    $this_page_first_result . ',' . $results_per_page;
}
else if($sort=="priceAsc"){
    $sql = 'SELECT * FROM products ORDER BY price ASC LIMIT ' .
    $this_page_first_result . ',' . $results_per_page;
}
else if($sort=="nameDesc"){
    $sql = 'SELECT * FROM products p ORDER BY p.name DESC LIMIT ' .
    $this_page_first_result . ',' . $results_per_page;
}
else if($sort=="nameAsc"){
```

```

        $sql = 'SELECT * FROM products p ORDER BY p.name ASC LIMIT ' .
$this_page_first_result . ',' . $results_per_page;
    }

    //$sql = 'SELECT * FROM products LIMIT ' . $this_page_first_result . ','
    . $results_per_page;
    $result = $connection->query($sql)->fetchAll();

    foreach ($result as $row) {
        echo '<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
        <div class="single-popular-items mb-50 text-center">
            <div class="popular-img">
                
                <div class="img-cap">
                    <span class="btnAddToCart" data-id=' . $row['id'] . '>Add
to cart</span>
                </div>
            </div>
            <div class="popular-caption">
                <h3><a href="product_details.html">' . $row['name'] .
'</a></h3>
                <span>$ ' . $row['price'] . '</span>
            </div>
        </div>
    </div>';
    }

    echo '<nav aria-label="Page navigation example">
        <ul class="pagination justify-content-center">';
    // <li class="page-item disabled">
    // <a class="page-link" href="#" tabindex="-1">Previous</a>
    // </li>';
    for ($page = 1; $page <= $number_of_pages; $page++) {
        echo '<li class="page-item"><a class="page-link"
href="index.php?sort=' . $sort . '&page=shop&brojProizvoda=' .
$results_per_page . '&stranicaProizvoda=' . $page . '">' . $page . '</a>
</li>';
    }
    echo '
        </ul>
    </nav>';

```

GetRoles

```
<?php
```

```

require_once '../config/connection.php';

try {
    $sqlRoles = $connection->query('SELECT id,name FROM roles')->fetchAll(PDO::FETCH_ASSOC);
    $data = $sqlRoles;
    echo json_encode($data);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

getSliderImages

```

<?php
require_once "../config/connection.php";

try {
    $query = $connection->query("SELECT * FROM slider_images")->fetchAll();
    echo json_encode($query);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

getSocialIcons

```

<?php
require_once "../config/connection.php";

try {
    $query = $connection->query("SELECT * FROM socials")->fetchAll();
    echo json_encode($query);
} catch (PDOException $e) {
    echo $e->getMessage();
}

```

getTopProducts

```

<?php

require_once '../config/connection.php';
try {

```

```

        $query = $connection->query("SELECT p.id, p.name, p.price, p.img,
pr.id_product, COUNT(pr.id_product) AS MostSold FROM products_orders pr
INNER JOIN products p ON pr.id_product=p.id INNER JOIN orders o ON
pr.id_order=o.id GROUP BY pr.id_product ORDER BY MostSold DESC LIMIT 3")-
>fetchAll();
        echo json_encode($query);
    } catch (PDOException $e) {
        echo $e->getMessage();
    }
}

```

Helper.php

```

<?php
Ob_start();
function redirect($url, $statusCode)
{
    header('Location: ' . $url, true, $statusCode);
    die();
}

```

Insert funkcije

Brand

```

<?php

if (isset($_POST['insert'])) {
    require_once '../config/connection.php';

    $msg = [];
    $code = 200;

    $brandName = trim($_POST['brandName']);

    $errors = 0;

    $regexBrandName = "/^[A-z]+[\s[A-z0-9]+]?$/";
    if (!preg_match($regexBrandName, $brandName)) {
        array_push($msg, "Brand name must contain letters! <br>");
        $errors++;
    }

    if ($errors == 0) {
        $sql = "INSERT INTO brands VALUES (null, :name)";
        $priprema = $connection->prepare($sql);
        $priprema->bindParam(":name", $brandName);
    }
}

```

```

        try {
            $priprema->execute();
            array_push($msg, "Brand successfully added!");
            echo json_encode($msg);
            http_response_code(201);
        } catch (PDOException $e) {
            //echo $e->getMessage();
            array_push($msg, "Server error!");
            echo json_encode($msg);
            http_response_code(500);
        }
    } else {
        echo json_encode($msg);
        http_response_code(422);
    }
}

```

Gender

```

<?php

if (isset($_POST['insert'])) {
    require_once '../config/connection.php';

    $msg = [];
    $code = 200;

    $genderName = trim($_POST['genderName']);

    $errors = 0;

    $regexGenderName = "/^[A-z]+[\s[A-z0-9]]?$/";
    if (!preg_match($regexGenderName, $genderName)) {
        array_push($msg, "Gender name must contain letters! <br>");
        $errors++;
    }

    if ($errors == 0) {
        $sql = "INSERT INTO gender VALUES (null, :name)";
        $priprema = $connection->prepare($sql);
        $priprema->bindParam(":name", $genderName);
        try {
            $priprema->execute();
            array_push($msg, "Gender successfully added!");
            echo json_encode($msg);
            http_response_code(201);
        } catch (PDOException $e) {

```



```

        //echo $e->getMessage();
        array_push($msg, "Server error!");
        echo json_encode($msg);
        http_response_code(500);
    }
} else {
    echo json_encode($msg);
    http_response_code(422);
}
}

```

## NewMenuItem

```

<?php

if (isset($_POST['insert'])) {
    require_once '../config/connection.php';

    $msg = [];
    $code = 200;

    $menuItemName = trim($_POST['menuItemName']);
    $menuItemPosition = trim($_POST['menuItemPosition']);
    $privilege = $_POST['selectPrivilege'];

    $errors = 0;

    $regexItemName = "/^[A-z]+[\s[A-z0-9]]?$/";
    if (!preg_match($regexItemName, $menuItemName)) {
        array_push($msg, "Menu item name must contain letters! <br>");
        $errors++;
    }
    if ($menuItemPosition == 0) {
        array_push($msg, "Position must be greater than 0. <br>");
        $errors++;
    }
    if ($privilege == 0) {
        array_push($msg, "You have to choose privilege. <br>");
        $errors++;
    }

    if ($errors == 0) {
        $sql = "INSERT INTO menus VALUES (null, :name, :position,
:privileges)";
        $priprema = $connection->prepare($sql);
        $priprema->bindParam(":name", $menuItemName);
        $priprema->bindParam(":position", $menuItemPosition);
    }
}

```

```

        $priprema->bindParam(":privileges", $privilege);
    try {
        $priprema->execute();
        array_push($msg, "Menu item successfully added!");
        echo json_encode($msg);
        http_response_code(201);
    } catch (PDOException $e) {
        //echo $e->getMessage();
        array_push($msg, "Server error!");
        echo json_encode($msg);
        http_response_code(500);
    }
} else {
    echo json_encode($msg);
    http_response_code(422);
}
}
}

```

## Product

```

<?php
if (isset($_POST['insert'])) {
    require_once '../config/connection.php';

    $msg = [];
    $code = 200;
    @$img = $_FILES['imgFile'];

    @$tmpName = $img['tmp_name'];
    @$size = $img['size'];
    @$tip = $img['type'];
    @$name = $img['name'];
    @$putanja = "../assets/img/gallery/$name";
    @$putanjaManja = "../assets/img/thumbnail/$name";

    $productName = trim($_POST['productName']);
    $price = trim($_POST['price']);
    $oldPrice = trim($_POST['oldPrice']);
    $description = trim($_POST['description']);
    $gender = $_POST['selectGender'];
    $brand = $_POST['selectBrand'];

    $errors = 0;

    $regexProductName = "/^[A-Z][a-z0-9]+[\s[A-z0-9]]?$/";
    $regexPrice = "/^[0-9]+$/";
    $regexOldPrice = "/^[0-9]+$/";
}

```

```

$regexDescription = "/^[A-Z][a-z0-9]+[\s[A-z0-9]+]?$/";

if (!preg_match($regexProductName, $productName)) {
    array_push($msg, "Name must begin with a capital letter and have
2 words! <br>");
    $errors++;
}
if (!preg_match($regexPrice, $price)) {
    array_push($msg, "Price must include only numbers! <br>");
    $errors++;
}
if (!preg_match($regexOldPrice, $oldPrice)) {
    array_push($msg, "Old price must include only numbers! <br>");
    $errors++;
}
if (!preg_match($regexDescription, $description)) {
    array_push($msg, "Description must begin with capital letter and
have at least 2 words. <br>");
    $errors++;
}
if ($gender == 0) {
    array_push($msg, "You have to choose gender <br>");
    $errors++;
}
if ($brand == 0) {
    array_push($msg, "You have to choose brand <br>");
    $errors++;
}
if (@!$_FILES['imgFile']) {
    array_push($msg, "You have to choose image <br>");
    $errors++;
}

if ($errors == 0) {
    $resultat = move_uploaded_file($tmpName, $putanja);
    //move_uploaded_file($tmpName, $putanjaManja);
    // File and new size
    $filename = $putanja;
    $percent = 0.5;

    // Content type
    header('Content-Type: image/jpeg');

    // Get new sizes
    list($width, $height) = getimagesize($filename);
    $newwidth = $width * $percent;
    $newheight = $height * $percent;

```

```

        // Load
        $thumb = imagecreatetruecolor($newwidth, $newheight);
        $source = imagecreatefromjpeg($filename);

        // Resize
        imagecopyresized($thumb, $source, 0, 0, 0, 0, $newwidth,
        $newheight, $width, $height);

        move_uploaded_file($thumb, $putanjaManja);

        // if (!$rezultat) {
        //     array_push($msg, "Error with image upload!");
        //     echo json_encode($msg);
        //     http_response_code(500);
        // } else {
        //     $sql = "INSERT INTO products VALUES (null, :name, :price,
:old_price, :img, :description, :gender_id, :brand_id, null)";
        //     $priprema = $connection->prepare($sql);
        //     $priprema->bindParam("name", $productName);
        //     $priprema->bindParam("price", $price);
        //     $priprema->bindParam("old_price", $oldPrice);
        //     $priprema->bindParam("img", $name);
        //     $priprema->bindParam("description", $description);
        //     $priprema->bindParam("gender_id", $gender);
        //     $priprema->bindParam("brand_id", $brand);
        //     try {
        //         $priprema->execute();
        //         array_push($msg, "Product successfully added!");
        //         echo json_encode($msg);
        //         http_response_code(201);
        //     } catch (PDOException $e) {
        //         echo $e->getMessage();
        //         // array_push($msg, "Server error!");
        //         // echo json_encode($msg);
        //         http_response_code(500);
        //     }
        // }
    } else {
        echo json_encode($msg);
        http_response_code(422);
    }
}

```

newUser

<?php

```
if (isset($_POST['insert'])) {
    require_once '../config/connection.php';

    $msg = [];
    $code = 200;

    $name = trim($_POST['firstName']);
    $lastname = trim($_POST['lastname']);
    $email = trim($_POST['email']);
    $password = trim($_POST['password']);
    $testPassword = $_POST['testPassword'];
    $roleID = $_POST['selectRole'];

    $errors = 0;

    $regexName = "/^[A-Z][a-z]{2,19}$/";
    $regexLastName = "/^[A-Z][a-z]{2,39}$/";
    $regexPassword = "/^[A-z0-9_-]{6,12}$/";

    if (!preg_match($regexName, $name)) {
        array_push($msg, "Name must have start with uppercase letter and
have at least 2 characters!");
        $errors++;
    }

    if (!preg_match($regexLastName, $lastname)) {
        array_push($msg, "Lastname must have start with uppercase letter
and have at least 2 characters!");
        $errors++;
    }

    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        array_push($msg, "Email is not valid!");
        $errors++;
    }

    if (!preg_match($regexPassword, $password)) {
        array_push($msg, "Enter valid password!");
        $errors++;
    }

    if (!preg_match($regexPassword, $testPassword)) {
        array_push($msg, "Re-type password correctly!");
        $errors++;
    }

    if ($testPassword != $password) {
        array_push($msg, "Re-type password correctly!");
        $errors++;
    }
}
```

```

    }
    if ($roleID == 0) {
        array_push($msg, "You have to choose a role <br>");
        $errors++;
    }

    if ($errors == 0) {
        $sql = "INSERT INTO users VALUES (null, :name, :lastname, :email,
:password, :role_id, :code, :dateofcreation, :isActive)";
        $priprema = $connection->prepare($sql);

        $priprema->bindParam(":name", $name);
        $priprema->bindParam(":lastname", $lastname);
        $priprema->bindParam(":email", $email);

        $password = md5($password);

        $priprema->bindParam(":password", $password);

        $priprema->bindParam(":role_id", $roleID);

        $code = sha1(md5(md5(time() . md5($email) . rand(1, 10000000))));
        $priprema->bindParam(":code", $code);

        $dateOfCreation = date("Y-m-d H:i:s");
        $priprema->bindParam(":dateofcreation", $dateOfCreation);

        $isActive = 0;
        $priprema->bindParam(":isActive", $isActive);
        try {
            $priprema->execute();
            //Pri dodavanju usera u lokalnu slanje maila nije moguce.
            //mail($email, "Registration Time Zone", "Click the link to
verify your registration:


```

```
}  
}
```

Login check

```
<?php  
session_start();  
  
if (isset($_POST['btnLogin'])) {  
  
    $email = $_POST['email'];  
    $password = $_POST['password'];  
  
    $regexPassword = "/^{4,50}$"/;  
  
    $errors = array();  
  
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {  
        array_push($errors, "Email is not good!");  
    }  
  
    if (!preg_match($regexPassword, $password)) {  
        array_push($errors, "Enter valid password");  
    }  
  
    if (count($errors) == 0) {  
  
        require "../config/connection.php";  
  
        $activated = 1;  
  
        $provera = $connection->query("SELECT * FROM users WHERE email =  
'$email' AND isActive = '$activated'");  
  
        if ($provera->rowCount() == 1) {  
  
            $password = md5($password);  
            $provera1 = $connection->query("SELECT * FROM users WHERE  
email = '$email' AND password = '$password'");  
  
            if ($provera1->rowCount() == 1) {  
                $user = $provera1->fetch();  
                $_SESSION['user'] = $user;  
                include_once 'helper.php';  
                redirect("../index.php", 303);  
            }  
        }  
    }  
}
```

```

        } else {
            $_SESSION['errors'] = "Password is incorrect";
            echo "password nije dobar!";
            include_once 'helper.php';
            redirect("../index.php?page=login", 303);
        }
    } else {
        $_SESSION['errors'] = "No such user with entered email or
account is not activated!";
        echo "nema maila ili nije aktivan";
        include_once 'helper.php';
        redirect("../index.php?page=login", 303);
    }
} else {
    $_SESSION['userErrors'] = $errors;
    echo "user errori";
    include_once 'helper.php';
    redirect("../index.php?page=login", 303);
}
}
}

```

Logout

```

<?php

session_destroy();
include_once 'helper.php';

redirect('index.php', 303);

```

registration insert

```

<?php
session_start();

if (isset($_POST['btnRegister'])) {
    $name = $_POST['name'];
    $lastname = $_POST['lastname'];
    $email = $_POST['email'];
    $password = $_POST['password'];
    $testPassword = $_POST['testpassword'];

    $regexName = "/^[A-Z][a-z]{2,19}$/";
    $regexLastName = "/^[A-Z][a-z]{2,39}$/";
    $regexPassword = "/^[A-z0-9_-]{6,12}$/";
}

```



```

$errors = array();

if (!preg_match($regexName, $name)) {
    array_push($errors, "Name must have start with uppercase letter
and has at least 2 characters!");
}

if (!preg_match($regexLastName, $lastname)) {
    array_push($errors, "Lastname must have start with uppercase
letter and has at least 2 characters!");
}

if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
    array_push($errors, "Email is not good!");
}

if (!preg_match($regexPassword, $password)) {
    array_push($errors, "Enter valid password!");
}

if (!preg_match($regexPassword, $testPassword)) {
    array_push($errors, "Re-type password correctly!");
}

if ($testPassword != $password) {
    array_push($errors, "Re-type password correctly!");
}

require "../config/connection.php";

$upitProveraEmailaUsera = "SELECT email FROM users where email =
:email";

$pripremaProvere = $connection->prepare($upitProveraEmailaUsera);
$pripremaProvere->bindParam(":email", $email);

$pripremaProvere->execute();
if ($pripremaProvere->rowCount() == 1) {
    array_push($errors, "There is already a user with entered
email!");
}

if (count($errors) == 0) {
    $upit = "INSERT INTO users VALUES(NULL, :name, :lastname, :email,
:password, :role_id, :code, :dateofcreation, :isActive)";

    $priprema = $connection->prepare($upit);

```

```

//bindParam

$priprema->bindParam(":name", $name);
$priprema->bindParam(":lastname", $lastname);
$priprema->bindParam(":email", $email);

$password = md5($password);

$priprema->bindParam(":password", $password);

define("KORISNIK_ULOGA", 2);
$roleID = KORISNIK_ULOGA;
$priprema->bindParam(":role_id", $roleID);

$code = sha1(md5(md5(time() . md5($email) . rand(1, 10000000))));
$priprema->bindParam(":code", $code);

$dateOfCreation = date("Y-m-d H:i:s");
$priprema->bindParam(":dateofcreation", $dateOfCreation);

$isActive = 0;
$priprema->bindParam(":isActive", $isActive);

try {
    $isUneto = $priprema->execute();

    $_SESSION["uspesnaRegistracija"] = "Uspela registracija";

    mail($email, "Registration Time Zone", "Click the link to
verify your registration:
http://localhost:8080/php1/sajt/models/verificationOfRegistration.php?code=" . $code);

    include_once 'helper.php';
    redirect("../index.php?page=login", 303);
} catch (PDOException $e) {
    $_SESSION["errorsDB"] = ["Greska sa bazom!"];

    include_once 'helper.php';
    redirect("../index.php?page=register", 303);
}
} else {
    $_SESSION["userErrors"] = $errors;

    include_once 'helper.php';
    redirect("../index.php?page=register", 303);
}
}

```

Verify registration

```
<?php
require_once '../config/connection.php';

if (isset($_GET['code'])) {
    $code = $_GET['code'];

    $upit = "SELECT * FROM users WHERE code=:code";

    $priprema = $connection->prepare($upit);
    $priprema->bindParam(":code", $code);

    try {
        $priprema->execute();
        if ($priprema->rowCount() == 1) {
            $user = $priprema->fetch(); //dohvatamo samo jednog

            $name = $user['name'];

            $isActive = 1;

            $upit1 = "UPDATE users SET isActive=:isActive WHERE
code=:code";

            $priprema1 = $connection->prepare($upit1);

            $priprema1->bindParam(":isActive", $isActive);
            $priprema1->bindParam(":code", $code);

            try {
                $priprema1->execute();
                $_SESSION['uspesnaVerifikacija'] = "$name, You have
successfully activated your account";
                include_once 'helper.php';
                redirect('../index.php?page=login', 303);
            } catch (PDOException $e) {
                echo $e;
            }
        } else {
            redirect('404.php', 404);
        }
    } catch (PDOException $e) {
        echo $e;
    }
}
```

```
}
```

## Css

```
.mojaIkonica {
    color: black;
    margin-right: 20px;
}

.mojeDugme{
    margin: 0 5px;
}

.mojA{
    color: black;
}

.tranzicije {
    transition: 1s;
}

#shopProducts a {
    color: black;
}

.formaUspeh {
    color: rgb(29, 172, 29);
}

.invisible {
    display: none;
}

.formaGreska {
    color: red !important;
    display: block;
}

.bojaIkonice {
    color: black;
}

.bojaIkonice:hover {
    color: gray;
}

.fas{
```

```

        color:black;
    }
    #cartPopup{
        display: none;
        justify-content: center;
        align-items: center;
        margin-top: 10%;
    }
    .image-cart{
        height: 15vh;
    }

```

## Main.js

```

$(document).ready(function () {

    getMenu();

    getSliderImages();

    getTopProducts();

    getNewArrivals();

    getPaymentOffers();

    getSocials();

    getGalleryImages();

    addToCart();


    $("#btnClear").click(clearCartBtn);

    $("#btnRegister").click(proveraRegistracije);

    $("#btnLogin").click(proveraLogovanje);


    if(window.location.href.indexOf("admin")){

        adminPanel();

    }

});

//funkcija za ciscenje cart iz localStorage

function clearCartBtn(){

    localStorage.removeItem("cart");

```

```

        localStorage.removeItem("brojProizvodaLS");

        //displayCartData();

        //printNumberOfProducts();

        //location.reload();

        $("#products").slideToggle();

        $("#buySuccess").slideToggle();

        $("#cart-sum").slideToggle();

        $("#btnClear").slideToggle();

    }

    //funkcije za dodavanje proizvoda u korpu
    function addToCart(){

        $(document).on('click', '.btnAddToCart', function(){

            //console.log(this.dataset.id);

            let idP=(this.dataset.id);

            //console.log(idP);

            $('#shopProducts').slideToggle();

            $('#cartPopup').slideToggle();

            setTimeout(cartAdd, 500);

            //novo

            let productsCart = getLS("cart");

            if(productsCart == null){

                addFirstItemToCart();

                printNumberOfProducts();

            }

            else{

                if(productIsAlreadyInCart()){

                    updateQty();

                    printNumberOfProducts();

                }

                else{

                    addItemToCart();

```

```
        printNumberOfProducts();
    }
}
```

```
function addFirstItemToCart(){
    let products = [
        {
            id: idP,
            qty: 1
        }
    ];
```

```
    setLS("cart", products);
}
```

```
function productsAlreadyInCart(){
    return productsCart.filter(el => el.id == idP).length;
}
```

```
function updateQty(){
    let productsLS = getLS("cart");
    for(let p of productsLS){
        if(p.id == idP){
            p.qty++;
            break;
        }
    }
}
```

```
    setLS("cart", productsLS);
}
```

```

function addItemToCart(){
    let productLS = getLS("cart");

    productLS.push({
        id: idP,
        qty: 1
    });

    setLS("cart", productLS);
}

})
}

//localStorage za korpu
function setLS(key,value){
    localStorage.setItem(key,JSON.stringify(value));
}
function getLS(key){
    return JSON.parse(localStorage.getItem(key));
}

function cartAdd(){
    $('#shopProducts').slideToggle();
    $('#cartPopup').slideToggle();
}

//korpa

var quantity=0;

```



```

function printNumberOfProducts(){
    let productsCart = getLS("cart");
    if(productsCart == null){
        $("#broj-proizvoda").html("(0 products)");
        console.log("problem?");
    }
    else{
        // quantity=0;
        // for(let i=0;i<productsCart.length;i++){
        //     quantity+=productsCart[i].qty;
        // }
        quantity=productsCart.length;
        let numberOfProducts = quantity;
        let txt = quantity+" ";

        if(numberOfProducts == 1){
            txt = quantity+" "+"product";
        }
        else{
            txt =quantity+" products";
        }
        setLS("brojProizvodaLS",txt);
        $("#broj-proizvoda").html(`${txt}`)
    }
}

```

//admin panel

```

function adminPanel(){
    getAllProducts();
    $("#navAdminMeni li a").click(function(e){
        e.preventDefault();
    });
}

```

```

var stranica = this.dataset.name;

//console.log(stranica);

if(stranica=="products"){
    getAllProducts();
}

if(stranica == "users"){
    getAllUsers();
}

if(stranica == "brands"){
    getAllBrands();
}

if(stranica == "genders"){
    getAllGenders();
}

if(stranica == "menus"){
    getAllMenus();
}

});
}

```

```
//Nav
```

```

function getMenu(){
    $.ajax({
        url:"navigation.php",
        method:"get",
        dataType:"json",
        success:function(data){
            writeMenu(data);
            writeMenuMobile(data);
        },
        error:function(xhr){

```

```
        console.log(xhr);
    }
});
}
```

//Ispisivanje navigacije

```
function writeMenu(data){
    let output = "";
    for (let item of data) {
        output += `<li><a href="index.php?page=${item.name}">${item.name}</a><li>`
    }

    $("#navigation").html(output)
}
```

```
function writeMenuMobile(data){
    let output = "";
    for (let item of data) {
        output += `<li><a href="index.php?page=${item.name}">${item.name}</a><li>`
    }

    $(".slicknav_nav").html(output)
}
```

//Slajder

```
var i = 0;
function getSliderImages(){
    $.ajax({
        url:"models/getSliderImages.php",
        method:"get",
        dataType:"json",
```

```

    success:function(data){
        showSliderImages(data);
    },
    error:function(xhr){
        console.log(xhr);
    }
});

```

```

function showSliderImages(nizSlika) {
    $(".hero__img").html(``);

    $("#naslovSlider").html(nizSlika[i].title);
    $("#opisSlider").html(nizSlika[i].caption);
    if(i < nizSlika.length-1){
        i++;
    }
    else{
        i=0;
    }

    setTimeout("getSliderImages()", 3000);
}
}

```

//Uzimanje novih proizvoda

```

function getTopProducts(){
    $.ajax({
        url:"models/getTopProducts.php",
        method:"get",
        dataType:"json",
        success:function(data){

```

```

        writeProducts(data);
    },
    error:function(xhr){
        console.log(xhr);
    }
});
}

```

// ispis proizvoda

```

function writeProducts(data){
    let output = "";
    for (const product of data) {
        output += `<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
        <div class="single-popular-items mb-50 text-center">
            <div class="popular-img">
                
            </div>
            <div class="popular-caption">
                <h3><a
href="index.php?page=product&product=${product.id}">${product.name}</a></h3>
                <span>${product.price}</span>
            </div>
        </div>
    </div>`;
    }
    $("#productArea").html(output);
}

```

//Dohvatanje najnovijih proizvoda, odnosno koji su poslednji insertovani u tabeli products

```

function getNewArrivals(){

```

```
$.ajax({  
    url:"models/getNewArrivals.php",  
    method:"get",  
    dataType:"json",  
    success:function(data){  
        writeNewArrivals(data);  
    },  
    error:function(xhr){  
        console.log(xhr);  
    }  
});  
}
```

//Dohvatanje proizvoda za product page

```
function getProduct(){  
    $.ajax({  
        url:"models/getProduct.php",  
        method:"get",  
        dataType:"json",  
        success:function(data){  
            writeProductPage(data);  
        },  
        error:function(xhr){  
            console.log(xhr);  
        }  
    });  
}
```

//Ispis product stranice

```
function writeProductPage(data){  
    let output = "";
```

```

for (const product of data) {
  output += `<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
    <div class="single-new-pro mb-30 text-center">
      <div class="product-img">
        
      </div>
      <div class="product-caption">
        <h3><a
href="index.php?page=product&product=${product.id}">${product.name}</a></h3>
        <span>${product.price}</span>
      </div>
    </div>
  </div>`;
}

$("#product-container").html(output);
}

```

```
    </div>
  </div>`;
}
```

```
$("#newArrivals").html(output);
}
```

//Dohvatanje ikonica u crvenom div-u pri dnu stranice

```
function getPaymentOffers(){
  $.ajax({
    url:"models/getPaymentOffers.php",
    method:"get",
    dataType:"json",
    success:function(data){
      showPaymentOffers(data);
    },
    error:function(xhr){
      console.log(xhr);
    }
  });
}
```

//Ispis ikonica za crveni div

```
function showPaymentOffers(data){
  let output = "";
  for (const item of data) {
    output += `<div class="col-xl-4 col-lg-4 col-md-6">
    <div class="single-method mb-40">
      <i class="${item.img}"></i>
      <h6>${item.name}</h6>
      <p>${item.description}</p>
    </div>
  </div>`;
  }
}
```



```
    </div>
  </div>`;
}
```

```
$("#methodPaymentOffers").html(output);
}
```

```
//Dohvatanje ikonica za drustvene mreze
```

```
function getSocials(data){
    $.ajax({
        url:"models/getSocialsIcons.php",
        method:"get",
        dataType:"json",
        success:function(data){
            showSocials(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}
```

```
//Ispis ikonica za drustvene mreze
```

```
function showSocials(data) {
    let output = "";
    for (const item of data) {
        output += `<a href="${item.link}"><i class="${item.img}"></i></a>`;
    }

    $(".footer-social").html(output);
}
```

//Dohvatanje slika za razlicit prikaz u dva diva

```
function getGalleryImages(){  
    $.ajax({  
        url:"models/getGalleryImages.php",  
        method:"get",  
        dataType:"json",  
        success:function(data){  
            showGalleryArea(data);  
            showWatchOfChoice(data);  
        },  
        error:function(xhr){  
            console.log(xhr);  
        }  
    });  
}
```

//Prikaz dva proizvoda sa razlicitom strukturom divova

```
function showWatchOfChoice(data){  
    let br = 2;  
    let output = "";  
    for (const item of data) {  
        if(item.purpose == 1){  
            if(br % 2 == 0){  
                output += parnilspis(item);  
                br++;  
            }  
            else{  
                output += neparnilspis(item);  
                br++;  
            }  
        }  
    }  
}
```

```

    }
}

$("#watchOfChoiceGallery").html(output);
}

//helper za showWatchChoice f-ju
function parnilspis(item){
    return `<div class="row align-items-center justify-content-between">
    <div class="col-lg-5 col-md-6">
        <div class="watch-details mb-40">
            <h2>${item.title}</h2>
            <p>${item.description}</p>
            <a href="index.php?page=shop" class="btn">Show Sneakers</a>
        </div>
    </div>
    <div class="col-lg-6 col-md-6 col-sm-10">
        <div class="choice-watch-img mb-40">
            
        </div>
    </div>
    </div>`;
}

```

```

//helper za showWatchChoice f-ju
function neparnilspis(item){
    return `<div class="row align-items-center justify-content-between">
    <div class="col-lg-6 col-md-6 col-sm-10">
        <div class="choice-watch-img mb-40">
            
        </div>
    </div>
    </div>`;
}

```

```

</div>

<div class="col-lg-5 col-md-6">

  <div class="watch-details mb-40">

    <h2>${item.title}</h2>

    <p>${item.description}</p>

    <a href="index.php?page=shop" class="btn">Show Sneakers</a>

  </div>

</div>
</div>`;
}

```

//helper za ispis 4 slike u vidu mini galerije na home stranici

```

function showGalleryArea(data){
  let output = "";
  for (const item of data) {
    if(item.purpose == 2){
      output += `<div class="col-xl-3 col-lg-3 col-md-6 col-sm-6">

        <div class="single-gallery mb-30">

          <div class="gallery-img big-img" style="background-image:
url(assets/img/gallery/${item.img});"></div>

        </div>

      </div>`;
    }
  }
  $("#placeForGalleryArea").html(output);
}

```

//Registracija regex

```

function proveraRegistracije(){
  var ime=document.getElementById('name');
  var prezime=document.getElementById('lastname');

```

```
var email=document.getElementById('email');
var password=document.getElementById('password');
var repassword=document.getElementById('testpassword');
```

```
var regime=/^[A-Z]{1}[a-z]{2,29}$/;
var regprezime=/^[A-Z]{1}[a-z]{2,29}$/;
var regmail = /\w+[\-\.\.]*\w+@(\w+\.)+[A-Za-z]+$/;
var repassword = /^[A-z0-9_-]{6,12}$/;
var otpadak = new Array();
```

```
if(!ime.value.match(regime))
{
    otpadak.push("First name is invalid");
    $("#name").removeClass("formaUspeh");
    $("#imeError").removeClass("invisible");
    $("#imeError").addClass("formaGreska");

} else {
    $("#imeError").addClass("invisible");
    $("#imeError").removeClass("formaGreska");
    $("#name").addClass("formaUspeh");
}

if(!prezime.value.match(regprezime))
{
    otpadak.push("Last name is invalid");
    $("#lastNameError").removeClass("invisible");
    $("#lastNameError").addClass("formaGreska");
} else {
    $("#lastNameError").addClass("invisible");
    $("#lastNameError").removeClass("formaGreska");
    $("#lastname").addClass("formaUspeh");
}
```

```

    }
    if(!email.value.match(regmail))
    {
        otpadak.push("Email is invalid");
        $("#emailError").removeClass("invisible");
        $("#emailError").addClass("formaGreska");
    } else {
        $("#emailError").addClass("invisible");
        $("#emailError").removeClass("formaGreska");
        $("#email").addClass("formaUspeh");
    }
    if(!password.value.match(regpassword))
    {
        otpadak.push("Password is invalid");
        $("#passwordError").removeClass("invisible");
        $("#passwordError").addClass("formaGreska");
    } else {
        $("#passwordError").addClass("invisible");
        $("#passwordError").removeClass("formaGreska");
        $("#password").addClass("formaUspeh");
    }

    if(password.value != repassword.value || repassword.value == ""){
        otpadak.push("Re password is invalid");
        $("#testPasswordError").removeClass("invisible");
        $("#testPasswordError").addClass("formaGreska");
    } else {
        $("#testPasswordError").addClass("invisible");
        $("#testPasswordError").removeClass("formaGreska");
        $("#testpassword").addClass("formaUspeh");
    }
}

```

```

if(otpadak.length==0){

    return true;
}
else{

    return false;

}
}

//Logovanje regex
function proveraLogovanje(){
    var email=document.getElementById('email');
    var password=document.getElementById('password');

    var regmail = /^(\w+[\-\.\.])*\w+@(\w+\.)+[A-Za-z]+\$/;
    var regpassword = /^[A-z0-9_-]{6,12}$/;
    var otpadak = new Array();

    if(!email.value.match(regmail))
    {
        otpadak.push("Email is invalid");
        $("#emailError").removeClass("invisible");
        $("#emailError").addClass("formaGreska");
    } else {
        $("#emailError").addClass("invisible");
        $("#emailError").removeClass("formaGreska");
        $("#email").addClass("formaUspeh");
    }
}

```

```

if(!password.value.match(regpassword))
{
    otpadak.push("Password is invalid");
    $("#passwordError").removeClass("invisible");
    $("#passwordError").addClass("formaGreska");
} else {
    $("#passwordError").addClass("invisible");
    $("#passwordError").removeClass("formaGreska");
    $("#password").addClass("formaUspeh");
}
if(otpadak.length==0){

    return true;
}
else{
    //console.log(otpadak);
    return false;

}
}

```

//adminPanel f-je

//dohvata sve proizvode i vraca f-ju za ispis istih

```

function getAllProducts(){
    $.ajax({
        url:"models/getAllProducts.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //console.log(data);

```



```
        setLS("sviProizvodi",data);

        tabelalispis(data, "products");

    },
    error:function(xhr){
        console.log(xhr);
    }
});
}
```

//dohvata sve users i vraca f-ju za ispis istih

```
function getAllUsers(){
    $.ajax({
        url:"models/getAllUsers.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //console.log(data);
            tabelalispis(data, "users");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}
```

//dohvata sve brendove i vraca f-ju za ispis istih

```
function getAllBrands(){
    $.ajax({
        url:"models/getAllBrands.php",
        method:"get",
        dataType:"json",
```

```
    success:function(data){
        //console.log(data);
        tabelIspis(data, "brands");
    },
    error:function(xhr){
        console.log(xhr);
    }
});
}
```

//dohvata sve genere i vraca f-ju za ispis istih

```
function getAllGenders(){
    $.ajax({
        url:"models/getAllGenders.php",
        method:"get",
        dataType:"json",
        success:function(data){
            tabelIspis(data, "gender");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}
```

```
function getAllMenus(){
    $.ajax({
        url:"models/getAllMenus.php",
        method:"get",
        dataType:"json",
```

```

    success:function(data){
        tabelalispis(data, "menus");
    },
    error:function(xhr){
        console.log(xhr);
    }
});
}

```

//f-ja za delete recorda, prosledjuje se ime tabele i id recorda koji se brise. U zavisnosti od tabele koja je poslata poziva se php fajl koji ce obrisati record sa prosledjenim id-jem

```

function deleteRecord(tableName,id){
    let r = confirm("Are you sure you want to delete this?");

    if(r){
        $.ajax({
            url:"models/delete"+ tableName +".php",
            method:"POST",
            dataType:"json",
            data:{
                "delete": true,
                "poslatiID": id
            },
            success:function(data){
                var responseData = data;
                alert(responseData.msg);

                if(responseData.productDeleted){
                    getAllProducts();
                }

                if(responseData.userDeleted){

```

```

        getAllUsers();
    }
    if(responseData.brandDeleted){
        getAllBrands();
    }
    if(responseData.genderDeleted){
        getAllGenders();
    }
    if(responseData.menuDeleted){
        getAllMenus();
    }
},
error:function(xhr){
    console.log(xhr);
}
});
}

}

//helper za ispis tabele
function tabelalispis(data, tableName){
    let output = `<input type="button" class="btn btn-danger mt-4 mb-4" value="Add New
    ${tableName}" id="btnAdd" onclick="addNewRecord('${tableName}')">`;
    output += `<table class="table table-striped">
    <thead class="thead-dark">
    <tr>`;

    let nizImena = Object.keys(data[0]);
    for(let i=0; i<nizImena.length;i++){
        output += `<th scope="col">${nizImena[i]}</th>`;
    }
}

```

```
}
```

```
output += `
```

```
<th scope="col">Edit</th>
```

```
<th scope="col">Delete</th>
```

```
</tr></thead>
```

```
<tbody>`;
```

```
for (const item of data) {
```

```
    output += `<tr>`;
```

```
    var br = 1;
```

```
    for (const cell of Object.values(item)) {
```

```
        if (br == 5 && tableName === "products"){
```

```
            output += `<td></td>`;
```

```
            br=1;
```

```
            continue;
```

```
        }
```

```
        br++
```

```
        output += `<td>${cell}</td>`;
```

```
    }
```

```
    output += `<td><a class="bojalkonice"><i class="fas fa-edit"></i></a></td>
```

```
    <td><a class="bojalkonice" onclick="deleteRecord('${tableName}', ${item.id})"><i  
class="fas fa-trash-alt"></i></a></td>
```

```
</tr>`;
```

```
}
```

```
output += `</tbody></table>`;
```

```
    output += `<input type="button" class="btn btn-danger mt-4 mb-4" value="Add New  
${tableName}" id="btnAdd" onclick="addNewRecord('${tableName}')">`;
```

```
$("#adminIspis").html(output);
```

```
}
```

```
function addNewRecord(tableName){
```

```
    if(tableName == "products"){
```

```
        formAddProduct();
```

```
    }
```

```
    if(tableName == "users"){
```

```
        formAddUser();
```

```
    }
```

```
    if(tableName == "brands"){
```

```
        formAddBrand();
```

```
    }
```

```
    if(tableName == "gender"){
```

```
        formAddGender();
```

```
    }
```

```
    if(tableName == "menus"){
```

```
        formAddMenu();
```

```
    }
```

```
}
```

```
//ispis forme za dodavanje novog proizvoda
```

```
function formAddProduct(){
```

```
    $.ajax({
```

```
        url:"models/getBrandsAndGenders.php",
```

```
        method:"get",
```

```
        dataType:"json",
```

```
        success:function(data){
```

```
            //response = JSON.parse(data);
```

```
            //console.log(data.genders[0]);
```

```
            var forma="
```

forma+=`

```
<h2>Add Product</h2>
```

```
<div>
```

```
  <form method="POST" action="">
```

```
    <div class="col-md-12 form-group p_star">
```

```
      <input type="text" class="form-control" id="productName" name="productName"
value="" placeholder="Product Name">
```

```
    </div>
```

```
    <span class="col-md-12 form-group invisible" id="productError">
```

```
      Enter valid product name!
```

```
    </span>
```

```
  <div class="col-md-12 form-group p_star">
```

```
    <input type="text" class="form-control" id="price" name="price" value=""
placeholder="Price">
```

```
  </div>
```

```
  <span class="col-md-12 form-group invisible" id="priceError">
```

```
    Enter valid product price!
```

```
  </span>
```

```
  <div class="col-md-12 form-group p_star">
```

```
    <input type="text" class="form-control" id="oldPrice" name="oldPrice" value=""
placeholder="Old Price">
```

```
  </div>
```

```
  <span class="col-md-12 form-group invisible" id="oldPriceError">
```

```
    Enter valid product old price!
```

```
  </span>
```

```
<label>Choose image</label>
```

```
<input type="file" id="imgFile">
```

```
<div class="col-md-12 form-group p_star">
```

```
        <input type="text" class="form-control" id="description" name="description"
value="" placeholder="Enter description">
```

```
    </div>
```

```
    <span class="col-md-12 form-group invisible" id="oldPriceError">
```

```
        Enter valid description!
```

```
    </span>
```

```
    <select id="selectGender">
```

```
        <option value="0">Genders</option>`
```

```
    for(let i=0;i<data.genders.length;i++){
```

```
        forma+=`<option value="${data.genders[i].id}">${data.genders[i].name}</option>`
```

```
    }
```

```
    forma+=`</select>
```

```
    <select id="selectBrand">
```

```
        <option value="0">Brands</option>`
```

```
    for(let i=0;i<data.brands.length;i++){
```

```
        forma+=`<option value="${data.brands[i].id}">${data.brands[i].name}</option>`
```

```
    }
```

```
    forma+=`</select>
```

```
    <div class="col-md-12 form-group mt-5">
```

```
        <button type="button" value="button" name="btnInsertProduct"
id="btnInsertProduct" class="btn btn-danger">
```

```
            Add New Product
```

```
        </button>
```

```
    </div>
```

```
    <div id="errorAdmin"></div>
```

```
    <div id="successAdmin"></div>
```

```
</form>
```

```
</div>`
```



```

$("#adminIspis").html(forma)

$("#btnInsertProduct").click(addNewProduct);
},
error:function(xhr){
    console.log(xhr);
}

});
}

function addNewProduct(){

    var imgFile=document.getElementById("imgFile").files[0]
    var name=document.getElementById("productName").value
    var price=document.getElementById("price").value
    var oldPrice=document.getElementById("oldPrice").value
    var description=document.getElementById("description").value
    var gender=document.getElementById("selectGender").value
    var brand=document.getElementById("selectBrand").value

    var dataForSend= new FormData();

    dataForSend.append("imgFile",imgFile)
    dataForSend.append("productName",name)
    dataForSend.append("price",price)
    dataForSend.append("oldPrice",oldPrice)
    dataForSend.append("description",description)
    dataForSend.append("selectGender",gender)

```

```
dataForSend.append("selectBrand",brand)
```

```
dataForSend.append("insert","true")
```

```
$.ajax({  
    url:"models/insertNewProduct.php",  
    method:"post",  
    dataType:"json",  
    processData:false,  
    contentType:false,  
    data:dataForSend,  
    success:function(data){  
        console.log(data);  
        $("#successAdmin").html(data);  
        $("#successAdmin").addClass("alert alert-success p-3");  
    },  
    error(xhr){  
        console.log(xhr);  
        $("#errorAdmin").html(JSON.parse(xhr.responseText))  
        $("#errorAdmin").addClass("alert alert-warning p-3")  
    }  
});  
}
```

```
function formAddUser(){  
    $.ajax({  
        url:"models/getRoles.php",  
        method:"get",  
        dataType:"json",  
        success:function(data){
```

```

// console.log(data[0].name);

var forma='
forma+=`
<h2>Add User</h2>

<div>

  <form method="POST" action="">

    <div class="col-md-12 form-group p_star">

      <input type="text" class="form-control" id="firstName" name="firstName" value=""
placeholder="Firstname">

    </div>

    <span class="col-md-12 form-group invisible" id="productError">

      Enter valid product name!

    </span>

    <div class="col-md-12 form-group p_star">

      <input type="text" class="form-control" id="lastname" name="lastname" value=""
placeholder="Lastname">

    </div>

    <span class="col-md-12 form-group invisible" id="priceError">

      Enter valid product price!

    </span>

    <div class="col-md-12 form-group p_star">

      <input type="email" class="form-control" id="email" name="email" value=""
placeholder="Email">

    </div>

    <span class="col-md-12 form-group invisible" id="oldPriceError">

      Enter valid product old price!

    </span>

    <div class="col-md-12 form-group p_star">

```

```
        <input type="password" class="form-control" id="password" name="password"
value="" placeholder="Password">
```

```
    </div>
```

```
    <span class="col-md-12 form-group invisible" id="oldPriceError">
```

```
        Enter valid description!
```

```
    </span>
```

```
    <div class="col-md-12 form-group p_star">
```

```
        <input type="password" class="form-control" id="testPassword"
name="testPassword" value="" placeholder="Re-type password">
```

```
    </div>
```

```
    <span class="col-md-12 form-group invisible" id="oldPriceError">
```

```
        Enter valid description!
```

```
    </span>
```

```
    <select id="selectRole">
```

```
        <option value="0">Roles</option>`
```

```
    for(let i=0;i<data.length;i++){
```

```
        forma+=`<option value="${data[i].id}">${data[i].name}</option>`
```

```
    }
```

```
    forma+=`</select>
```

```
    <div class="col-md-12 form-group mt-5">
```

```
        <button type="button" value="button" name="btnInsertUser" id="btnInsertUser"
class="btn btn-danger">
```

```
            Add New User
```

```
        </button>
```

```
    </div>
```

```
    <div id="errorAdmin"></div>
```

```
    <div id="successAdmin"></div>
```

```
</form>
```

```
</div>`
```

```

$("#adminIspis").html(forma)

$("#btnInsertUser").click(addNewUser);
},
error:function(xhr){
    console.log(xhr);
}

});
}

function addNewUser(){

    var firstname=document.getElementById("firstName").value
    var lastname=document.getElementById("lastname").value
    var email=document.getElementById("email").value
    var password=document.getElementById("password").value
    var testPassword=document.getElementById("testPassword").value
    var role=document.getElementById("selectRole").value

    var dataForSend= new FormData();

    dataForSend.append("firstName",firstname)
    dataForSend.append("lastname",lastname)
    dataForSend.append("email",email)
    dataForSend.append("password",password)
    dataForSend.append("testPassword",testPassword)
    dataForSend.append("selectRole",role)

    dataForSend.append("insert","true")

```

```

$.ajax({
    url:"models/insertNewUser.php",
    method:"post",
    dataType:"json",
    processData:false,
    contentType:false,
    data:dataForSend,
    success:function(data){
        console.log(data);
        $("#successAdmin").html(data);
        $("#successAdmin").addClass("alert alert-success p-3");
    },
    error(xhr){
        console.log(xhr);
        $("#errorAdmin").html(JSON.parse(xhr.responseText))
        $("#errorAdmin").addClass("alert alert-warning p-3")
    }
});
}

```

```

function formAddBrand(){
    var forma="";
    forma+=`
    <h2>Add Brand</h2>
    <div>
        <form method="POST" action="">
            <div class="col-md-12 form-group p_star">
                <input type="text" class="form-control" id="brandName" name="brandName"
value="" placeholder="Brand name">

```

```

</div>

<div class="col-md-12 form-group mt-5">
    <button type="button" value="button" name="btnInsertBrand" id="btnInsertBrand"
class="btn btn-danger">
        Add New Brand
    </button>
</div>

<div id="errorAdmin"></div>
<div id="successAdmin"></div>
</form>
</div>`

```

```

$("#adminIspis").html(forma)

```

```

$("#btnInsertBrand").click(addNewBrand);
}

```

```

function addNewBrand(){
    var brandName=document.getElementById("brandName").value

    var dataForSend= new FormData();

    dataForSend.append("brandName",brandName)
    dataForSend.append("insert","true")

    $.ajax({
        url:"models/insertNewBrand.php",
        method:"post",
        dataType:"json",
        processData:false,

```

```

contentType:false,
data:dataForSend,
success:function(data){
    console.log(data);
    $("#successAdmin").html(data);
    $("#successAdmin").addClass("alert alert-success p-3");
},
error(xhr){
    console.log(xhr);
    $("#errorAdmin").html(JSON.parse(xhr.responseText))
    $("#errorAdmin").addClass("alert alert-warning p-3")
}
});
}

```

```

function formAddGender(){
    var forma="";
    forma+=`
    <h2>Add Gender</h2>
    <div>
        <form method="POST" action="">
            <div class="col-md-12 form-group p_star">
                <input type="text" class="form-control" id="genderName" name="genderName"
value="" placeholder="Gender name">
            </div>

            <div class="col-md-12 form-group mt-5">
                <button type="button" value="button" name="btnInsertGender"
id="btnInsertGender" class="btn btn-danger">

                    Add New Gender

                </button>
            </div>
    `
}

```



```
        <div id="errorAdmin"></div>

        <div id="successAdmin"></div>

    </form>
</div>
```

```
$("#adminIspis").html(forma)
```

```
$("#btnInsertGender").click(addNewGender);
```

```
}
```

```
function addNewGender(){
```

```
    var genderName=document.getElementById("genderName").value
```

```
    var dataForSend= new FormData();
```

```
    dataForSend.append("genderName",genderName)
```

```
    dataForSend.append("insert","true")
```

```
$.ajax({
```

```
    url:"models/insertNewGender.php",
```

```
    method:"post",
```

```
    dataType:"json",
```

```
    processData:false,
```

```
    contentType:false,
```

```
    data:dataForSend,
```

```
    success:function(data){
```

```
        console.log(data);
```

```
        $("#successAdmin").html(data);
```

```
        $("#successAdmin").addClass("alert alert-success p-3");
```

```
    },
```

```
    error(xhr){
```

```
console.log(xhr);

$("#errorAdmin").html(JSON.parse(xhr.responseText))

$("#errorAdmin").addClass("alert alert-warning p-3")

}

});

}
```

```
function formAddMenu(){
$.ajax({
    url:"models/getPrivileges.php",
    method:"get",
    dataType:"json",
    success:function(data){
        // console.log(data[0].name);
        var forma="";
        forma+=`
            <h2>Add Menu Item</h2>
            <div>
                <form method="POST" action="">
                    <div class="col-md-12 form-group p_star">
                        <input type="text" class="form-control" id="menuItemName"
name="menuItemName" value="" placeholder="Menu item name">
                    </div>

                    <div class="col-md-12 form-group p_star">
                        <input type="text" class="form-control" id="menuItemPosition"
name="menuItemPosition" value="" placeholder="Menu item position">
                    </div>

                    <select id="selectPrivilege">
                        <option value="0">Privilege</option>`
        for(let i=0;i<data.length;i++){
```

```

        forma+=`<option value="${data[i].id}">${data[i].name}</option>`
    }
    forma+=`</select>

    <div class="col-md-12 form-group mt-5">
        <button type="button" value="button" name="btnInsertMenu"
id="btnInsertMenu" class="btn btn-danger">
            Add New Menu Item
        </button>
    </div>
    <div id="errorAdmin"></div>
    <div id="successAdmin"></div>
</form>
</div>`

```

```

$("#adminIspis").html(forma)

```

```

$("#btnInsertMenu").click(addNewMenu);
},
error:function(xhr){
    console.log(xhr);
}

});
}

```

```

function addNewMenu(){
    var menuItemName=document.getElementById("menuItemName").value
    var menuItemPosition=document.getElementById("menuItemPosition").value
    var privilege=document.getElementById("selectPrivilege").value

```

```

var dataForSend= new FormData();

dataForSend.append("menuItemName",menuItemName)
dataForSend.append("menuItemPosition",menuItemPosition)
dataForSend.append("selectPrivilege",privilege)
dataForSend.append("insert","true")

$.ajax({
    url:"models/insertNewMenuItem.php",
    method:"post",
    dataType:"json",
    processData:false,
    contentType:false,
    data:dataForSend,
    success:function(data){
        console.log(data);
        $("#successAdmin").html(data);
        $("#successAdmin").addClass("alert alert-success p-3");
    },
    error(xhr){
        console.log(xhr);
        $("#errorAdmin").html(JSON.parse(xhr.responseText))
        $("#errorAdmin").addClass("alert alert-warning p-3")
    }
});
}

```

```

//ispis i brisanje proizvoda u korpi
$(document).ready(function(){
    let productsCartLS=getLS("cart");

```

```

if (productsCartLS==null || productsCartLS.length<1){
    $('#products').html(`<div id="greskaShop"><h1>Your shopping cart is
empty!</h1></div>`);
    $('#btnClear').slideToggle();
}
else{
    displayCartData();
    generateTable(productInfo);
}
});
var productsCartLS=getLS("cart");
//funkcija za dohvacanje proizvoda za korpu

```

```

function displayCartData(){
    productInfo=getLS("sviProizvodi");
    //console.log(productInfo);
    productInfo=productInfo.filter(p=>{
        for(let product of productsCartLS){
            if(p.id == product.id){
                p.quantity=product.qty;
                return true;
            }
        }
    })
    //console.log(productInfo);
}

```

```

function generateTable(){
    let html=`
    <table class="table">

```

```

<thead class="thead-dark">

  <tr>

    <th scope="col">Product</th>

    <th scope="col">Image</th>

    <th scope="col">Qty</th>

    <th scope="col">Price</th>

    <th scope="col">Total</th>

    <th scope="col">Remove</th>

  </tr>

</thead>

<tbody>`

let total=0;

for(let p of productInfo){

  html+=`

    <tr>

      <th scope="row">${p.name}</th>

      <td></td>

      <td>${p.quantity}</td>

      <td>${p.price}</td>

      <td>${parseFloat(p.price)*p.quantity}</td>

      <td><button onclick="removeFromCart(${p.id})" class="btn"
value="remove">remove</button></td>

    </tr>`

  total+=parseInt((p.price)*p.quantity);

}

//console.log(total);

$("#cart-sum").html(`<p class="text-center">total: ${total}</p>`);

html+=`

</tbody>

</table>

`

```

```

    $("#products").html(html);
}

function removeFromCart(id){
    let korpa=getLS("cart");
    let filtered=korpa.filter(p=>p.id != id);
    setLS("cart",filtered);
    displayCartData();
    printNumberOfProducts();
    location.reload();
}

var quantity=0;

function printNumberOfProducts(){
    let productsCart = getLS("cart");
    if(productsCart == null){
        $("#broj-proizvoda").html("(0 products)");
        // console.log("problem?");
    }
    else{
        quantity=productsCart.length;
        let numberOfProducts = quantity;
        let txt = quantity+" ";

        if(numberOfProducts == 1){
            txt = quantity+" "+"product";
        }
        else{
            txt =quantity+" products";
        }
        setLS("brojProizvodaLS",txt);
        $("#broj-proizvoda").html(`${txt}`)
    }
}

```

```
}  
}
```

Main.js iz Visual Studio Code (prethodni je menjan direktno na 000webhost pa je prekopiran plain text)

```
$(document).ready(function () {  
    getMenu();  
    getSliderImages();  
    getTopProducts();  
    getNewArrivals();  
    getPaymentOffers();  
    getSocials();  
    getGalleryImages();  
    addToCart();  
  
    $("btnRegister").click(proveraRegistracije);  
    $("btnLogin").click(proveraLogovanje);  
  
    if(window.location.href.indexOf("admin")){  
        adminPanel();  
    }  
});  
  
//funkcije za dodavanje proizvoda u korpu  
function addToCart(){  
    $(document).on('click', '.btnAddToCart', function(){  
        //console.log(this.dataset.id);  
        let idP=(this.dataset.id);  
        //console.log(idP);  
        $('#shopProducts').slideToggle();  
        $('#cartPopup').slideToggle();  
        setTimeout(cartAdd, 500);  
        //novo  
        let productsCart = getLS("cart");  
        if(productsCart == null){  
            addFirstItemToCart();  
            printNumberOfProducts();  
        }  
        else{  
            if(productIsAlreadyInCart()){  
                updateQty();  
                printNumberOfProducts();  
            }  
            else{  
                addItemToCart();  
            }  
        }  
    });  
}
```



```

        printNumberOfProducts();
    }
}

function addFirstItemToCart(){
    let products = [
        {
            id: idP,
            qty: 1
        }
    ];

    setLS("cart", products);
}

function productIsAlreadyInCart(){
    return productsCart.filter(el => el.id == idP).length;
}

function updateQty(){
    let productsLS = getLS("cart");
    for(let p of productsLS){
        if(p.id == idP){
            p.qty++;
            break;
        }
    }

    setLS("cart", productsLS);
}

function addItemToCart(){
    let productLS = getLS("cart");

    productLS.push({
        id: idP,
        qty: 1
    });

    setLS("cart", productLS);
}

}))
}

//localStorage za korpu
function setLS(key,value){
    localStorage.setItem(key,JSON.stringify(value));
}

```

```

function getLS(key){
    return JSON.parse(localStorage.getItem(key));
}

function cartAdd(){
    $('#shopProducts').slideToggle();
    $('#cartPopup').slideToggle();
}

//korpa

var quantity=0;
function printNumberOfProducts(){
    let productsCart = getLS("cart");
    if(productsCart == null){
        $("#broj-proizvoda").html("(0 products)");
        console.log("problem?");
    }
    else{
        // quantity=0;
        // for(let i=0;i<productsCart.length;i++){
        //     quantity+=productsCart[i].qty;
        // }
        quantity=productsCart.length;
        let numberOfProducts = quantity;
        let txt = quantity+" ";

        if(numberOfProducts == 1){
            txt = quantity+" "+"product";
        }
        else{
            txt =quantity+" products";
        }
        setLS("brojProizvodaLS",txt);
        $("#broj-proizvoda").html(`${txt}`)
    }
}

//admin panel
function adminPanel(){
    getAllProducts();
    $("#navAdminMeni li a").click(function(e){
        e.preventDefault();
        var stranica = this.dataset.name;
        //console.log(stranica);
        if(stranica=="products"){
            getAllProducts();
        }
    })
}

```

```

        if(stranica == "users"){
            getAllUsers();
        }
        if(stranica == "brands"){
            getAllBrands();
        }
        if(stranica == "genders"){
            getAllGenders();
        }
        if(stranica == "menus"){
            getAllMenus();
        }
    });
}

//Nav
function getMenu(){
    $.ajax({
        url:"navigation.php",
        method:"get",
        dataType:"json",
        success:function(data){
            writeMenu(data);
            writeMenuMobile(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//Ispisivanje navigacije
function writeMenu(data){
    let output = "";
    for (let item of data) {
        output += `<li><a
href="index.php?page=${item.name}">${item.name}</a><li>`
    }

    $("#navigation").html(output)
}

function writeMenuMobile(data){
    let output = "";
    for (let item of data) {
        output += `<li><a
href="index.php?page=${item.name}">${item.name}</a><li>`
    }
}

```

```

        $(".slicknav_nav").html(output)
    }

    //Slajder
    var i = 0;
    function getSliderImages(){
        $.ajax({
            url:"models/getSliderImages.php",
            method:"get",
            dataType:"json",
            success:function(data){
                showSliderImages(data);
            },
            error:function(xhr){
                console.log(xhr);
            }
        });

        function showSliderImages(nizSlika) {
            $(".hero__img").html(``);
            $("#naslovSlider").html(nizSlika[i].title);
            $("#opisSlider").html(nizSlika[i].caption);
            if(i < nizSlika.length-1){
                i++;
            }
            else{
                i=0;
            }

            setTimeout("getSliderImages()", 3000);
        }
    }

    //Uzimanje novih proizvoda
    function getTopProducts(){
        $.ajax({
            url:"models/getTopProducts.php",
            method:"get",
            dataType:"json",
            success:function(data){
                writeProducts(data);
            },
            error:function(xhr){
                console.log(xhr);
            }
        });
    }
}

```

```

// ispis proizvoda
function writeProducts(data){
    let output = "";
    for (const product of data) {
        output += `<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
        <div class="single-popular-items mb-50 text-center">
            <div class="popular-img">
                
            </div>
            <div class="popular-caption">
                <h3><a
href="index.php?page=product&product=${product.id}">${product.name}</a></
h3>
                <span>$ ${product.price}</span>
            </div>
        </div>`;
    }
    $("#producstArea").html(output);
}

//Dohvatanje najnovijih proizvoda, odnosno koji su poslednji insertovani
u tabeli products
function getNewArrivals(){
    $.ajax({
        url:"models/getNewArrivals.php",
        method:"get",
        dataType:"json",
        success:function(data){
            writeNewArrivals(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//Dohvatanje proizvoda za product page
function getProduct(){
    $.ajax({
        url:"models/getProduct.php",
        method:"get",
        dataType:"json",
        success:function(data){
            writeProductPage(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

```

```

    }
  });
}

//Ispis product stranice
function writeProductPage(data){
  let output = "";
  for (const product of data) {
    output += `<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
      <div class="single-new-pro mb-30 text-center">
        <div class="product-img">
          
        </div>
        <div class="product-caption">
          <h3><a
href="index.php?page=product&product=${product.id}">${product.name}</a></
h3>
          <span>$ ${product.price}</span>
        </div>
      </div>
    </div>`;
  }

  $("#product-container").html(output);
}

//Ispis najnovijih proizvoda
function writeNewArrivals(data){
  let output = "";
  for (const product of data) {
    output += `<div class="col-xl-4 col-lg-4 col-md-6 col-sm-6">
      <div class="single-new-pro mb-30 text-center">
        <div class="product-img">
          
        </div>
        <div class="product-caption">
          <h3><a
href="index.php?page=product&product=${product.id}">${product.name}</a></
h3>
          <span>$ ${product.price}</span>
        </div>
      </div>
    </div>`;
  }

  $("#newArrivals").html(output);
}

//Dohvatanje ikonica u crvenom div-u pri dnu stranice

```

```

function getPaymentOffers(){
    $.ajax({
        url:"models/getPaymentOffers.php",
        method:"get",
        dataType:"json",
        success:function(data){
            showPaymentOffers(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//Ispis ikonica za crveni div
function showPaymentOffers(data){
    let output = "";
    for (const item of data) {
        output += `<div class="col-xl-4 col-lg-4 col-md-6">
            <div class="single-method mb-40">
                <i class="${item.img}"></i>
                <h6>${item.name}</h6>
                <p>${item.description}</p>
            </div>
        </div>`;
    }

    $("#methodPaymentOffers").html(output);
}

//Dohvatanje ikonica za drustvene mreze
function getSocials(data){
    $.ajax({
        url:"models/getSocialsIcons.php",
        method:"get",
        dataType:"json",
        success:function(data){
            showSocials(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//Ispis ikonica za drustvene mreze
function showSocials(data) {
    let output = "";
    for (const item of data) {

```

```

        output += `<a href="${item.link}"><i
class="${item.img}"></i></a>`;
    }

    $(".footer-social").html(output);
}

//Dohvatanje slika za razlicit prikaz u dva diva
function getGalleryImages(){
    $.ajax({
        url:"models/getGalleryImages.php",
        method:"get",
        dataType:"json",
        success:function(data){
            showGalleryArea(data);
            showWatchOfChoice(data);
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//Prikaz dva proizvoda sa razlicitom strukturom divova
function showWatchOfChoice(data){
    let br = 2;
    let output = "";
    for (const item of data) {
        if(item.purpose == 1){
            if(br % 2 == 0){
                output += parniIspis(item);
                br++;
            }
            else{
                output += neparniIspis(item);
                br++;
            }
        }
    }

    $("#watchOfChoiceGallery").html(output);
}

//helper za showWatchChoice f-ju
function parniIspis(item){
    return `<div class="row align-items-center justify-content-between">
<div class="col-lg-5 col-md-6">
<div class="watch-details mb-40">
<h2>${item.title}</h2>

```



```

        <p>${item.description}</p>
        <a href="index.php?page=shop" class="btn">Show Sneakers</a>
    </div>
</div>
<div class="col-lg-6 col-md-6 col-sm-10">
    <div class="choice-watch-img mb-40">
        
    </div>
</div>
</div>`;
}

//helper za showWatchChoice f-ju
function neparniIspis(item){
    return `<div class="row align-items-center justify-content-between">
    <div class="col-lg-6 col-md-6 col-sm-10">
        <div class="choice-watch-img mb-40">
            
        </div>
    </div>
    <div class="col-lg-5 col-md-6">
        <div class="watch-details mb-40">
            <h2>${item.title}</h2>
            <p>${item.description}</p>
            <a href="index.php?page=shop" class="btn">Show Sneakers</a>
        </div>
    </div>
</div>`;
}

//helper za ispis 4 slike u vidu mini galerije na home stranici
function showGalleryArea(data){
    let output = "";
    for (const item of data) {
        if(item.purpose == 2){
            output += `<div class="col-xl-3 col-lg-3 col-md-6 col-sm-6">
            <div class="single-gallery mb-30">
                <div class="gallery-img big-img" style="background-image:
url(assets/img/gallery/${item.img});"></div>
            </div>
        </div>`;
        }
    }
    $("#placeForGalleryArea").html(output);
}

//Registracija regex
function proveraRegistracije(){
    var ime=document.getElementById('name');

```

```

var prezime=document.getElementById('lastname');
var email=document.getElementById('email');
var password=document.getElementById('password');
var repassword=document.getElementById('testpassword');

var regime=/^[A-Z]{1}[a-z]{2,29}$/;
var regprezime=/^[A-Z]{1}[a-z]{2,29}$/;
var regmail = /^(\\w+([\\-\\.])*\\w+@((\\w+\\.)+[A-Za-z]+)$/;
var regpassword = /^[A-z0-9_-]{6,12}$/;
var otpadak = new Array();

if(!ime.value.match(regime))
{
    otpadak.push("First name is invalid");
    $("#name").removeClass("formaUspeh");
    $("#imeError").removeClass("invisible");
    $("#imeError").addClass("formaGreska");

} else {
    $("#imeError").addClass("invisible");
    $("#imeError").removeClass("formaGreska");
    $("#name").addClass("formaUspeh");
}

if(!prezime.value.match(regprezime))
{
    otpadak.push("Last name is invalid");
    $("#lastNameError").removeClass("invisible");
    $("#lastNameError").addClass("formaGreska");
} else {
    $("#lastNameError").addClass("invisible");
    $("#lastNameError").removeClass("formaGreska");
    $("#lastname").addClass("formaUspeh");
}

if(!email.value.match(regmail))
{
    otpadak.push("Email is invalid");
    $("#emailError").removeClass("invisible");
    $("#emailError").addClass("formaGreska");
} else {
    $("#emailError").addClass("invisible");
    $("#emailError").removeClass("formaGreska");
    $("#email").addClass("formaUspeh");
}

if(!password.value.match(regpassword))
{
    otpadak.push("Password is invalid");
    $("#passwordError").removeClass("invisible");
    $("#passwordError").addClass("formaGreska");
} else {

```

```

        $("#passwordError").addClass("invisible");
        $("#passwordError").removeClass("formaGreska");
        $("#password").addClass("formaUspeh");
    }

    if(password.value != repassword.value || repassword.value == ""){
        otpadak.push("Re password is invalid");
        $("#testPasswordError").removeClass("invisible");
        $("#testPasswordError").addClass("formaGreska");
    } else {
        $("#testPasswordError").addClass("invisible");
        $("#testPasswordError").removeClass("formaGreska");
        $("#testpassword").addClass("formaUspeh");
    }
    if(otpadak.length==0){

        return true;
    }
    else{

        return false;
    }
}

//Logovanje regex
function proveraLogovanje(){
    var email=document.getElementById('email');
    var password=document.getElementById('password');

    var regmail = /^(\\w+[-\\.])*\\w+@(\\w+\\.)+[A-Za-z]+$ /;
    var regpassword = /^[A-z0-9_-]{6,12}$/;
    var otpadak = new Array();

    if(!email.value.match(regmail))
    {
        otpadak.push("Email is invalid");
        $("#emailError").removeClass("invisible");
        $("#emailError").addClass("formaGreska");
    } else {
        $("#emailError").addClass("invisible");
        $("#emailError").removeClass("formaGreska");
        $("#email").addClass("formaUspeh");
    }
    if(!password.value.match(regpassword))
    {
        otpadak.push("Password is invalid");
        $("#passwordError").removeClass("invisible");

```

```

        $("#passwordError").addClass("formaGreska");
    } else {
        $("#passwordError").addClass("invisible");
        $("#passwordError").removeClass("formaGreska");
        $("#password").addClass("formaUspeh");
    }
    if(otpadak.length==0){

        return true;
    }
    else{
        //console.log(otpadak);
        return false;
    }
}

```

```

//adminPanel f-je

```

```

//dohvata sve proizvode i vraca f-ju za ispis istih

```

```

function getAllProducts(){
    $.ajax({
        url:"models/getAllProducts.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //console.log(data);
            setLS("sviProizvodi",data);
            tabelaIspis(data, "products");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

```

```

//dohvata sve users i vraca f-ju za ispis istih

```

```

function getAllUsers(){
    $.ajax({
        url:"models/getAllUsers.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //console.log(data);
            tabelaIspis(data, "users");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

```

```

    });
}

//dohvata sve brendove i vraca f-ju za ispis istih
function getAllBrands(){
    $.ajax({
        url:"models/getAllBrands.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //console.log(data);
            tabelaIspis(data, "brands");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

//dohvata sve genere i vraca f-ju za ispis istih
function getAllGenders(){
    $.ajax({
        url:"models/getAllGenders.php",
        method:"get",
        dataType:"json",
        success:function(data){
            tabelaIspis(data, "gender");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

function getAllMenus(){
    $.ajax({
        url:"models/getAllMenus.php",
        method:"get",
        dataType:"json",
        success:function(data){
            tabelaIspis(data, "menus");
        },
        error:function(xhr){
            console.log(xhr);
        }
    });
}

```

//f-ja za delete recorda, prosledjuje se ime tabele i id recorda koji se brise. U zavisnosti od tabele koja je poslata poziva se php fajl koji ce obrisati record sa prosledjenim id-jem

```
function deleteRecord(tableName,id){
    let r = confirm("Are you sure you want to delete this?");

    if(r){
        $.ajax({
            url:"models/delete"+ tableName +".php",
            method:"POST",
            dataType:"json",
            data:{
                "delete": true,
                "poslatiID": id
            },
            success:function(data){
                var responseData = data;
                alert(responseData.msg);

                if(responseData.productDeleted){
                    getAllProducts();
                }
                if(responseData.userDeleted){
                    getAllUsers();
                }
                if(responseData.brandDeleted){
                    getAllBrands();
                }
                if(responseData.genderDeleted){
                    getAllGenders();
                }
                if(responseData.menuDeleted){
                    getAllMenus();
                }
            },
            error:function(xhr){
                console.log(xhr);
            }
        });
    }
}

//helper za ispis tabele
function tabelaIspis(data, tableName){
    let output = `<input type="button" class="btn btn-danger mt-4 mb-4"
value="Add New ${tableName}" id="btnAdd"
onclick="addNewRecord('${tableName}')">`;
    output += `<table class="table table-striped">
```

```

<thead class="thead-dark">
  <tr>`;

  let nizImena = Object.keys(data[0]);
  for(let i=0; i<nizImena.length;i++){
    output += `<th scope="col">${nizImena[i]}</th>`;
  }

  output += `
  <th scope="col">Edit</th>
  <th scope="col">Delete</th>
</tr></thead>
<tbody>`;

  for (const item of data) {
    output += `<tr>`;
    var br = 1;
    for (const cell of Object.values(item)) {
      if(br == 5 && tableName=="products"){
        output += `<td></td>`;
        br=1;
        continue;
      }
      br++;
      output += `<td>${cell}</td>`;
    }
    output += `<td><a class="bojaIkonice"><i class="fas fa-
edit"></i></a></td>
    <td><a class="bojaIkonice" onclick="deleteRecord('${tableName}',
${item.id})"><i class="fas fa-trash-alt"></i></a></td>
    </tr>`;
  }

  output += `</tbody></table>`;

  output += `<input type="button" class="btn btn-danger mt-4 mb-4"
value="Add New ${tableName}" id="btnAdd"
onclick="addNewRecord('${tableName}')">`;

  $("#adminIspis").html(output);
}

function addNewRecord(tableName){
  if(tableName == "products"){
    formAddProduct();
  }
  if(tableName == "users"){

```

```

        formAddUser();
    }
    if(tableName == "brands"){
        formAddBrand();
    }
    if(tableName == "gender"){
        formAddGender();
    }
    if(tableName == "menus"){
        formAddMenu();
    }
}

//ispis forme za dodavanje novog proizvoda
function formAddProduct(){
    $.ajax({
        url:"models/getBrandsAndGenders.php",
        method:"get",
        dataType:"json",
        success:function(data){
            //response = JSON.parse(data);
            //console.log(data.genders[0]);
            var forma='`
forma+=`
        <h2>Add Product</h2>
        <div>
            <form method="POST" action="">
                <div class="col-md-12 form-group p_star">
                    <input type="text" class="form-control"
id="productName" name="productName" value="" placeholder="Product Name">
                </div>
                <span class="col-md-12 form-group invisible"
id="productError">
                    Enter valid product name!
                </span>

                <div class="col-md-12 form-group p_star">
                    <input type="text" class="form-control" id="price"
name="price" value="" placeholder="Price">
                </div>
                <span class="col-md-12 form-group invisible"
id="priceError">
                    Enter valid product price!
                </span>

                <div class="col-md-12 form-group p_star">
                    <input type="text" class="form-control" id="oldPrice"
name="oldPrice" value="" placeholder="Old Price">
                </div>

```



```

        <span class="col-md-12 form-group invisible"
id="oldPriceError">
            Enter valid product old price!
        </span>

        <label>Choose image</label>
        <input type="file" id="imgFile">

        <div class="col-md-12 form-group p_star">
            <input type="text" class="form-control"
id="description" name="description" value="" placeholder="Enter
description">
        </div>
        <span class="col-md-12 form-group invisible"
id="oldPriceError">
            Enter valid description!
        </span>

        <select id="selectGender">
            <option value="0">Genders</option>`
            for(let i=0;i<data.genders.length;i++){
                forma+=`<option
value="${data.genders[i].id}">${data.genders[i].name}</option>`
            }
            forma+=`</select>

        <select id="selectBrand">
            <option value="0">Brands</option>`
            for(let i=0;i<data.brands.length;i++){
                forma+=`<option
value="${data.brands[i].id}">${data.brands[i].name}</option>`
            }
            forma+=`</select>

        <div class="col-md-12 form-group mt-5">
            <button type="button" value="button"
name="btnInsertProduct" id="btnInsertProduct" class="btn btn-danger">
                Add New Product
            </button>
        </div>
        <div id="errorAdmin"></div>
        <div id="successAdmin"></div>
    </form>
</div>`

$("#adminIspis").html(forma)

$("#btnInsertProduct").click(addNewProduct);

```

```

    },
    error:function(xhr){
        console.log(xhr);
    }

});
}

function addNewProduct(){

    var imgFile=document.getElementById("imgFile").files[0]
    var name=document.getElementById("productName").value
    var price=document.getElementById("price").value
    var oldPrice=document.getElementById("oldPrice").value
    var description=document.getElementById("description").value
    var gender=document.getElementById("selectGender").value
    var brand=document.getElementById("selectBrand").value

    var dataForSend= new FormData();

    dataForSend.append("imgFile",imgFile)
    dataForSend.append("productName",name)
    dataForSend.append("price",price)
    dataForSend.append("oldPrice",oldPrice)
    dataForSend.append("description",description)
    dataForSend.append("selectGender",gender)
    dataForSend.append("selectBrand",brand)

    dataForSend.append("insert","true")

    $.ajax({
        url:"models/insertNewProduct.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:dataForSend,
        success:function(data){
            console.log(data);
            $("#successAdmin").html(data);
            $("#successAdmin").addClass("alert alert-success p-3");
        },
        error(xhr){
            console.log(xhr);
            $("#errorAdmin").html(JSON.parse(xhr.responseText))
            $("#errorAdmin").addClass("alert alert-warning p-3")
        }
    })
}

```

```

    });
}

function formAddUser(){
    $.ajax({
        url:"models/getRoles.php",
        method:"get",
        dataType:"json",
        success:function(data){
            // console.log(data[0].name);
            var forma=' '
            forma+=`
                <h2>Add User</h2>
                <div>
                    <form method="POST" action="">
                        <div class="col-md-12 form-group p_star">
                            <input type="text" class="form-control"
id="firstName" name="firstName" value="" placeholder="Firstname">
                        </div>
                        <span class="col-md-12 form-group invisible"
id="productError">
                            Enter valid product name!
                        </span>

                        <div class="col-md-12 form-group p_star">
                            <input type="text" class="form-control" id="lastname"
name="lastname" value="" placeholder="Lastname">
                        </div>
                        <span class="col-md-12 form-group invisible"
id="priceError">
                            Enter valid product price!
                        </span>

                        <div class="col-md-12 form-group p_star">
                            <input type="email" class="form-control" id="email"
name="email" value="" placeholder="Email">
                        </div>
                        <span class="col-md-12 form-group invisible"
id="oldPriceError">
                            Enter valid product old price!
                        </span>

                        <div class="col-md-12 form-group p_star">
                            <input type="password" class="form-control"
id="password" name="password" value="" placeholder="Password">
                        </div>
                        <span class="col-md-12 form-group invisible"
id="oldPriceError">
                            Enter valid description!

```

```

        </span>

        <div class="col-md-12 form-group p_star">
            <input type="password" class="form-control"
id="testPassword" name="testPassword" value="" placeholder="Re-type
password">
        </div>
        <span class="col-md-12 form-group invisible"
id="oldPriceError">
            Enter valid description!
        </span>

        <select id="selectRole">
            <option value="0">Roles</option>`
            for(let i=0;i<data.length;i++){
                forma+=`<option
value="${data[i].id}">${data[i].name}</option>`
            }
            forma+=`</select>

        <div class="col-md-12 form-group mt-5">
            <button type="button" value="button"
name="btnInsertUser" id="btnInsertUser" class="btn btn-danger">
                Add New User
            </button>
        </div>
        <div id="errorAdmin"></div>
        <div id="successAdmin"></div>
    </form>
</div>`

$("#adminIspis").html(forma)

$("#btnInsertUser").click(addNewUser);
},
error:function(xhr){
    console.log(xhr);
}

});
}

function addNewUser(){

    var firstname=document.getElementById("firstName").value
    var lastname=document.getElementById("lastname").value
    var email=document.getElementById("email").value
    var password=document.getElementById("password").value
    var testPassword=document.getElementById("testPassword").value

```

```

var role=document.getElementById("selectRole").value

var dataForSend= new FormData();

dataForSend.append("firstName",firstname)
dataForSend.append("lastname",lastname)
dataForSend.append("email",email)
dataForSend.append("password",password)
dataForSend.append("testPassword",testPassword)
dataForSend.append("selectRole",role)

dataForSend.append("insert","true")

$.ajax({
  url:"models/insertNewUser.php",
  method:"post",
  dataType:"json",
  processData:false,
  contentType:false,
  data:dataForSend,
  success:function(data){
    console.log(data);
    $("#successAdmin").html(data);
    $("#successAdmin").addClass("alert alert-success p-3");
  },
  error(xhr){
    console.log(xhr);
    $("#errorAdmin").html(JSON.parse(xhr.responseText))
    $("#errorAdmin").addClass("alert alert-warning p-3")
  }
});
}

function formAddBrand(){
  var forma='';
  forma+=`
    <h2>Add Brand</h2>
    <div>
      <form method="POST" action="">
        <div class="col-md-12 form-group p_star">
          <input type="text" class="form-control"
id="brandName" name="brandName" value="" placeholder="Brand name">
        </div>

        <div class="col-md-12 form-group mt-5">
          <button type="button" value="button"
name="btnInsertBrand" id="btnInsertBrand" class="btn btn-danger">
            Add New Brand

```

```

        </button>
    </div>
    <div id="errorAdmin"></div>
    <div id="successAdmin"></div>
</form>
</div>`

$("#adminIspis").html(forma)

$("#btnInsertBrand").click(addNewBrand);
}

function addNewBrand(){
    var brandName=document.getElementById("brandName").value

    var dataForSend= new FormData();

    dataForSend.append("brandName",brandName)
    dataForSend.append("insert","true")

    $.ajax({
        url:"models/insertNewBrand.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:dataForSend,
        success:function(data){
            console.log(data);
            $("#successAdmin").html(data);
            $("#successAdmin").addClass("alert alert-success p-3");
        },
        error(xhr){
            console.log(xhr);
            $("#errorAdmin").html(JSON.parse(xhr.responseText))
            $("#errorAdmin").addClass("alert alert-warning p-3")
        }
    });
}

function formAddGender(){
    var forma='';
    forma+=`
    <h2>Add Gender</h2>
    <div>
        <form method="POST" action="">
            <div class="col-md-12 form-group p_star">
                <input type="text" class="form-control"
id="genderName" name="genderName" value="" placeholder="Gender name">
    `
}

```

```

        </div>

        <div class="col-md-12 form-group mt-5">
            <button type="button" value="button"
name="btnInsertGender" id="btnInsertGender" class="btn btn-danger">
                Add New Gender
            </button>
        </div>
        <div id="errorAdmin"></div>
        <div id="successAdmin"></div>
    </form>
</div>`

$("#adminIspis").html(forma)

$("#btnInsertGender").click(addNewGender);
}

function addNewGender(){
    var genderName=document.getElementById("genderName").value

    var dataForSend= new FormData();

    dataForSend.append("genderName",genderName)
    dataForSend.append("insert","true")

    $.ajax({
        url:"models/insertNewGender.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:dataForSend,
        success:function(data){
            console.log(data);
            $("#successAdmin").html(data);
            $("#successAdmin").addClass("alert alert-success p-3");
        },
        error(xhr){
            console.log(xhr);
            $("#errorAdmin").html(JSON.parse(xhr.responseText))
            $("#errorAdmin").addClass("alert alert-warning p-3")
        }
    });
}

function formAddMenu(){
    $.ajax({
        url:"models/getPrivileges.php",

```





```

}

function addNewMenu(){
    var menuItemName=document.getElementById("menuItemName").value
    var
menuItemPosition=document.getElementById("menuItemPosition").value
    var privilege=document.getElementById("selectPrivilege").value

    var dataForSend= new FormData();

    dataForSend.append("menuItemName",menuItemName)
    dataForSend.append("menuItemPosition",menuItemPosition)
    dataForSend.append("selectPrivilege",privilege)
    dataForSend.append("insert","true")

    $.ajax({
        url:"models/insertNewMenuItem.php",
        method:"post",
        dataType:"json",
        processData:false,
        contentType:false,
        data:dataForSend,
        success:function(data){
            console.log(data);
            $("#successAdmin").html(data);
            $("#successAdmin").addClass("alert alert-success p-3");
        },
        error(xhr){
            console.log(xhr);
            $("#errorAdmin").html(JSON.parse(xhr.responseText))
            $("#errorAdmin").addClass("alert alert-warning p-3")
        }
    });
}

//ispis i brisanje proizvoda u korpi
$(document).ready(function(){
    let productsCartLS=getLS("cart");

    if (productsCartLS==null || productsCartLS.length<1){
        $("#products").html(`<div id="greskaShop"><h1>Your shopping cart
is empty!</h1></div>`);
    }
    else{
        displayCartData();
        generateTable(productInfo);
    }
});
var productsCartLS=getLS("cart");

```

```
//funkcija za dohvatanje proizvoda za korpu
```

```
function displayCartData(){
  productInfo=getLS("sviProizvodi");
  //console.log(productInfo);
  productInfo=productInfo.filter(p=>{
    for(let product of productsCartLS){
      if(p.id == product.id){
        p.quantity=product.qty;
        return true;
      }
    }
  })
  //console.log(productInfo);
}
```

```
function generateTable(){
  let html=`
  <table class="table">
  <thead class="thead-dark">
  <tr>
    <th scope="col">Product</th>
    <th scope="col">Image</th>
    <th scope="col">Qty</th>
    <th scope="col">Price</th>
    <th scope="col">Total</th>
    <th scope="col">Remove</th>
  </tr>
  </thead>
  <tbody>`
  for(let p of productInfo){
    html+=`
      <tr>
        <th scope="row">${p.name}</th>
        <td></td>
        <td>${p.quantity}</td>
        <td>${p.price}</td>
        <td>${parseFloat(p.price)*p.quantity}</td>
        <td><button onclick="removeFromCart(${p.id})"
class="btn" value="remove">remove</button></td>
      </tr>
    `
  }
  html+=`
  </tbody>
  </table>
  `
}
```

```

        $("#products").html(html);
    }
    function removeFromCart(id){
        let korpa=getLS("cart");
        let filtered=korpa.filter(p=>p.id != id);
        setLS("cart",filtered);
        displayCartData();
        printNumberOfProducts();
        location.reload();
    }

var quantity=0;
function printNumberOfProducts(){
    let productsCart = getLS("cart");
    if(productsCart == null){
        $("#broj-proizvoda").html("(0 products)");
        // console.log("problem?");
    }
    else{
        quantity=productsCart.length;
        let numberOfProducts = quantity;
        let txt = quantity+" ";

        if(numberOfProducts == 1){
            txt = quantity+" "+"product";
        }
        else{
            txt =quantity+" products";
        }
        setLS("brojProizvodaLS",txt);
        $("#broj-proizvoda").html(`${txt}`)
    }
}

```