***Лабораторна робота №6-7***

**Тема: JSON, LocalStorage, SessionStorage, Cookie.**

**Мета роботи:** навчитись передавати та зберігати данні з сторінки.

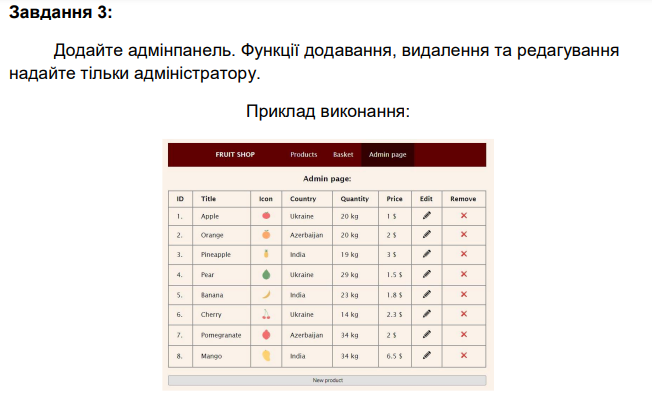
***Хід роботи:***

***Завдання на лабораторну роботу:***

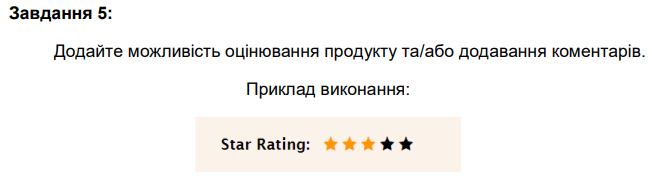
1. ***Завдання на лабораторну роботу:***

******

******

******

******

******

***Виконання:***

* ***PHP:***
  + ***task.php:***

<?php session\_start(); ?>  
  
<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.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="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.9.1/font/bootstrap-icons.css">  
  
 <title>RPRC</title>  
</head>  
<body>  
<header class="header">  
 <?php include("navbar.php") ?>  
</header>  
  
<main role="main" class="container main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12">  
 <?php  
 // $DB = new PDO("mysql:host=localhost;dbname=users\_goods;charset=utf8", "root", "");  
 // $DB->setAttribute(PDO::ATTR\_ERRMODE, PDO::ERRMODE\_EXCEPTION);  
  
 // $sqlScript = "CREATE DATABASE users\_goods;";  
 // $DB->exec($sqlScript);  
 //  
 // $sqlScript = "DROP DATABASE user;";  
 // $DB->exec($sqlScript);  
 // $sqlScript = "DROP TABLE users;";  
 // $DB->exec($sqlScript);  
 // $sqlScript = "DROP TABLE goods;";  
 // $DB->exec($sqlScript);  
 //  
 // $sqlScript = "CREATE TABLE users(  
 // id INTEGER PRIMARY KEY NOT NULL AUTO\_INCREMENT,  
 // username VARCHAR(30) NOT NULL,  
 // login VARCHAR(30) NOT NULL,  
 // password VARCHAR(30) NOT NULL  
 // );";  
 // $DB->exec($sqlScript);  
 echo "<h1 class='title fs-4 text-center mb-2'>Hello, " . $\_SESSION['login'] . "</h1>";  
 ?>  
 </div>  
  
 <?php include('sidebar.php'); ?>  
 </div>  
</main>  
  
<footer>  
 <?php include('footer.php') ?>  
</footer>  
  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
 crossorigin="anonymous"></script>  
</body>  
</html>

* + ***login.php:***

<?php  
include('users\_db\_check.php');  
session\_start();  
$login = '';  
$password = '';  
?>  
  
<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>Login</title>  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.css?v=<?php echo time(); ?>">  
  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">  
</head>  
<body>  
<?php include('navbar.php'); ?>  
  
<main class="container" role="main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12">  
 <div class="content-section">  
 <?php  
 try {  
 $DB = new PDO("mysql:host=localhost;dbname=users\_goods", "root", "");  
 $DB->setAttribute(PDO::*ATTR\_ERRMODE*, PDO::*ERRMODE\_EXCEPTION*);  
  
 $sqlScript = "USE users\_goods";  
 $DB->exec($sqlScript);  
 } catch (PDOException $exception) {  
 echo "<div class='alert alert-danger'>Connection failed: " . $exception->getMessage() . "</div>";  
 }  
 ?>  
 <form action="" method="post" enctype="multipart/form-data" class="container\_\_form">  
 <fieldset class="form-group">  
 <legend class="border-bottom mb-4">Join to our team!</legend>  
  
 <div class="form-group mb-3">  
 <label for="textLogin" id="labelLogin" class="form\_\_label\_login form-label">Login:</label>  
 <?php  
 if (isset($\_POST['textLogin'])) {  
 $login = $\_POST['textLogin'];  
 $\_SESSION["login"] = $login;  
  
 if (!is\_login\_exist($DB, $login)) {  
 if ($login != '') {  
 echo '<input type="text" name="textLogin" id="textLogin" class="form\_\_text\_login form-control form-control-lg is-invalid">';  
 echo "<div class='invalid-feedback'>Login doesn't exist!</div>";  
 }  
 } else {  
 echo '<input type="text" name="textLogin" id="textLogin" class="form\_\_text\_login form-control form-control-lg">';  
 }  
 } else {  
 echo '<input type="text" name="textLogin" id="textLogin" class="form\_\_text\_login form-control form-control-lg">';  
 }  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <label for="password" id="labelPassword" class="form\_\_label\_password form-label">Password:</label>  
  
 <?php  
 if (isset($\_POST['password'])) {  
 $password = $\_POST['password'];  
 $\_SESSION["password"] = $password;  
  
 if (is\_password\_correct($DB, $login, $password)) {  
 $\_SESSION["admin"] = get\_user\_id\_by\_login($DB, $login);  
 header("Location: task.php");  
 } else {  
 echo '<input type="password" name="password" id="password" class="form\_\_password form-control form-control-lg is-invalid">';  
 echo "<div class='invalid-feedback'>Password isn't correct!</div>";  
 }  
 } else {  
 echo '<input type="password" name="password" id="password" class="form\_\_password form-control form-control-lg">';  
 }  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <button type="submit" id="submitButton" class="form\_\_button\_submit btn btn-outline-dark">Log in</button>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 <div class="border-top pt-3 pb-3">  
 <small class="text-muted">  
 Need an account? <a href="register.php" class="ml-2">Sign up now!</a>  
 </small>  
 </div>  
 </div>  
  
 <?php include('sidebar.php'); ?>  
 </div>  
</main>  
  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
 crossorigin="anonymous"></script>  
</body>  
</html>

* + ***register.php:***

<?php  
session\_start();  
$username = '';  
$usernameCheck = false;  
$login = '';  
$loginCheck = false;  
$password = '';  
$passwordCheck = false;  
$repeatPassword = '';  
$repeatPasswordCheck = false;  
?>  
  
<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
 <title>Register</title>  
  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.css?v=<?php echo time(); ?>">  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOmLASjC" crossorigin="anonymous">  
</head>  
<body>  
<?php include('navbar.php'); ?>  
  
<main class="container" role="main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12">  
 <div class="content-section">  
 <form action="" method="post" enctype="multipart/form-data" class="container\_\_form">  
 <fieldset class="form-group">  
 <legend class="border-bottom mb-4">Join to our team!</legend>  
  
 <div class="form-group mb-3">  
 <label for="textUsername" id="labelUsername" class="form\_\_label\_username form-label">Username:</label>  
 <?php  
 if (isset($\_POST['textUsername'])) {  
 $username = $\_POST['textUsername'];  
  
 if (strlen($username) <= 3) {  
 echo "<input type='text' name='textUsername' id='textUsername' class='form\_\_text\_username form-control form-control-lg is-invalid'>";  
 echo "<div class='invalid-feedback'>Username length is less or equal than 3!</div>";  
 } else {  
 echo "<input type='text' name='textUsername' id='textUsername' class='form\_\_text\_username form-control form-control-lg'>";  
 $usernameCheck = true;  
 }  
 } else {  
 echo "<input type='text' name='textUsername' id='textUsername' class='form\_\_text\_username form-control form-control-lg'>";  
 }  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <label for="textLogin" id="labelLogin" class="form\_\_label\_login form-label">Login:</label>  
  
 <?php  
 function isCyrillic($text)  
 {  
 return preg\_match('/[А-Яа-яЁёІіЇїЄє]/u', $text);  
 }  
  
 if (isset($\_POST['textLogin'])) {  
 $login = $\_POST['textLogin'];  
  
 if (isCyrillic($login)) {  
 echo "<input type='text' name='textLogin' id='textLogin' class='form\_\_text\_login form-control form-control-lg is-invalid'>";  
 echo "<div class='invalid-feedback'>Login has cyrillic symbols!</div>";  
 } else {  
 echo "<input type='text' name='textLogin' id='textLogin' class='form\_\_text\_login form-control form-control-lg'>";  
 $loginCheck = true;  
 }  
 } else {  
 echo "<input type='text' name='textLogin' id='textLogin' class='form\_\_text\_login form-control form-control-lg'>";  
 }  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <label for="password" id="labelPassword" class="form\_\_label\_password form-label">Password:</label>  
  
 <?php  
 if (isset($\_POST['password']) && isset($\_POST['repeatPassword'])) {  
 $password = $\_POST['password'];  
 $repeatPassword = $\_POST['repeatPassword'];  
 if ($password != $repeatPassword) {  
 echo "<input type='password' name='password' id='password' class='form\_\_password form-control form-control-lg is-invalid'>";  
 echo "<div class='invalid-feedback'>Passwords don't match!</div>";  
 } else {  
 echo "<input type='password' name='password' id='password' class='form\_\_password form-control form-control-lg'>";  
 $passwordCheck = true;  
 }  
 } else {  
 echo "<input type='password' name='password' id='password' class='form\_\_password form-control form-control-lg'>";  
 }  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <label for="repeatPassword" id="labelRepeatPassword" class="form\_\_label\_repeat\_password form-label">Repeat password:</label>  
  
 <?php  
 if (isset($\_POST['password']) && isset($\_POST['repeatPassword'])) {  
 $password = $\_POST['password'];  
 $repeatPassword = $\_POST['repeatPassword'];  
 if ($password != $repeatPassword) {  
 echo "<input type='password' name='repeatPassword' id='repeatPassword' class='form\_\_password form-control form-control-lg is-invalid'>";  
 echo "<div class='invalid-feedback'>Passwords don't match!</div>";  
 } else {  
 echo "<input type='password' name='repeatPassword' id='repeatPassword' class='form\_\_password form-control form-control-lg'>";  
 $repeatPasswordCheck = true;  
 }  
 } else {  
 echo "<input type='password' name='repeatPassword' id='repeatPassword' class='form\_\_password form-control form-control-lg'>";  
 }  
  
 ?>  
 </div>  
  
 <div class="form-group mb-3">  
 <button type="submit" id="submitButton" class="form\_\_button\_submit btn btn-outline-dark">Sign in</button>  
 </div>  
 </fieldset>  
 </form>  
 </div>  
 <div class="border-top pt-3 pb-3">  
 <small class="text-muted">  
 Already have an account? <a href="login.php" class="ml-2">Sign in!</a>  
 </small>  
 </div>  
 </div>  
 <?php  
 include('sidebar.php')  
 ?>  
 </div>  
 <?php  
 if ($usernameCheck && $loginCheck && $passwordCheck && $repeatPasswordCheck) {  
 try {  
 $DB = new PDO("mysql:host=localhost;dbname=users\_goods", "root", "");  
 $DB->setAttribute(PDO::*ATTR\_ERRMODE*, PDO::*ERRMODE\_EXCEPTION*);  
  
 $sqlScript = "USE users\_goods";  
 $DB->exec($sqlScript);  
  
 include('users\_db\_check.php');  
  
 if (is\_username\_exist($DB, $username)) {  
 echo "<div class='alert alert-danger'>Username '$username' is already exist!</div>";  
 } elseif (is\_login\_exist($DB, $login)) {  
 echo "<div class='alert alert-danger'>Login '$login' is already exist!</div>";  
 } else {  
 $sqlScript = "INSERT INTO users (username, login, password) VALUES (:username, :login, :password)";  
  
 $statement = $DB->prepare($sqlScript);  
 $statement->bindValue(":username", $username);  
 $statement->bindValue(":login", $login);  
 $statement->bindValue(":password", $password);  
  
 $affectedRowsNumber = $statement->execute();  
 if ($affectedRowsNumber > 0) {  
 echo "<div class='alert alert-success'>Data successfully added! Username: $username. Login: $login. Password: $password.</div>";  
 } else {  
 echo "<div class='alert alert-success'>Data add failed!</div>";  
 }  
 }  
 } catch (PDOException $exception) {  
 echo "<div class='alert alert-danger'>Connection failed: " . $exception->getMessage() . "</div>";  
 }  
 }  
 ?>  
</main>  
  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
 crossorigin="anonymous"></script>  
</body>  
</html>

* + ***logout.php:***

<?php  
session\_start();  
unset($\_SESSION['admin']);  
unset($\_SESSION['username']);  
unset($\_SESSION['login']);  
unset($\_SESSION['password']);  
header("Location: login.php");

* + ***navbar.php:***

<?php  
session\_start();  
$is\_admin = false;  
if (isset($\_SESSION["admin"]))  
 $is\_admin = $\_SESSION["admin"];  
?>  
  
<nav class="navbar navbar-expand-lg navbar-light navbar-bg">  
 <div class="container">  
 <img src="../../../img/raccoon-logo-1.png" alt="logo" width="50" height="50" class="img-fluid">  
 <a href="task.php" class="navbar-brand pl-2 text-wrap">Raccoon's Programming Courses</a>  
 <button class="navbar-toggler" type="button" data-bs-toggle="collapse" data-bs-target="#navbarNav" aria-controls="navbarNav" aria-expanded="false"  
 aria-label="Toggle navigation">  
 <span class="navbar-toggler-icon"></span>  
 </button>  
 <div class="collapse navbar-collapse" id="navbarNav">  
 <ul class="navbar-nav me-auto mb-2 mb-lg-0">  
 <li class='nav-item'>  
 <a class='nav-link' href='products.php'>Products</a>  
 </li>  
 <?php  
 if ($is\_admin) {  
 echo "  
 <li class='nav-item'>  
 <a class='nav-link' href='admin\_page.php'>Admin page</a>  
 </li>  
 ";  
 }  
 ?>  
 </ul>  
 <ul class="navbar-nav">  
 <?php  
 if ($is\_admin) {  
 echo "  
 <li class='nav-item'>  
 <a class='nav-link' href='basket.php'>Basket</a>  
 </li>  
 ";  
 echo "  
 <li class='nav-item'>  
 <a class='nav-link' href='logout.php'>Logout</a>  
 </li>  
 ";  
 } else {  
 echo "  
 <li class='nav-item'>   
 <a class='nav-link' href='login.php'>Login</a>  
 </li>  
 <li class='nav-item'>  
 <a class='nav-link' href='register.php'>Register</a>  
 </li>  
 ";  
 }  
 ?>  
 </ul>  
 </div>  
 </div>  
</nav>

* + ***sidebar.php:***

<?php  
echo "<div class='col-xl-4 col-lg-4 col-md-12 col-sm-12'>";  
echo "<div class='content-section'>";  
echo "<h3>Our Sidebar</h3>";  
echo "<p class='text-muted' > You can put any information here you'd like.</p>";  
echo "<ul class='list-group'>";  
echo "<li class='list-group-item list-group-item-light'>Latest Posts</li>";  
echo "<li class='list-group-item list-group-item-light'>Announcements</li>";  
echo "<li class='list-group-item list-group-item-light'>Calendars</li>";  
echo "<li class='list-group-item list-group-item-light'>etc</li>";  
echo "</ul>";  
echo "</div>";  
echo "</div>";

* + ***footer.php:***

<footer class="bg-light text-center text-white">  
 <!-- Grid container -->  
 <div class="container p-4 pb-0">  
 <!-- Section: Social media -->  
 <section class="mb-4">  
 <!-- Facebook -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #3b5998;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-facebook"></i>  
 </a>  
  
 <!-- Twitter -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #55acee;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-twitter"></i>  
 </a>  
  
 <!-- Google -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #dd4b39;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-google"></i>  
 </a>  
  
 <!-- Instagram -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #ac2bac;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-instagram"></i>  
 </a>  
  
 <!-- Linkedin -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #0082ca;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-linkedin"></i>  
 </a>  
 <!-- Github -->  
 <a  
 class="btn text-white btn-floating m-1"  
 style="background-color: #333333;"  
 href="#!"  
 role="button"  
 >  
 <i class="bi bi-github"></i>  
 </a>  
 </section>  
 <!-- Section: Social media -->  
 </div>  
 <!-- Grid container -->  
  
 <!-- Copyright -->  
 <div class="text-center p-3" style="background-color: rgba(0, 0, 0, 0.2);">  
 © 2022 Copyright:  
 <a class="text-white" href="https://github.com/AndriiBabushko/Frontend">Andrii Babushko</a>  
 </div>  
 <!-- Copyright -->  
</footer>

* + ***products.php:***

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.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="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.3.0/font/bootstrap-icons.css">  
 <script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
 crossorigin="anonymous" defer></script>  
  
 <title>RPRC - Products</title>  
</head>  
<body>  
<header class="header">  
 <?php include("navbar.php") ?>  
</header>  
  
<main role="main" class="container main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12">  
 <h1 class="title fs-4 text-center mb-2 pb-2 border-bottom" id="productsTitle">All products:</h1>  
 <div class="mb-3 text-start fs-5 fw-normal border-bottom">  
 <label class="form-label" for="sortBy">Sort by:</label>  
 <select class="form-select form-select-lg" name="sortBy" id="sortBy">  
 <option selected>Choose one</option>  
 <option value="title">By Title</option>  
 <option value="price">By Price</option>  
 <option value="time">By Time</option>  
 </select>  
 </div>  
 <div class="mb-3 pb-2 text-start fs-5 fw-normal border-bottom">  
 <label class="form-label" for="">Sorting type:</label>  
 <div class="form-check">  
 <input class="form-check-input" type="radio" name="sortingType" id="ascendingSort" value="ascendingSort" checked>  
 <label class="form-check-label" for="ascendingSort">Ascending sort</label>  
 </div>  
 <div class="form-check">  
 <input class="form-check-input" type="radio" name="sortingType" id="descendingSort" value="descendingSort">  
 <label class="form-check-label" for="descendingSort">Descending sort</label>  
 </div>  
 </div>  
 </div>  
 <?php include('sidebar.php'); ?>  
 </div>  
</main>  
  
<footer class="footer">  
 <?php include("footer.php") ?>  
</footer>  
  
<script type="module" defer>  
 <?php require('../js/localStorageData.js'); ?>  
 <?php require('../js/products.js'); ?>  
</script>  
</body>  
</html>

* + ***admin\_page.php:***

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.css">  
 <link href="https://cdn.jsdelivr.net/npm/bootstrap@5.2.3/dist/css/bootstrap.min.css" rel="stylesheet"  
 integrity="sha384-rbsA2VBKQhggwzxH7pPCaAqO46MgnOM80zW1RWuH61DGLwZJEdK2Kadq2F9CUG65" crossorigin="anonymous">  
 <link rel="stylesheet" href="https://cdn.jsdelivr.net/npm/bootstrap-icons@1.3.0/font/bootstrap-icons.css">  
  
 <title>RPRC - Admin Page</title>  
</head>  
<body>  
<header class="header">  
 <?php include("navbar.php") ?>  
</header>  
  
<main role="main" class="container main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12" id="tableBlock">  
 <h1 class="title fs-4 text-center mb-2" id="adminTitle">Admin page</h1>  
 </div>  
  
 <?php include('sidebar.php'); ?>  
 </div>  
</main>  
  
<footer class="footer">  
 <?php include("footer.php") ?>  
</footer>  
  
<script type="module" defer>  
 <?php require('../js/localStorageData.js'); ?>  
 <?php require('../js/admin\_page.js'); ?>  
</script>  
<script src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js" integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"  
 crossorigin="anonymous">  
</script>  
</body>

* + ***basket.php:***

<!doctype html>  
<html lang="en">  
<head>  
 <meta charset="UTF-8">  
 <meta name="viewport" content="width=device-width, initial-scale=1.0, minimum-scale=1.0">  
 <meta http-equiv="X-UA-Compatible" content="ie=edge">  
  
 <link rel="icon" href="../../../../img/icons/racoon\_animal\_icon\_123571.ico">  
 <link rel="stylesheet" href="../styles/css/style.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">  
  
 <title>RPRC - Basket</title>  
</head>  
<body>  
<header class="header">  
 <?php include("navbar.php") ?>  
</header>  
  
<main role="main" class="container main">  
 <div class="row mt-3">  
 <div class="col-xl-8 col-lg-8 col-md-12 col-sm-12">  
 <h1 class="title fs-4 text-center mb-2" id="basketTitle">Your basket:</h1>  
 </div>  
  
 <?php include('sidebar.php'); ?>  
 </div>  
</main>  
  
<footer class="footer">  
 <?php include("footer.php") ?>  
</footer>  
  
<script type="module" defer>  
 <?php require('../js/localStorageData.js'); ?>  
 <?php require('../js/basket.js'); ?>  
</script>  
</body>

* + ***users\_db\_check.php:***

<?php  
function is\_username\_exist($DB, $username): bool  
{  
 $sqlScript = "SELECT username FROM users";  
 $DB\_usernames = $DB->query($sqlScript);  
  
 while ($row = $DB\_usernames->fetch()) {  
 $DB\_username = $row["username"];  
 if ($DB\_username == $username)  
 return true;  
 }  
  
 return false;  
}  
  
function is\_login\_exist($DB, $login): bool  
{  
 $sqlScript = "SELECT login FROM users";  
 $DB\_logins = $DB->query($sqlScript);  
  
 while ($row = $DB\_logins->fetch()) {  
 $DB\_login = $row["login"];  
 if ($DB\_login == $login)  
 return true;  
 }  
  
 return false;  
}  
  
function is\_password\_correct($DB, $login, $password): bool  
{  
 $sqlScript = "SELECT password FROM users WHERE login = :login";  
  
 $statement = $DB->prepare($sqlScript);  
 $statement->bindvalue(":login", $login);  
 $statement->execute();  
  
 foreach ($statement as $row) {  
 $DB\_password = $row["password"];  
 if ($DB\_password == $password)  
 return true;  
 }  
  
 return false;  
}  
  
function get\_user\_id\_by\_login($DB, $login): int  
{  
 $sqlScript = "SELECT \* FROM users WHERE login = :login";  
  
 $statement = $DB->prepare($sqlScript);  
 $statement->bindvalue(":login", $login);  
 $statement->execute();  
 $row = $statement->fetch();  
  
 return $row["id"];  
}

* ***SASS:***

\*  
 box-sizing: border-box  
 padding: 0  
 margin: 0

* ***JS:***
  + ***product.js:***

'use strict';  
  
const showProductItems = (divItems) => {  
 const productsTitle = document.querySelector('#productsTitle');  
 productsTitle.parentElement.appendChild(divItems);  
};  
  
const createProductItems = () => {  
 const programmingCourses = readProgrammingCoursesData();  
 const divItemsContainer = document.createElement('div');  
 divItemsContainer.setAttribute('id', 'itemsContainer');  
 divItemsContainer.setAttribute('class', 'd-flex flex-row flex-wrap justify-content-center justify-content-lg-center justify-content-md-between align-items-center"');  
 for (let i = 0; i < programmingCourses.length; i++) {  
 const divItemBlock = document.createElement('div');  
 divItemBlock.setAttribute('id', 'itemBlock');  
 divItemBlock.setAttribute('class', 'card m-1 text-center');  
 divItemBlock.style.width = '200px';  
  
 // creating card header  
 const divItemHeader = document.createElement('div');  
 divItemHeader.setAttribute('class', 'card-header border-bottom');  
 divItemHeader.style.backgroundColor = 'lightgrey';  
 const cardTitle = document.createElement('h5');  
 cardTitle.setAttribute('class', 'card-title');  
 cardTitle.innerHTML = programmingCourses[i].title;  
 divItemHeader.appendChild(cardTitle);  
  
 // creating card body  
 const divItemBody = document.createElement('div');  
 divItemBody.setAttribute('class', 'card-body border-bottom');  
 divItemBody.style.backgroundColor = '#f1f1f1';  
 const cardPriceSubtitle = document.createElement('p');  
 cardPriceSubtitle.setAttribute('class', 'card-subtitle fs-5');  
 cardPriceSubtitle.innerHTML = 'Price';  
 const cardPriceText = document.createElement('p');  
 cardPriceText.setAttribute('class', 'card-text fw-normal fs-6');  
 cardPriceText.innerHTML = programmingCourses[i].price + ' UAH';  
 const cardTimeSubtitle = document.createElement('p');  
 cardTimeSubtitle.setAttribute('class', 'card-subtitle fs-5');  
 cardTimeSubtitle.innerHTML = 'Time to learn';  
 const cardTimeText = document.createElement('p');  
 cardTimeText.setAttribute('class', 'card-text fw-normal fs-6');  
 cardTimeText.innerHTML = programmingCourses[i].time;  
 divItemBody.appendChild(cardPriceSubtitle);  
 divItemBody.appendChild(cardPriceText);  
 divItemBody.appendChild(cardTimeSubtitle);  
 divItemBody.appendChild(cardTimeText);  
  
 // creating card footer  
 const divItemFooter = document.createElement('div');  
 divItemFooter.setAttribute('class', 'card-footer');  
 divItemFooter.style.backgroundColor = 'darkgrey';  
 const cardBasketButton = document.createElement('button');  
 cardBasketButton.setAttribute('type', 'button');  
 cardBasketButton.setAttribute('id', 'basketButton');  
 cardBasketButton.setAttribute('class', 'btn btn-outline-light my-2');  
 cardBasketButton.style.width = '100%';  
 cardBasketButton.innerHTML = 'Add To Busket';  
  
 // event to add product to basket  
 cardBasketButton.addEventListener('click', (event) => {  
 const basketData = readBasketData();  
 let newBasketData = {};  
  
 if (basketData.length > 0) {  
 newBasketData = {  
 'id': basketData.length + 1,  
 'title': programmingCourses[i].title,  
 'price': programmingCourses[i].price  
 }  
 } else {  
 newBasketData = {  
 'id': 1,  
 'title': programmingCourses[i].title,  
 'price': programmingCourses[i].price  
 }  
 }  
  
 addBasketData(newBasketData);  
  
 cardBasketButton.setAttribute('class', 'btn btn-outline-success my-2');  
 cardBasketButton.innerHTML = 'Successfully added!';  
 });  
 divItemFooter.appendChild(cardBasketButton);  
  
 // creating comments modal button  
 const divCommentButton = document.createElement('div');  
 divCommentButton.setAttribute('class', 'btn-group');  
 divCommentButton.style.width = '100%';  
 const cardCommentsButton = document.createElement('button');  
 cardCommentsButton.setAttribute('type', 'button');  
 cardCommentsButton.setAttribute('id', 'commentsButton');  
 cardCommentsButton.setAttribute('class', 'btn btn-outline-light my-2');  
 cardCommentsButton.setAttribute('data-bs-toggle', 'modal');  
 cardCommentsButton.setAttribute('data-bs-target', `#staticBackdrop${i}`);  
 cardCommentsButton.style.width = '100%';  
 cardCommentsButton.innerHTML = 'Show comments';  
  
 // modal header  
 const titleModalHeader = document.createElement('h1');  
 titleModalHeader.setAttribute('class', 'modal-title fs-5');  
 titleModalHeader.setAttribute('id', 'staticBackdropLabel');  
 titleModalHeader.innerHTML = 'Comments';  
  
 const buttonModalHeader = document.createElement('button');  
 buttonModalHeader.setAttribute('type', 'button');  
 buttonModalHeader.setAttribute('class', 'btn-close');  
 buttonModalHeader.setAttribute('data-bs-dismiss', 'modal');  
 buttonModalHeader.setAttribute('aria-label', 'Close');  
  
 const divModalHeader = document.createElement('div');  
 divModalHeader.setAttribute('class', 'modal-header');  
 divModalHeader.appendChild(titleModalHeader);  
 divModalHeader.appendChild(buttonModalHeader);  
  
 // create comment Button  
 const createCommentButton = document.createElement('button');  
 createCommentButton.setAttribute('type', 'button');  
 createCommentButton.setAttribute('class', 'btn btn-outline-success my-2');  
 createCommentButton.setAttribute('id', 'createCommentButton');  
 createCommentButton.style.width = '100%';  
 createCommentButton.innerHTML = 'Create comment';  
 // add event to create comment button  
 let divForm = document.createElement('div');  
 createCommentButton.addEventListener('click', () => {  
 for (let j = 0; j < 2; j++) {  
 divForm = createCommentForm(divForm, j, i);  
 }  
  
 createCommentButton.parentElement.appendChild(divForm);  
 createCommentButton.remove();  
 });  
  
 // modal body  
 const divModalBody = document.createElement('div');  
 divModalBody.setAttribute('class', 'modal-body');  
 divModalBody.appendChild(createCommentButton);  
  
 createModalCommentCards(divModalBody, createCommentButton, i);  
  
 // modal footer  
 const divModalFooterCloseButton = document.createElement('button');  
 divModalFooterCloseButton.setAttribute('type', 'button');  
 divModalFooterCloseButton.setAttribute('class', 'btn btn-secondary');  
 divModalFooterCloseButton.setAttribute('data-bs-dismiss', 'modal');  
 divModalFooterCloseButton.innerHTML = 'Close';  
  
 const divModalFooter = document.createElement('div');  
 divModalFooter.setAttribute('class', 'modal-footer');  
 divModalFooter.appendChild(divModalFooterCloseButton);  
  
 // modal content  
 const divModalContent = document.createElement('div');  
 divModalContent.setAttribute('class', 'modal-content');  
 divModalContent.appendChild(divModalHeader);  
 divModalContent.appendChild(divModalBody);  
 divModalContent.appendChild(divModalFooter);  
  
 const divModalDialog = document.createElement('div');  
 divModalDialog.setAttribute('class', 'modal-dialog modal-dialog-scrollable');  
 divModalDialog.appendChild(divModalContent);  
  
 const divModalFade = document.createElement('div');  
 divModalFade.setAttribute('class', 'modal fade');  
 divModalFade.setAttribute('id', `staticBackdrop${i}`);  
 divModalFade.setAttribute('data-bs-backdrop', 'static');  
 divModalFade.setAttribute('data-bs-keyboard', 'false');  
 divModalFade.setAttribute('tabindex', '-1');  
 divModalFade.setAttribute('aria-labelledby', 'staticBackdropLabel');  
 divModalFade.setAttribute('aria-hidden', 'true');  
 divModalFade.appendChild(divModalDialog);  
  
 divItemFooter.appendChild(cardCommentsButton);  
 divItemFooter.appendChild(divModalFade);  
  
 // add all blocks to one item  
 divItemBlock.appendChild(divItemHeader);  
 divItemBlock.appendChild(divItemBody);  
 divItemBlock.appendChild(divItemFooter);  
 divItemsContainer.appendChild(divItemBlock);  
 }  
  
 return divItemsContainer;  
};  
  
function createCommentForm(divForm, j, i) {  
 divForm.setAttribute('id', 'createCommentForm');  
  
 const divFormBlock = document.createElement('div');  
 divFormBlock.setAttribute('class', 'mb-3 text-start fs-5 fw-normal');  
  
 switch (j) {  
 case 0: {  
 const label = document.createElement('label');  
 label.setAttribute('class', 'form-label');  
 label.innerHTML = 'Input comment here:';  
 label.setAttribute('for', 'commentText');  
  
 const textArea = document.createElement('textarea');  
 textArea.setAttribute('class', 'form-control form-control-lg');  
 textArea.setAttribute('rows', '3');  
 textArea.setAttribute('id', 'commentText');  
 textArea.setAttribute('name', 'commentText');  
  
 divFormBlock.appendChild(label);  
 divFormBlock.appendChild(textArea);  
 break;  
 }  
 case 1: {  
 const button = document.createElement('button');  
  
 button.setAttribute('class', 'btn btn-outline-success mb-2');  
 button.setAttribute('type', 'button');  
 button.setAttribute('id', 'buttonOK');  
 button.setAttribute('name', 'buttonOK');  
 button.style.width = '100%';  
 button.innerHTML = 'OK'  
  
 button.addEventListener('click', () => {  
 const commentText = document.querySelector('#commentText').value;  
 const commentsData = readCommentsData(i);  
 const userData = 'admin';  
 const date = new Date();  
 const current\_date = date.getDate() + "." + (date.getMonth() + 1) + "." + date.getFullYear();  
 const current\_time = date.getHours() + ":" + date.getMinutes() + ":" + date.getSeconds();  
 const timePost = current\_date + " " + current\_time;  
 let commentData = {};  
  
 if (commentsData.length > 0) {  
 let id = commentsData[commentsData.length - 1].id;  
 id++;  
 commentData = {  
 'id': id,  
 'username': userData,  
 'time': timePost,  
 'commentText': commentText  
 };  
 } else {  
 commentData = {  
 'id': 1,  
 'username': userData,  
 'time': timePost,  
 'commentText': commentText  
 };  
 }  
  
 addCommentsData(commentData, i);  
 location.reload();  
 });  
  
 divFormBlock.appendChild(button);  
 break;  
 }  
 }  
  
 divForm.appendChild(divFormBlock);  
  
 return divForm;  
}  
  
function createModalCommentCards(divModalBody, createCommentButton, i) {  
 const commentsData = readCommentsData(i);  
 if (commentsData.length > 0) {  
 for (let j = 0; j < commentsData.length; j++) {  
 // creating comment card header  
 const commentCardUsername = document.createElement('h5');  
 commentCardUsername.setAttribute('class', 'card-title');  
 commentCardUsername.innerHTML = commentsData[j].username;  
  
 const commentCardTime = document.createElement('p');  
 commentCardTime.setAttribute('class', 'card-subtitle fs-6');  
 commentCardTime.innerHTML = commentsData[j].time;  
  
 const divCommentCardHeader = document.createElement('div');  
 divCommentCardHeader.setAttribute('class', 'card-header border-bottom');  
 divCommentCardHeader.style.backgroundColor = 'lightgrey';  
 divCommentCardHeader.appendChild(commentCardUsername);  
 divCommentCardHeader.appendChild(commentCardTime);  
  
 // creating comment card body  
 const commentCardComment = document.createElement('p');  
 commentCardComment.setAttribute('class', 'card-text fw-normal fs-6');  
 commentCardComment.innerHTML = commentsData[j].commentText;  
  
 const divCommentCardBody = document.createElement('div');  
 divCommentCardBody.setAttribute('class', 'card-body border-bottom');  
 divCommentCardBody.style.backgroundColor = '#f1f1f1';  
 divCommentCardBody.appendChild(commentCardComment);  
  
 // creating card footer  
 const deleteCommentButton = document.createElement('button');  
 deleteCommentButton.setAttribute('type', 'button');  
 deleteCommentButton.setAttribute('class', 'btn btn-outline-danger my-2');  
 deleteCommentButton.style.width = '100%';  
 deleteCommentButton.innerHTML = 'Delete Comment';  
  
 deleteCommentButton.addEventListener('click', () => {  
 deleteCommentsDataById(i, commentsData[j].id);  
 deleteModalCommentCards(deleteCommentButton.parentNode.parentNode.parentNode);  
 createModalCommentCards(divModalBody, createCommentButton, i);  
 });  
  
 const divCommentCardFooter = document.createElement('div');  
 divCommentCardFooter.setAttribute('class', 'card-footer');  
 divCommentCardFooter.style.backgroundColor = 'lightgrey';  
 divCommentCardFooter.appendChild(deleteCommentButton);  
  
 const divCommentCard = document.createElement('div');  
 divCommentCard.setAttribute('id', 'commentCard');  
 divCommentCard.setAttribute('class', 'card m-1');  
 divCommentCard.appendChild(divCommentCardHeader);  
 divCommentCard.appendChild(divCommentCardBody);  
 divCommentCard.appendChild(divCommentCardFooter);  
  
 divModalBody.firstChild.before(divCommentCard);  
 }  
 } else {  
 const warningNoCommentsBlock = document.createElement('div');  
 warningNoCommentsBlock.setAttribute('class', 'alert alert-secondary');  
 warningNoCommentsBlock.innerHTML = 'No comments yet here!';  
 createCommentButton.before(warningNoCommentsBlock);  
 }  
}  
  
function deleteModalCommentCards(commentCard) {  
 const divCommentCards = commentCard.querySelectorAll('#commentCard');  
 for (let i = 0; i < divCommentCards.length; i++) {  
 divCommentCards[i].remove();  
 }  
}  
  
function deleteComments() {  
 const divItemsContainer = document.querySelector('#itemsContainer');  
 divItemsContainer.remove();  
}  
  
function sortingProgrammingCourses(sortingBy, sortingType) {  
 const programmingCourses = readProgrammingCoursesData();  
  
 if (sortingBy === 'title' && sortingType === 'ascendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.title.length > obj2.title.length) ? 1 : (obj1.title.length < obj2.title.length) ? -1 : 0);  
 }  
  
 if (sortingBy === 'title' && sortingType === 'descendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.title.length < obj2.title.length) ? 1 : (obj1.title.length > obj2.title.length) ? -1 : 0);  
 }  
  
 if (sortingBy === 'price' && sortingType === 'ascendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.price > obj2.price) ? 1 : (obj1.price < obj2.price) ? -1 : 0);  
 }  
  
 if (sortingBy === 'price' && sortingType === 'descendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.price < obj2.price) ? 1 : (obj1.price > obj2.price) ? -1 : 0);  
 }  
  
 if (sortingBy === 'time' && sortingType === 'ascendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.time.length > obj2.time.length) ? 1 : (obj1.time.length < obj2.time.length) ? -1 : 0);  
 }  
  
 if (sortingBy === 'time' && sortingType === 'descendingSort') {  
 programmingCourses.sort((obj1, obj2) => (obj1.time.length < obj2.time.length) ? 1 : (obj1.time.length > obj2.time.length) ? -1 : 0);  
 }  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
 deleteComments();  
 showProductItems(createProductItems());  
}  
  
showProductItems(createProductItems());  
  
const sortingSelect = document.querySelector("#sortBy");  
sortingSelect.addEventListener('change', (event) => {  
 const sortingBy = sortingSelect.value;  
 const sortingType = document.querySelector('input[name="sortingType"]:checked').value;  
  
 sortingProgrammingCourses(sortingBy, sortingType);  
});  
  
const sortingRadios = document.querySelectorAll('input[name="sortingType"]');  
for (let i = 0; i < sortingRadios.length; i++) {  
 sortingRadios[i].addEventListener('change', (event) => {  
 const sortingBy = sortingSelect.value;  
 const sortingType = document.querySelector('input[name="sortingType"]:checked').value;  
  
 sortingProgrammingCourses(sortingBy, sortingType);  
 });  
}

* + ***admin\_page.js:***

const showTable = () => {  
 const programmingCourses = readProgrammingCoursesData();  
 addEventToCreateOKButtons(programmingCourses);  
 addEventsToEditButtons(programmingCourses);  
 addEventsToRemoveButtons(programmingCourses);  
};  
  
const createTable = (programmingCourses) => {  
 const table = document.createElement('table');  
 table.setAttribute('class', 'table fs-5 fw-normal');  
 table.setAttribute('id', 'languagesTable');  
 const thead = document.createElement('thead');  
 thead.setAttribute('class', 'table-dark')  
 const tbody = document.createElement('tbody');  
  
 // create table header  
 const thead\_tr = document.createElement('tr');  
 for (let i = 0; i < 5; i++) {  
 const thead\_th = document.createElement('th');  
 thead\_th.setAttribute('scope', 'col');  
  
 switch (i) {  
 case 0: {  
 thead\_th.innerHTML = 'ID';  
 break;  
 }  
 case 1: {  
 thead\_th.innerHTML = 'Title';  
 break;  
 }  
 case 2: {  
 thead\_th.innerHTML = 'Price';  
 break;  
 }  
 case 3: {  
 thead\_th.innerHTML = 'Time to learn';  
 break;  
 }  
 case 4: {  
 thead\_th.setAttribute('colspan', '2');  
 thead\_th.innerHTML = 'Buttons';  
 break;  
 }  
 }  
  
 thead\_tr.appendChild(thead\_th);  
 }  
 thead.appendChild(thead\_tr);  
  
 // create table body  
 for (let i = 0; i < programmingCourses.length; i++) {  
 const tbody\_tr = document.createElement('tr');  
 const values = Object.values(programmingCourses[i]).filter(elem => typeof elem !== 'object');  
 for (let j = 0; j < values.length; j++) {  
 if (j === 0) {  
 const tbody\_th = document.createElement('th');  
 tbody\_th.setAttribute('scope', 'row');  
 tbody\_th.innerHTML = values[j].toString();  
 tbody\_tr.appendChild(tbody\_th);  
 } else {  
 const tbody\_td = document.createElement('td');  
 tbody\_td.innerHTML = values[j].toString();  
 tbody\_tr.appendChild(tbody\_td);  
 }  
 }  
 const tbody\_td\_edit\_button = document.createElement('td');  
 const editButton = document.createElement('button');  
  
 editButton.setAttribute('type', 'button');  
 editButton.setAttribute('class', 'btn btn-outline-primary p-2');  
 editButton.setAttribute('id', 'buttonEdit');  
 editButton.setAttribute('value', values[0].toString());  
  
 editButton.innerHTML = 'Edit';  
 tbody\_td\_edit\_button.appendChild(editButton);  
  
 const tbody\_td\_remove\_button = document.createElement('td');  
 const removeButton = document.createElement('button');  
  
 removeButton.setAttribute('type', 'button');  
 removeButton.setAttribute('class', 'btn btn-outline-danger p-2');  
 removeButton.setAttribute('id', 'buttonRemove');  
 removeButton.setAttribute('value', values[0].toString());  
  
 removeButton.innerHTML = 'Remove';  
 tbody\_td\_remove\_button.appendChild(removeButton);  
  
 tbody\_tr.appendChild(tbody\_td\_edit\_button);  
 tbody\_tr.appendChild(tbody\_td\_remove\_button);  
 tbody.appendChild(tbody\_tr);  
 }  
  
 const createButton = document.createElement('button');  
  
 createButton.setAttribute('type', 'button');  
 createButton.setAttribute('class', 'btn btn-outline-success mb-2');  
 createButton.setAttribute('id', 'buttonCreate');  
  
 createButton.style.width = '100%';  
 createButton.innerHTML = 'Create';  
  
 table.appendChild(thead);  
 table.appendChild(tbody);  
  
 const adminPageTitle = document.querySelector('#adminTitle');  
 adminPageTitle.parentElement.appendChild(table);  
 adminPageTitle.parentElement.appendChild(createButton);  
  
 return table;  
};  
  
const deleteTable = () => {  
 const table = document.querySelector('#languagesTable');  
 table.remove();  
};  
  
function addEventToCreateOKButtons(programmingCourses) {  
 const createButton = document.querySelector('#buttonCreate');  
  
 if (document.body.contains(createButton)) {  
 createButton.addEventListener('click', (event) => {  
 let divForm = document.createElement('div');  
 for (let i = 0; i < 3; i++) {  
 divForm = createForm(divForm, i);  
 }  
  
 createButton.parentElement.appendChild(divForm);  
  
 const buttonOK = document.createElement('button');  
  
 buttonOK.setAttribute('type', 'button');  
 buttonOK.setAttribute('class', `btn btn-outline-success mb-2`);  
 buttonOK.setAttribute('id', 'buttonOK');  
  
 buttonOK.style.width = '100%';  
 buttonOK.innerHTML = 'OK';  
  
 createButton.parentElement.appendChild(buttonOK);  
 createButton.remove();  
  
 buttonOK.addEventListener('click', (event) => {  
 const titleValue = document.querySelector('#title').value;  
 const priceValue = document.querySelector('#price').value;  
 const timeToLearnValue = document.querySelector('#timeToLearn').value;  
  
 let programmingCourse = {};  
 if (programmingCourses.length > 0) {  
 let id = programmingCourses[programmingCourses.length - 1].id;  
 id++;  
 programmingCourse = {  
 'id': id,  
 'title': titleValue,  
 'price': priceValue,  
 'time': timeToLearnValue,  
 'comments': []  
 };  
 } else {  
 programmingCourse = {  
 'id': 1,  
 'title': titleValue,  
 'price': priceValue,  
 'time': timeToLearnValue,  
 'comments': []  
 };  
 }  
  
 addProgrammingCoursesData(programmingCourse);  
 deleteTable();  
 programmingCourses = readProgrammingCoursesData();  
 showTable(createTable(programmingCourses));  
  
 document.querySelector('#createForm').remove();  
 buttonOK.remove();  
 });  
 });  
 }  
}  
  
function addEventsToEditButtons(programmingCourses) {  
 const editButtons = document.querySelectorAll('#buttonEdit');  
  
 for (let i = 0; i < editButtons.length; i++) {  
 editButtons[i].addEventListener('click', (event) => {  
 let divForm = document.createElement('div');  
 for (let i = 0; i < 3; i++) {  
 divForm = createForm(divForm, i);  
 }  
  
 const createButton = document.querySelector('#buttonCreate');  
  
 createButton.parentElement.appendChild(divForm);  
  
 const buttonEdit = document.createElement('button');  
  
 buttonEdit.setAttribute('type', 'button');  
 buttonEdit.setAttribute('class', `btn btn-outline-primary mb-2`);  
 buttonEdit.setAttribute('id', 'buttonEdit');  
  
 buttonEdit.style.width = '100%';  
 buttonEdit.innerHTML = 'EDIT';  
  
 createButton.parentElement.appendChild(buttonEdit);  
 createButton.remove();  
  
 buttonEdit.addEventListener('click', () => {  
 const titleValue = document.querySelector('#title').value;  
 const priceValue = document.querySelector('#price').value;  
 const timeToLearnValue = document.querySelector('#timeToLearn').value;  
  
 const id = event.target.getAttribute('value');  
 const programmingCourse = {  
 'title': titleValue,  
 'price': priceValue,  
 'time': timeToLearnValue,  
 'comments': []  
 };  
  
 updateProgrammingCoursesDataById(programmingCourse, id);  
 deleteTable();  
 programmingCourses = readProgrammingCoursesData();  
 showTable(createTable(programmingCourses));  
  
 document.querySelector('#createForm').remove();  
 buttonEdit.remove();  
 });  
 });  
 }  
}  
  
function addEventsToRemoveButtons(programmingCourses) {  
 const removeButtons = document.querySelectorAll('#buttonRemove');  
  
 for (let i = 0; i < removeButtons.length; i++) {  
 removeButtons[i].addEventListener('click', (event) => {  
 const target = event.target;  
 const id = parseInt(target.getAttribute('value'));  
  
 deleteProgrammingCoursesDataById(id);  
 deleteTable();  
 programmingCourses = readProgrammingCoursesData();  
 showTable(createTable(programmingCourses));  
  
 document.querySelector('#buttonCreate').remove();  
 addEventToCreateOKButtons(programmingCourses);  
 });  
 }  
}  
  
function createForm(divForm, i) {  
 const divFormBlock = document.createElement('div');  
 const label = document.createElement('label');  
 const input = document.createElement('input');  
  
 divForm.setAttribute('id', 'createForm');  
 divFormBlock.setAttribute('class', 'mb-3 text-start fs-5 fw-normal');  
 label.setAttribute('class', 'form-label');  
 input.setAttribute('class', 'form-control form-control-lg');  
  
 switch (i) {  
 case 0: {  
 label.innerHTML = 'Title:';  
 label.setAttribute('for', 'title');  
  
 input.setAttribute('type', 'text');  
 input.setAttribute('id', 'title');  
 input.setAttribute('name', 'title');  
 break;  
 }  
 case 1: {  
 label.innerHTML = 'Price:';  
 label.setAttribute('for', 'price');  
  
 input.setAttribute('type', 'number');  
 input.setAttribute('id', 'price');  
 input.setAttribute('name', 'price');  
 break;  
 }  
 case 2: {  
 label.innerHTML = 'Time to learn:';  
 label.setAttribute('for', 'timeToLearn');  
  
 input.setAttribute('type', 'text');  
 input.setAttribute('id', 'timeToLearn');  
 input.setAttribute('name', 'timeToLearn');  
 break;  
 }  
 }  
  
 divFormBlock.appendChild(label);  
 divFormBlock.appendChild(input);  
 divForm.appendChild(divFormBlock);  
  
 return divForm;  
}  
  
ascendingProgrammingCoursesDataSort();  
let programmingCourses = readProgrammingCoursesData();  
  
  
createTable(programmingCourses)  
showTable();

* + ***basket.js:***

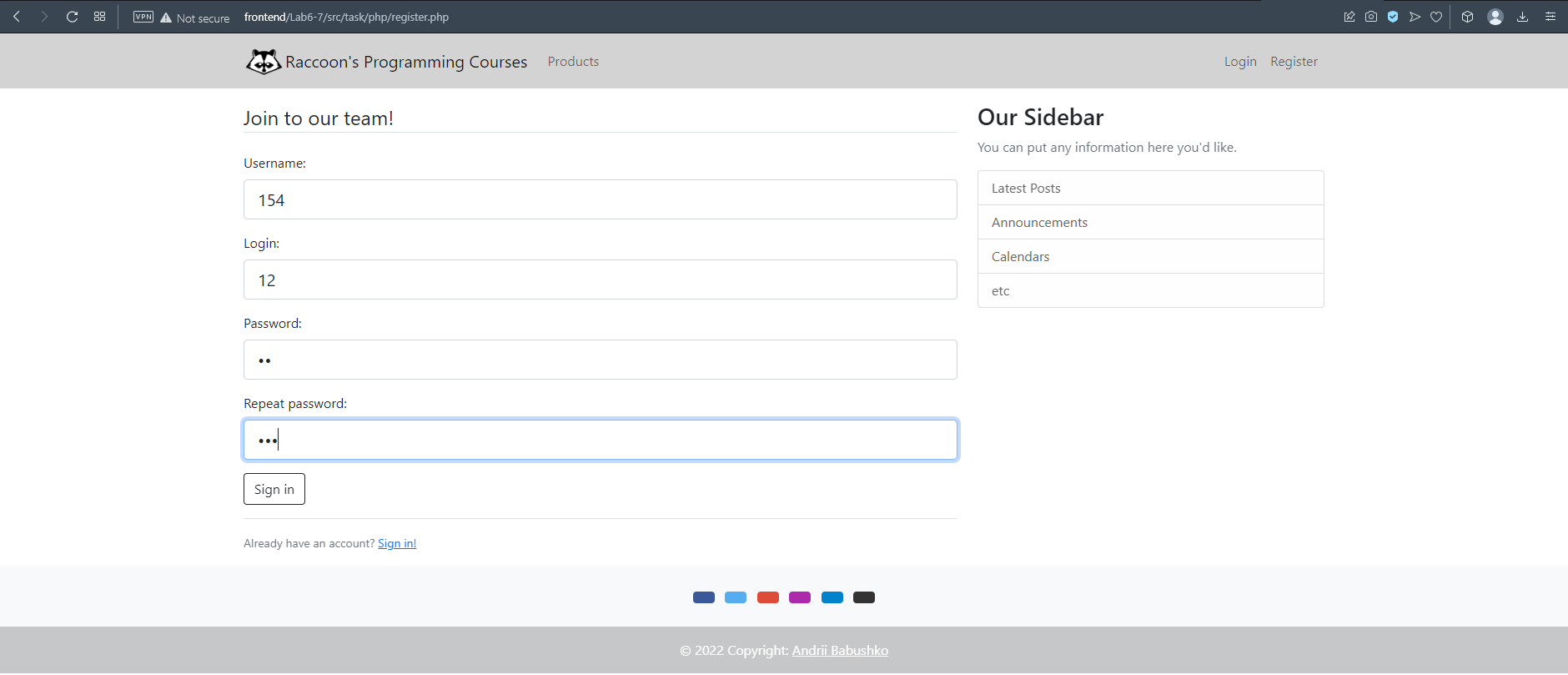
'use strict';  
  
const showBasketTable = (basketTable) => {  
 const basketTitle = document.querySelector('#basketTitle');  
 basketTitle.parentElement.appendChild(basketTable);  
 addEventsToRemoveButtons();  
};  
  
const createBasketTable = () => {  
 const basketData = readBasketData();  
 if (basketData.length > 0) {  
 const alertNoProduct = document.querySelector('#alertNoProduct');  
 if (document.body.contains(alertNoProduct))  
 alertNoProduct.remove();  
 const table = document.createElement('table');  
 table.setAttribute('class', 'table fs-5 fw-normal');  
 table.setAttribute('id', 'basketTable');  
 const thead = document.createElement('thead');  
 thead.setAttribute('class', 'table-dark')  
 const tbody = document.createElement('tbody');  
  
 // create basket table header  
 const thead\_tr = document.createElement('tr');  
 for (let i = 0; i < 4; i++) {  
 const thead\_th = document.createElement('th');  
 thead\_th.setAttribute('scope', 'col');  
  
 switch (i) {  
 case 0: {  
 thead\_th.innerHTML = '№';  
 break;  
 }  
 case 1: {  
 thead\_th.innerHTML = 'Title';  
 break;  
 }  
 case 2: {  
 thead\_th.innerHTML = 'Price';  
 break;  
 }  
 case 3: {  
 thead\_th.innerHTML = 'Action';  
 break;  
 }  
 }  
  
 thead\_tr.appendChild(thead\_th);  
 }  
 thead.appendChild(thead\_tr);  
  
 // create table body  
 for (let i = 0; i < basketData.length; i++) {  
 const tbody\_tr = document.createElement('tr');  
 const values = Object.values(basketData[i]);  
  
 for (let j = 0; j < 3; j++) {  
 if (j === 0) {  
 const tbody\_th = document.createElement('th');  
 tbody\_th.setAttribute('scope', 'row');  
 tbody\_th.innerHTML = values[j].toString();  
 tbody\_tr.appendChild(tbody\_th);  
 } else {  
 const tbody\_td = document.createElement('td');  
 tbody\_td.innerHTML = values[j].toString();  
 tbody\_tr.appendChild(tbody\_td);  
 }  
 }  
  
 const tbody\_td\_remove\_button = document.createElement('td');  
 const removeButton = document.createElement('button');  
  
 removeButton.setAttribute('type', 'button');  
 removeButton.setAttribute('class', 'btn btn-outline-danger p-2');  
 removeButton.setAttribute('id', 'buttonRemove');  
 removeButton.setAttribute('value', values[0].toString());  
  
 removeButton.innerHTML = 'Remove';  
  
 tbody\_td\_remove\_button.appendChild(removeButton);  
 tbody\_tr.appendChild(tbody\_td\_remove\_button);  
 tbody.appendChild(tbody\_tr);  
 }  
  
 const tfoot = document.createElement('tfoot');  
 tfoot.setAttribute('class', 'table-dark');  
 const tfootTr = document.createElement('tr');  
 const tfootLabelTd = document.createElement('td');  
 tfootLabelTd.setAttribute('colspan', '2');  
 tfootLabelTd.setAttribute('scope', 'row');  
 tfootLabelTd.setAttribute('class', 'text-center');  
 tfootLabelTd.innerHTML = 'Total price:';  
  
 const tfootTotalPriceTd = document.createElement('td');  
 tfootTotalPriceTd.setAttribute('colspan', '2');  
 tfootTotalPriceTd.setAttribute('class', 'text-center');  
 let totalPrice = 0;  
 for (let i = 0; i < basketData.length; i++) {  
 totalPrice += parseInt(basketData[i].price);  
 }  
 tfootTotalPriceTd.innerHTML = totalPrice.toString();  
  
 tfootTr.appendChild(tfootLabelTd);  
 tfootTr.appendChild(tfootTotalPriceTd);  
 tfoot.appendChild(tfootTr);  
 table.appendChild(thead);  
 table.appendChild(tbody);  
 table.appendChild(tfoot);  
  
 return table;  
 } else {  
 const alert = document.createElement('div');  
 alert.setAttribute('class', 'alert alert-dark');  
 alert.setAttribute('id', 'alertNoProduct');  
 alert.innerHTML = 'Here is no items in basket!';  
 document.querySelector('#basketTitle').parentElement.appendChild(alert);  
  
 return null;  
 }  
};  
  
const deleteTable = () => {  
 const table = document.querySelector('#basketTable');  
 table.remove();  
};  
  
function addEventsToRemoveButtons() {  
 const removeButtons = document.querySelectorAll('#buttonRemove');  
  
 for (let i = 0; i < removeButtons.length; i++) {  
 removeButtons[i].addEventListener('click', (event) => {  
 const target = event.target;  
 const id = parseInt(target.getAttribute('value'));  
  
 deleteBasketDataById(id);  
 deleteTable();  
 callTableCreation();  
 });  
 }  
}  
  
function callTableCreation() {  
 const table = createBasketTable();  
 if (table) {  
 showBasketTable(table);  
 }  
}  
  
callTableCreation()

* + ***localStorageData.js:***

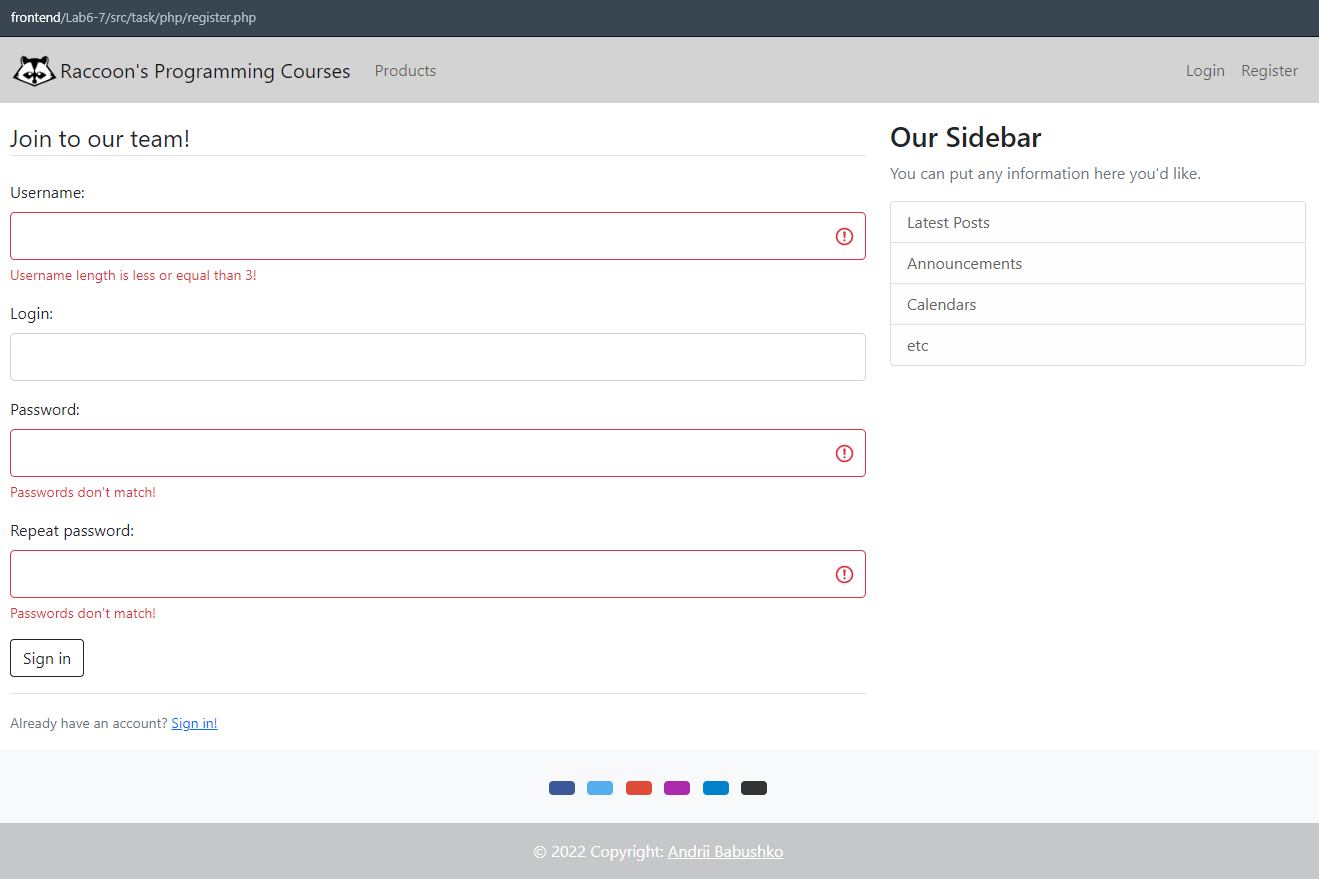
function readProgrammingCoursesData() {  
 let programmingCourses = null;  
  
 if (localStorage.getItem('programmingCourses') === null) {  
 programmingCourses = [];  
 } else {  
 programmingCourses = JSON.parse(localStorage.getItem('programmingCourses'));  
 }  
  
 return programmingCourses;  
}  
  
function addProgrammingCoursesData(addData) {  
 const programmingCourses = readProgrammingCoursesData();  
  
 programmingCourses.push(addData);  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses))  
}  
  
function updateProgrammingCoursesDataById(updateData, id) {  
 const programmingCourses = readProgrammingCoursesData();  
  
 const objIndex = programmingCourses.findIndex((obj => obj.id == id));  
  
 programmingCourses[objIndex].title = updateData.title;  
 programmingCourses[objIndex].price = updateData.price;  
 programmingCourses[objIndex].time = updateData.time;  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
}  
  
function deleteProgrammingCoursesDataById(id) {  
 const programmingCourses = readProgrammingCoursesData();  
  
 for (let j = programmingCourses.length - 1; j >= 0; --j) {  
 if (programmingCourses[j].id === id) {  
 programmingCourses.splice(j, 1);  
 }  
 }  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
}  
  
function ascendingProgrammingCoursesDataSort(){  
 const programmingCourses = readProgrammingCoursesData();  
 programmingCourses.sort((obj1, obj2) => (obj1.id > obj2.id) ? 1 : (obj1.id < obj2.id) ? -1 : 0);  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
}  
  
function readBasketData() {  
 let basketData = null;  
  
 if (localStorage.getItem('basketData') === null) {  
 basketData = [];  
 } else {  
 basketData = JSON.parse(localStorage.getItem('basketData'));  
 }  
  
 return basketData;  
}  
  
function addBasketData(addData) {  
 const basketData = readBasketData();  
  
 basketData.push(addData);  
  
 localStorage.setItem('basketData', JSON.stringify(basketData))  
}  
  
function deleteBasketDataById(id) {  
 const basketData = readBasketData();  
  
 for (let j = basketData.length - 1; j >= 0; --j) {  
 if (basketData[j].id === id) {  
 basketData.splice(j, 1);  
 }  
 }  
  
 localStorage.setItem('basketData', JSON.stringify(basketData));  
}  
  
function readCommentsData(programmingCourseIndex) {  
 let commentsData = null;  
  
 if (localStorage.getItem('programmingCourses') === null) {  
 commentsData = [];  
 } else {  
 const programmingCourses = JSON.parse(localStorage.getItem('programmingCourses'));  
 commentsData = programmingCourses[programmingCourseIndex].comments;  
 }  
  
 return commentsData;  
}  
  
function addCommentsData(addData, programmingCourseIndex) {  
 const commentsData = readCommentsData(programmingCourseIndex);  
  
 commentsData.push(addData);  
  
 let programmingCourses = JSON.parse(localStorage.getItem('programmingCourses'));  
 programmingCourses[programmingCourseIndex].comments = commentsData;  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
}  
  
function deleteCommentsDataById(programmingCourseIndex, id) {  
 const commentsData = readCommentsData(programmingCourseIndex);  
  
 for (let j = commentsData.length - 1; j >= 0; --j) {  
 if (commentsData[j].id === id) {  
 commentsData.splice(j, 1);  
 }  
 }  
  
 let programmingCourses = JSON.parse(localStorage.getItem('programmingCourses'));  
 programmingCourses[programmingCourseIndex].comments = commentsData;  
  
 localStorage.setItem('programmingCourses', JSON.stringify(programmingCourses));  
}

***Результат програми:***

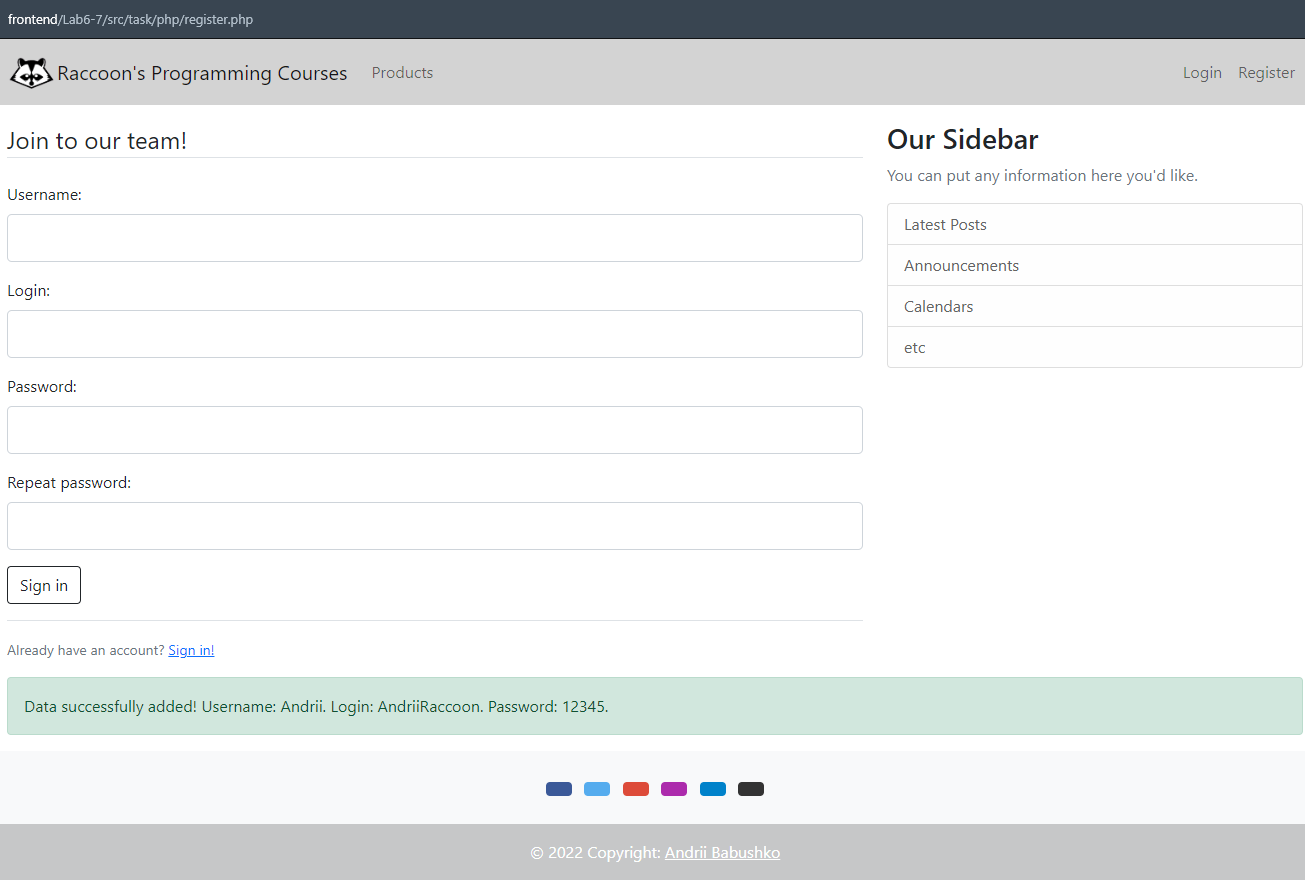
1. ***Форма регістрації:***

******

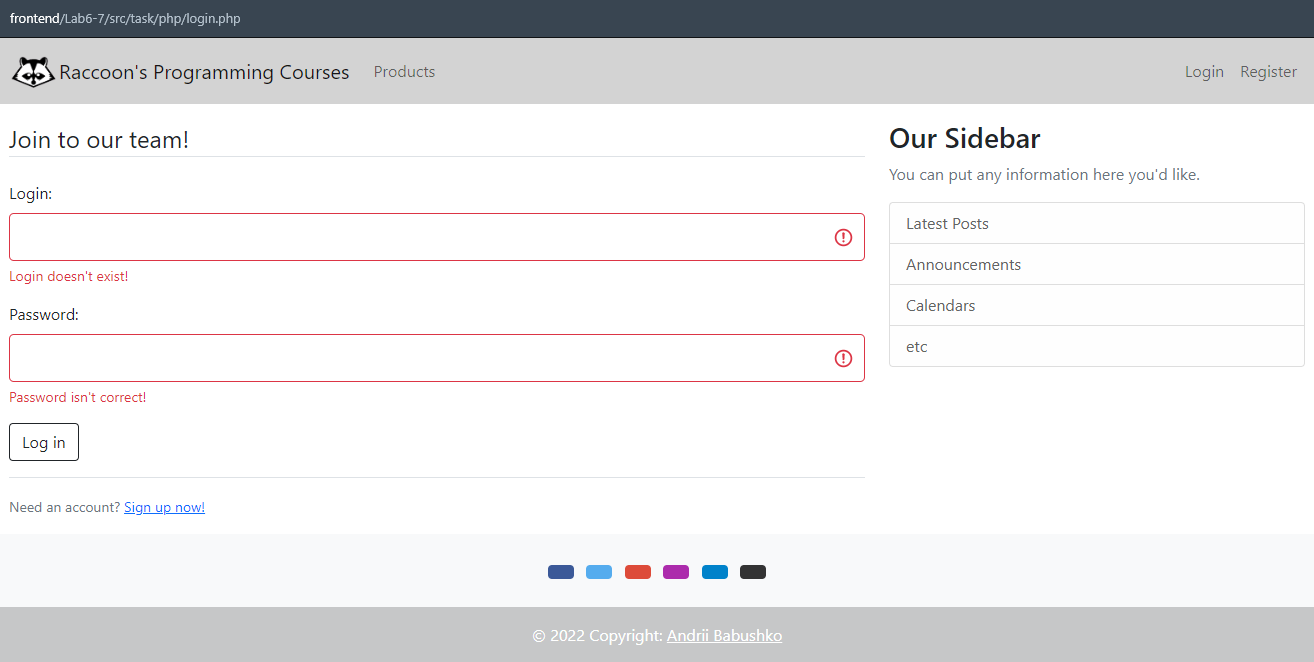
1. ***Валідація полів форми регістрації:***

******

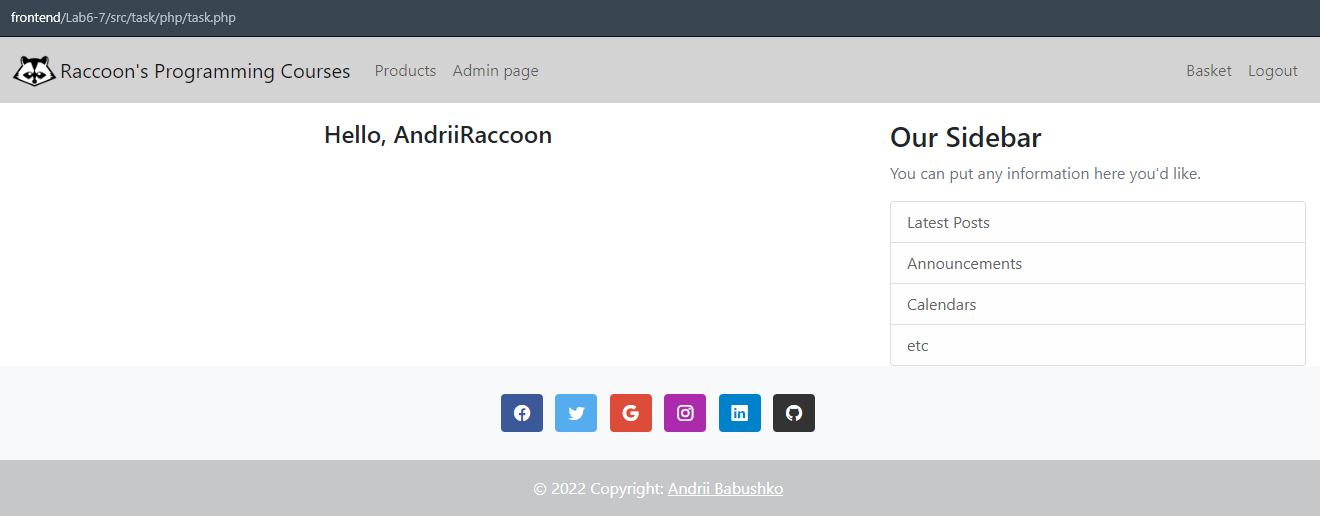
1. ***Успішна регістрація:***

******

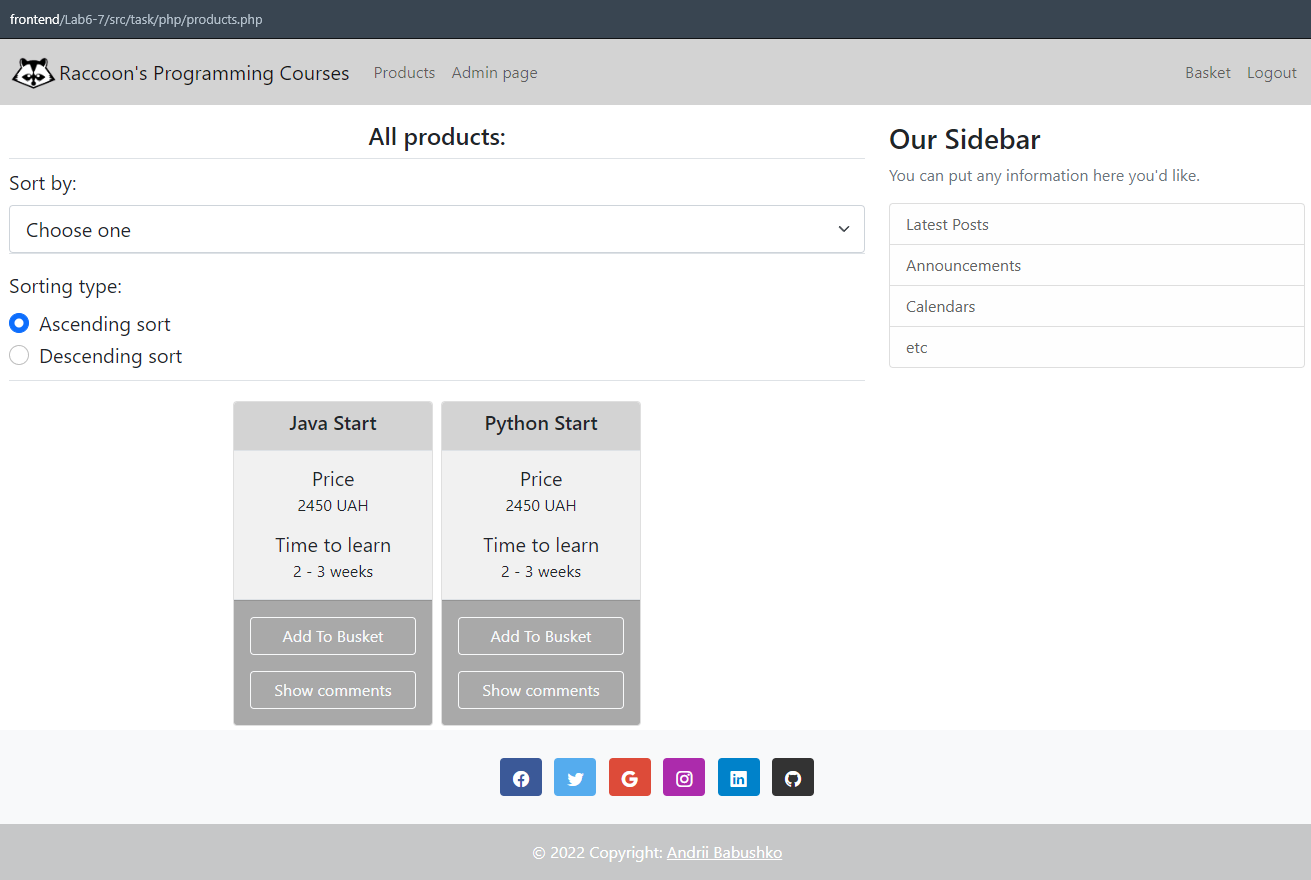
1. ***Валідація полів форми логіну:***

******

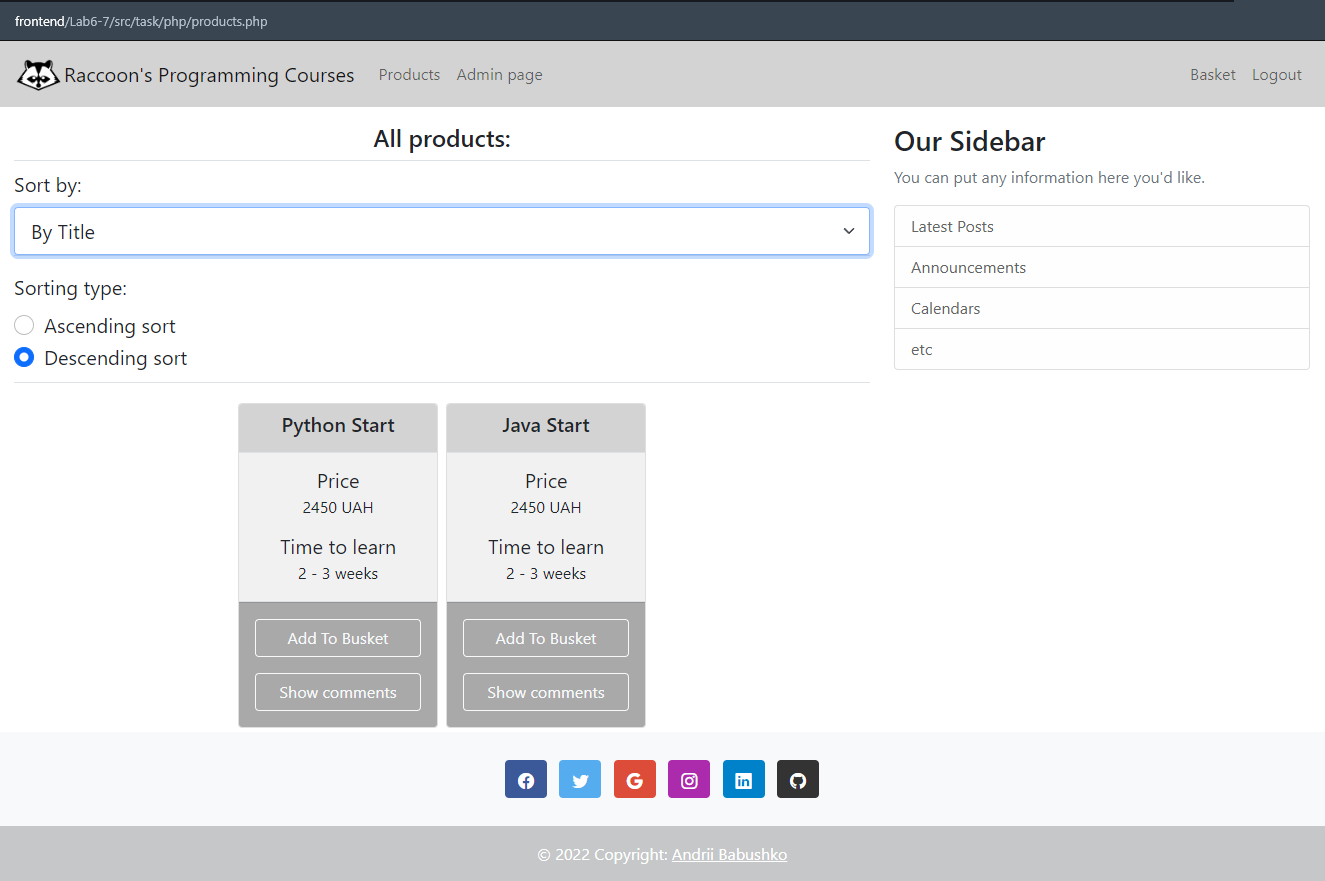
1. ***Успішний логін:***

******

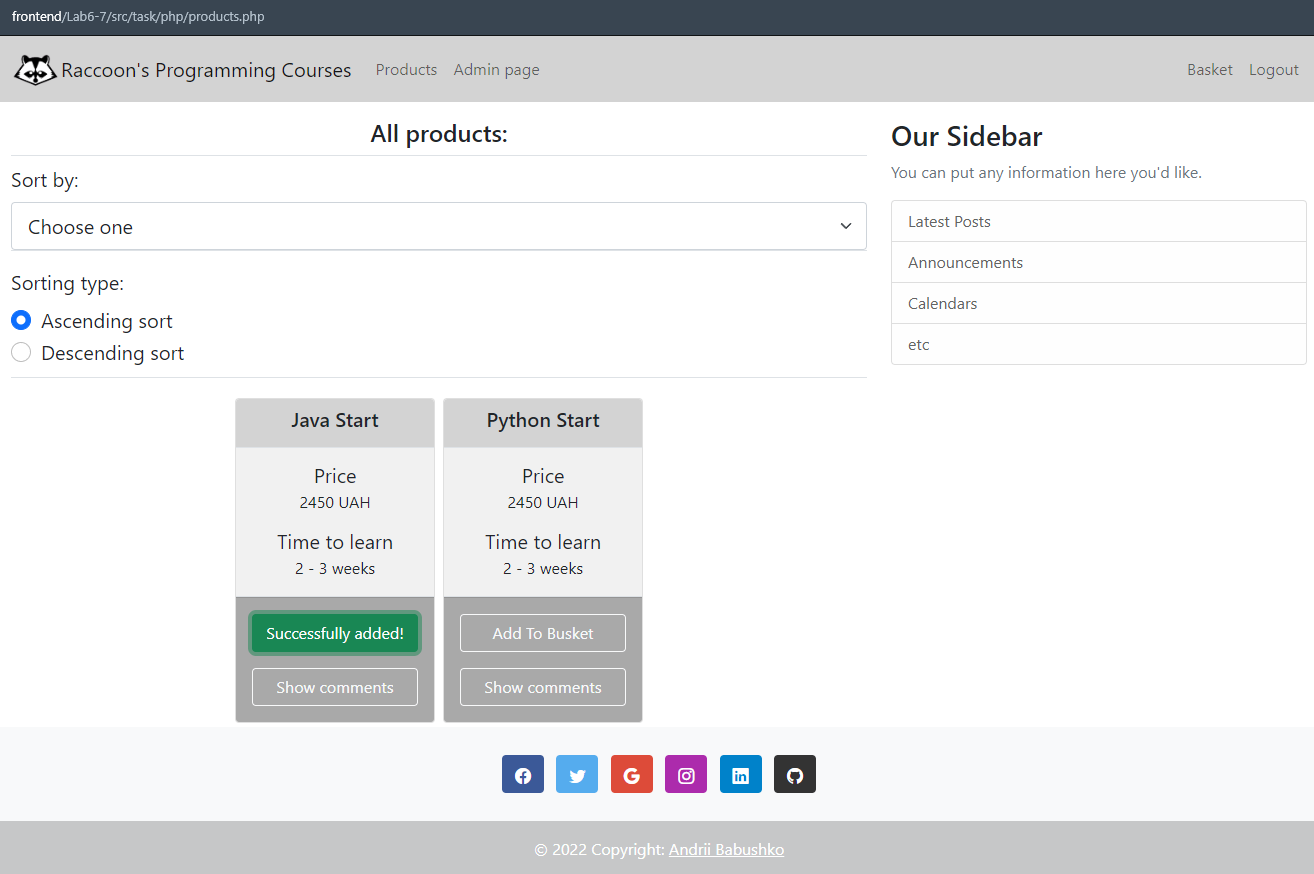
1. ***Сторінка продуктів:***

******

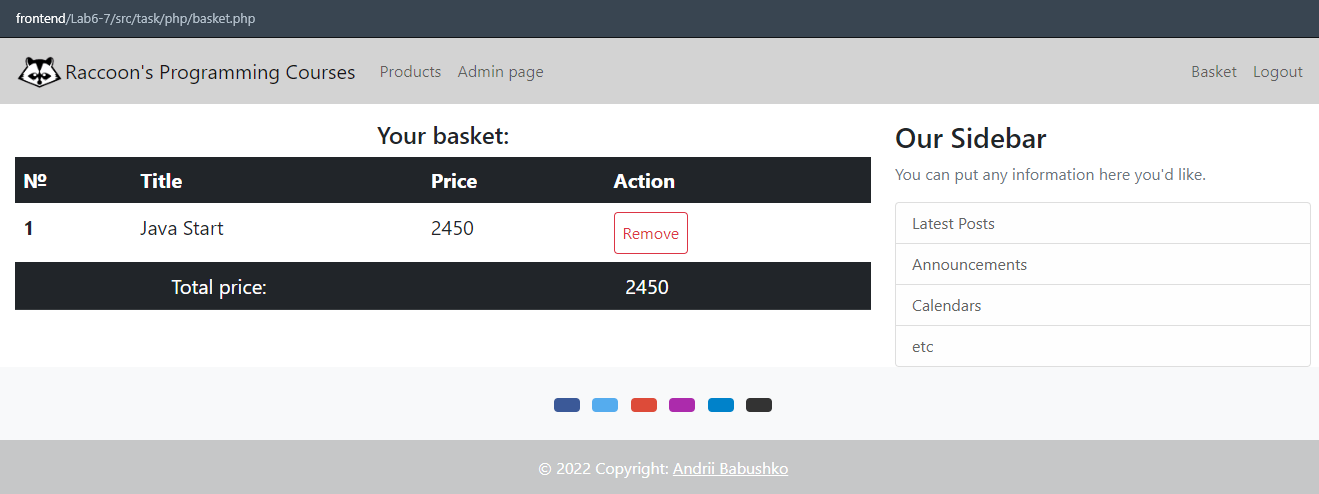
1. ***Сортування продуктів:***

******

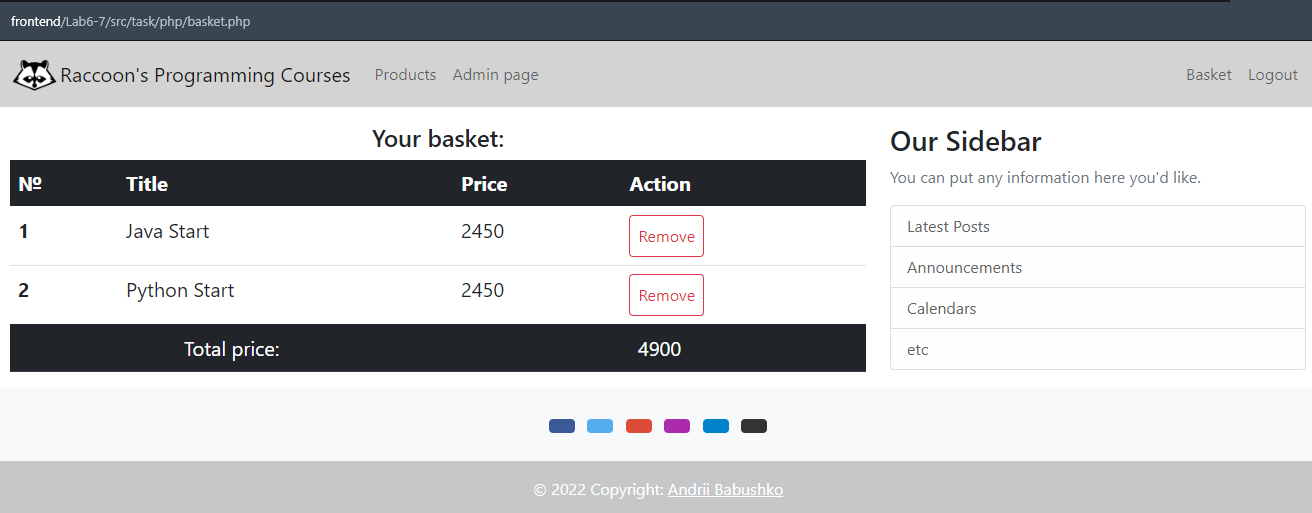
1. ***Додавання продуктів до корзини:***

******

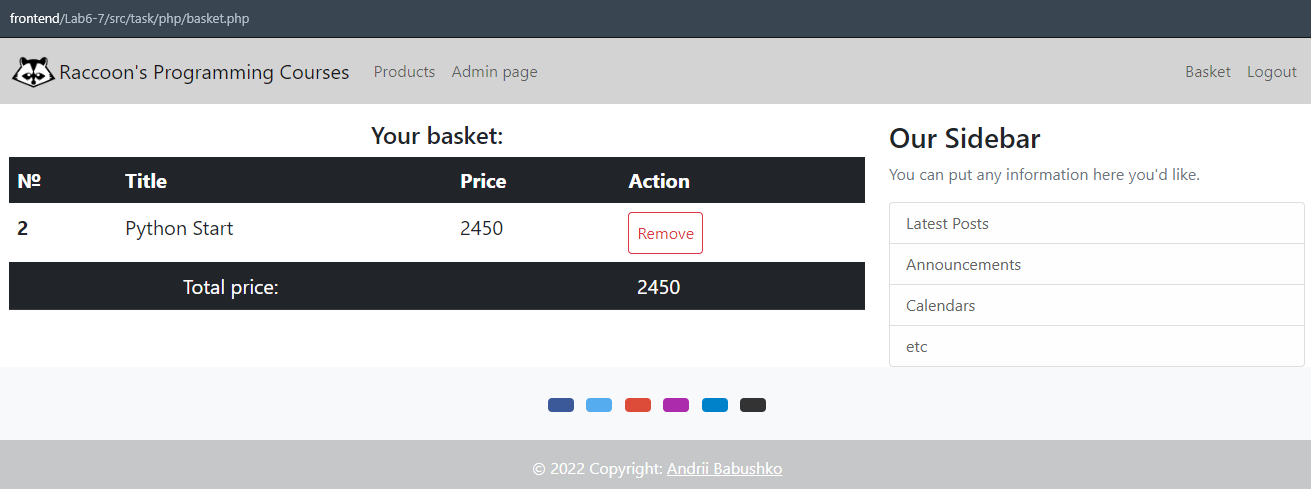
1. ***Сторінка корзини після додавання:***

******

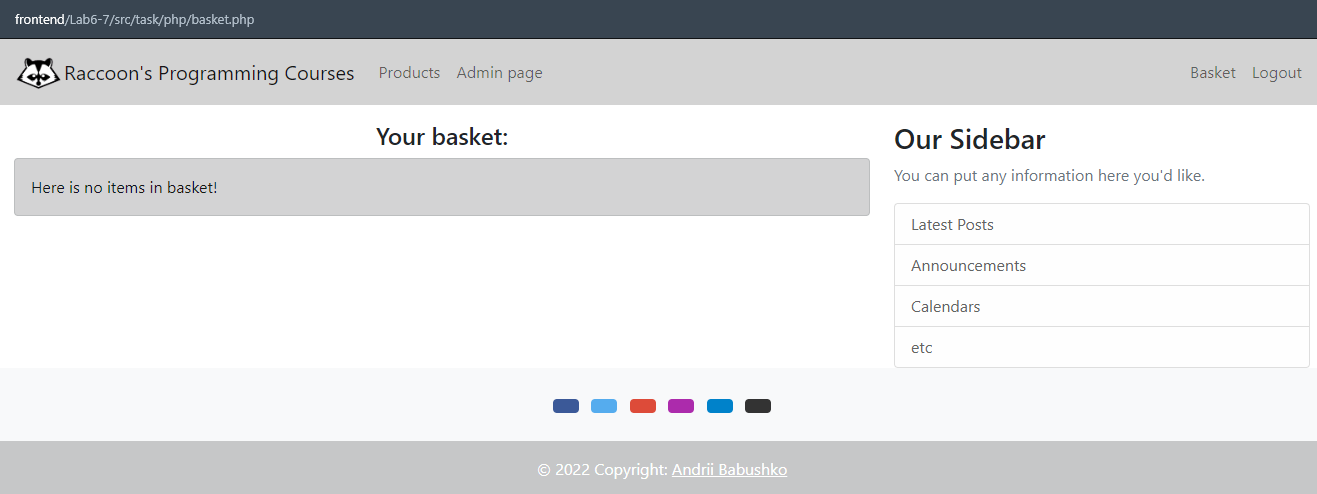
1. ***Сума покупки у корзині розраховується:***

******

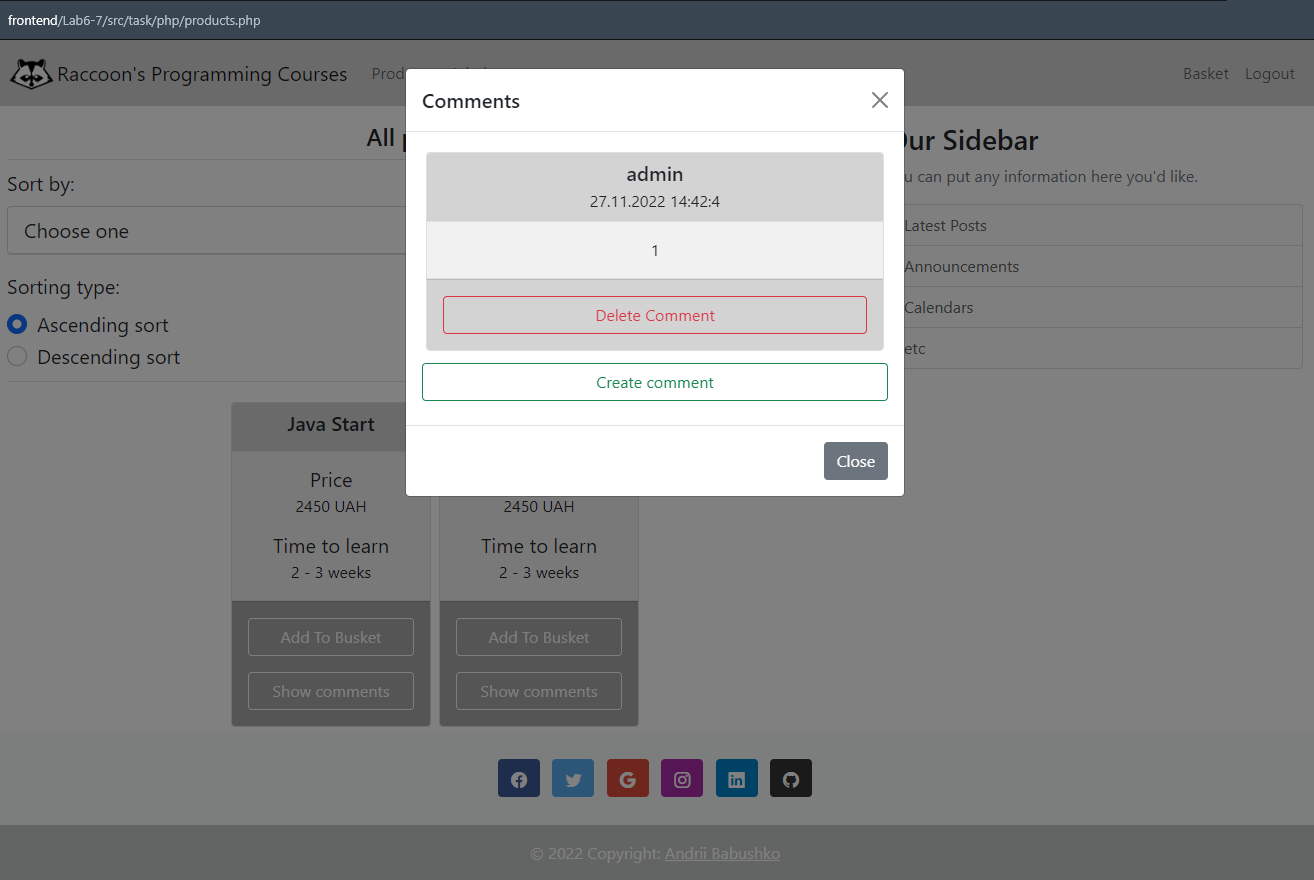
1. ***Сторінка корзини після видалення одного продукту:***

******

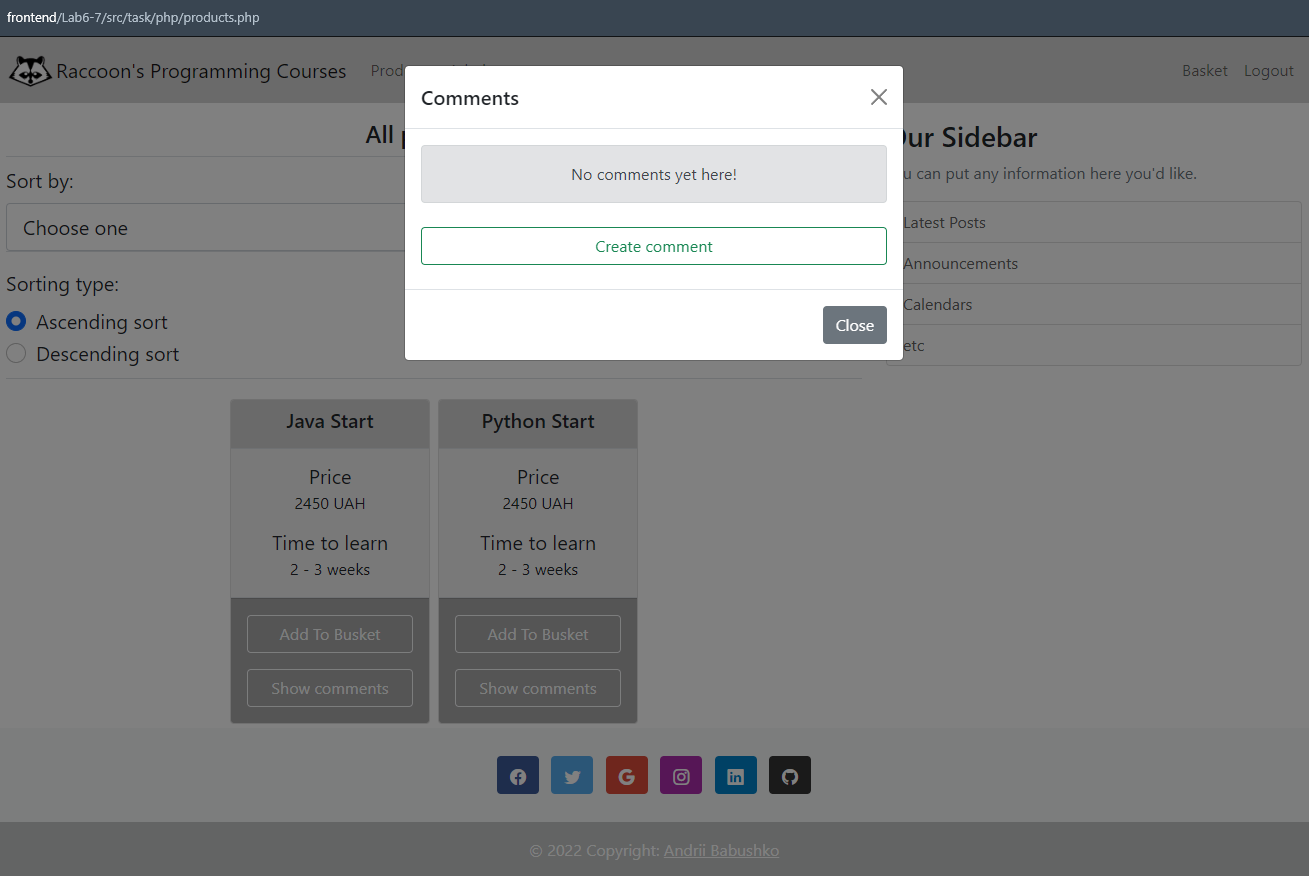
1. ***Сторінка корзини після видалення усіх продуктів:***

******

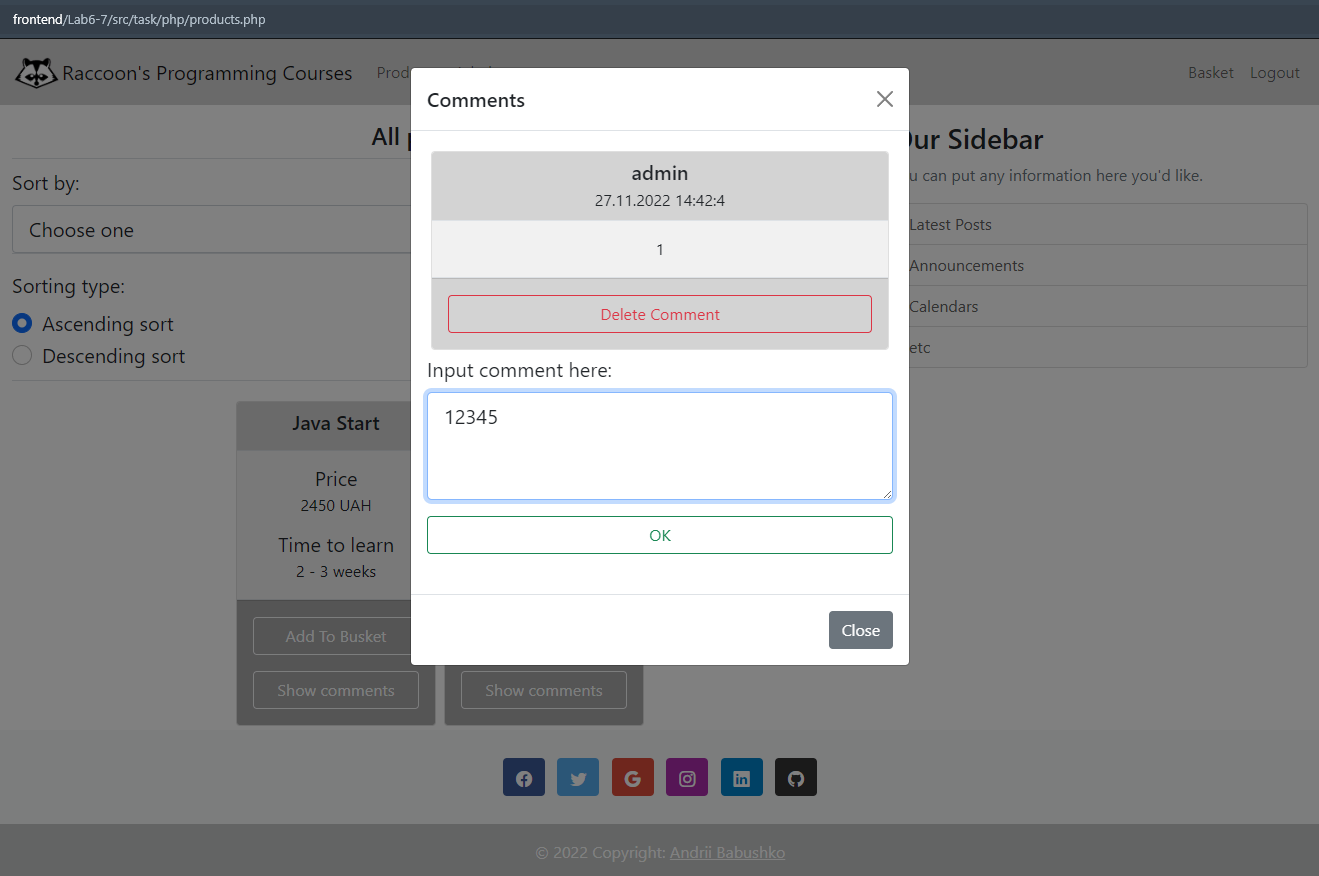
1. ***Відображення коментарів продукту:***

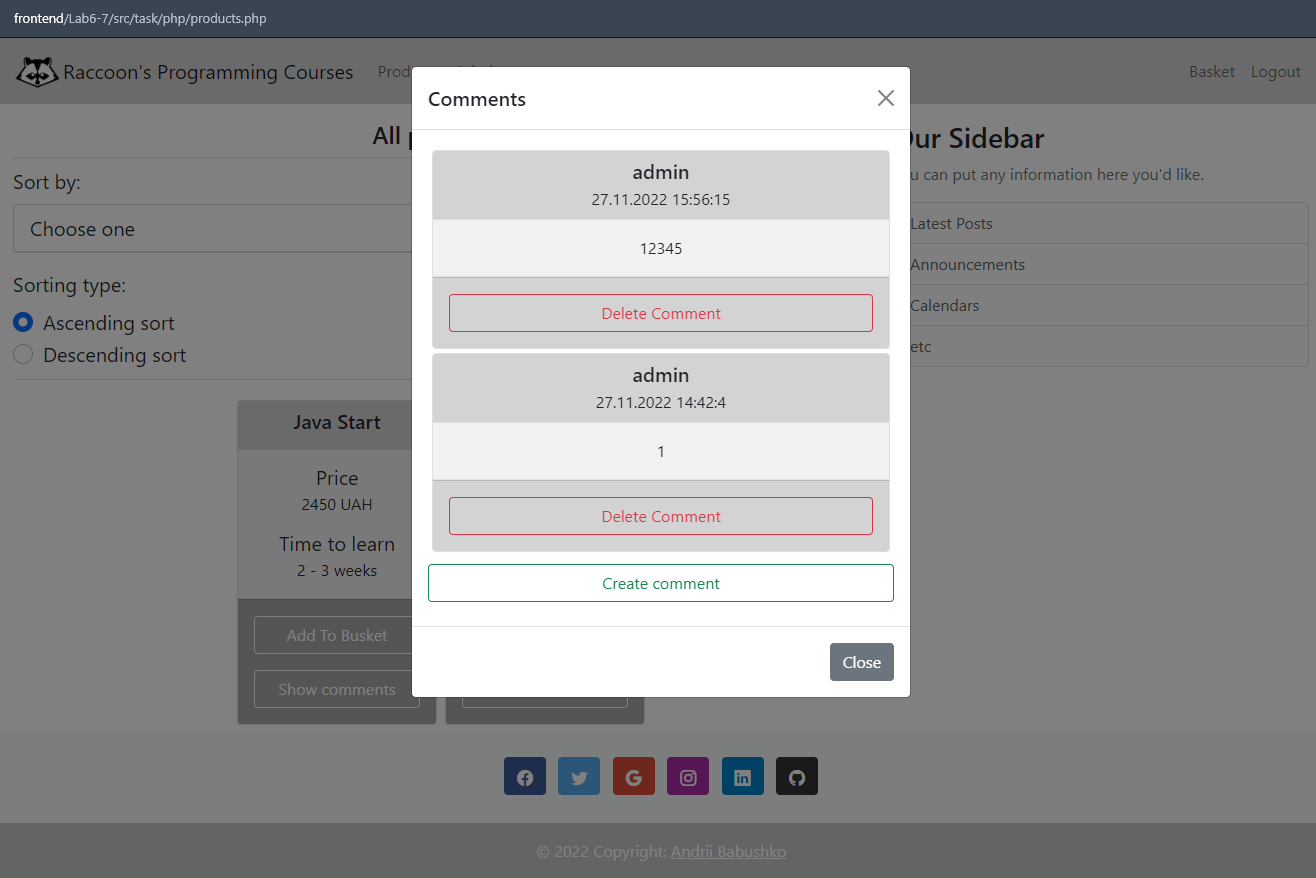
******

1. ***Коментарі іншого продукту:***

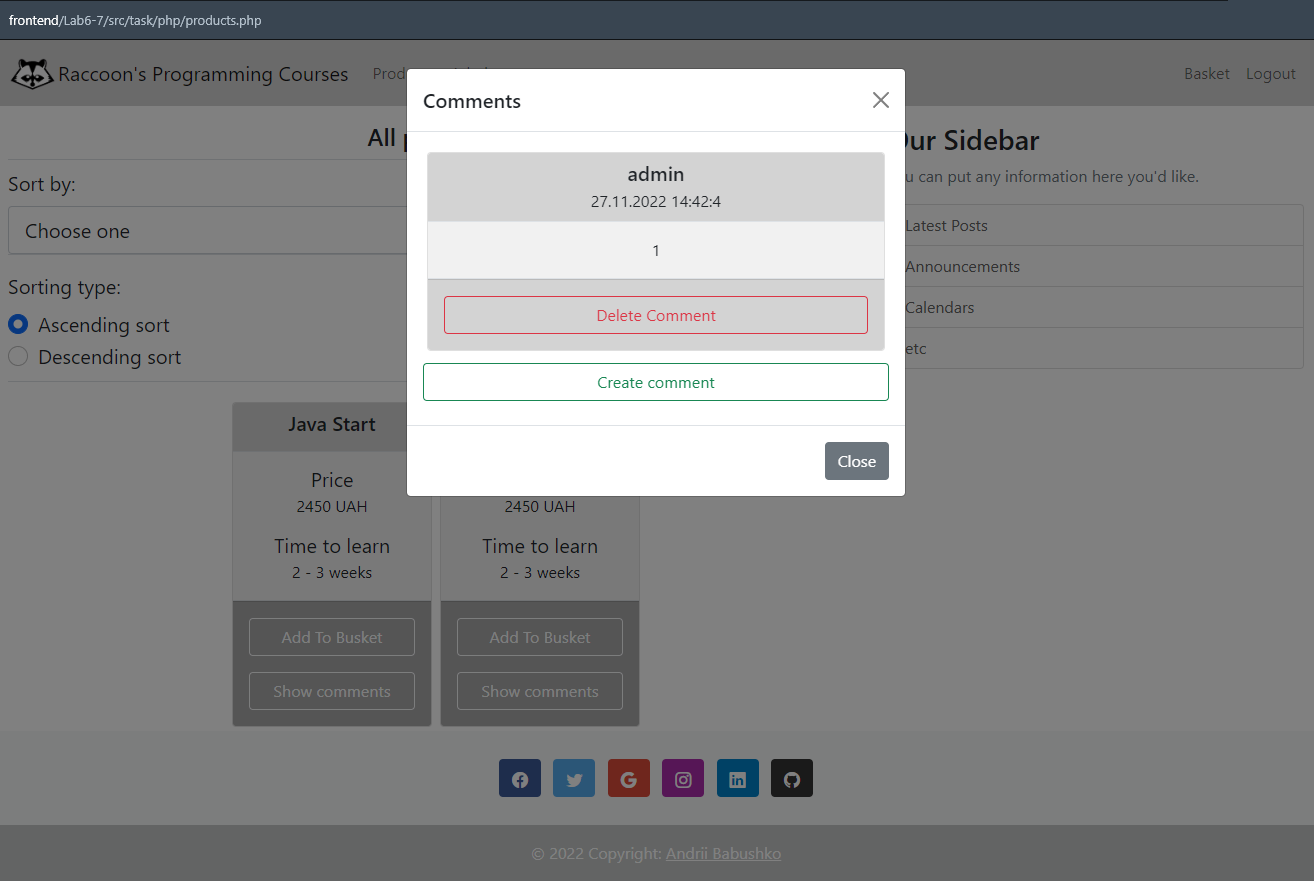
******

1. ***Додавання коментарів до 1 продукту:***

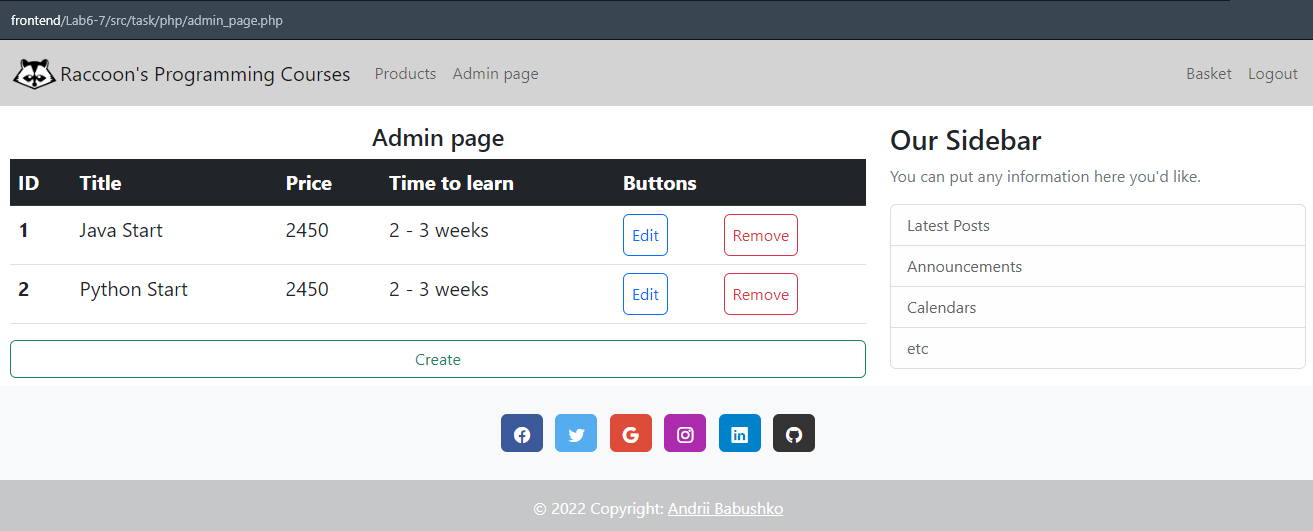
******

******

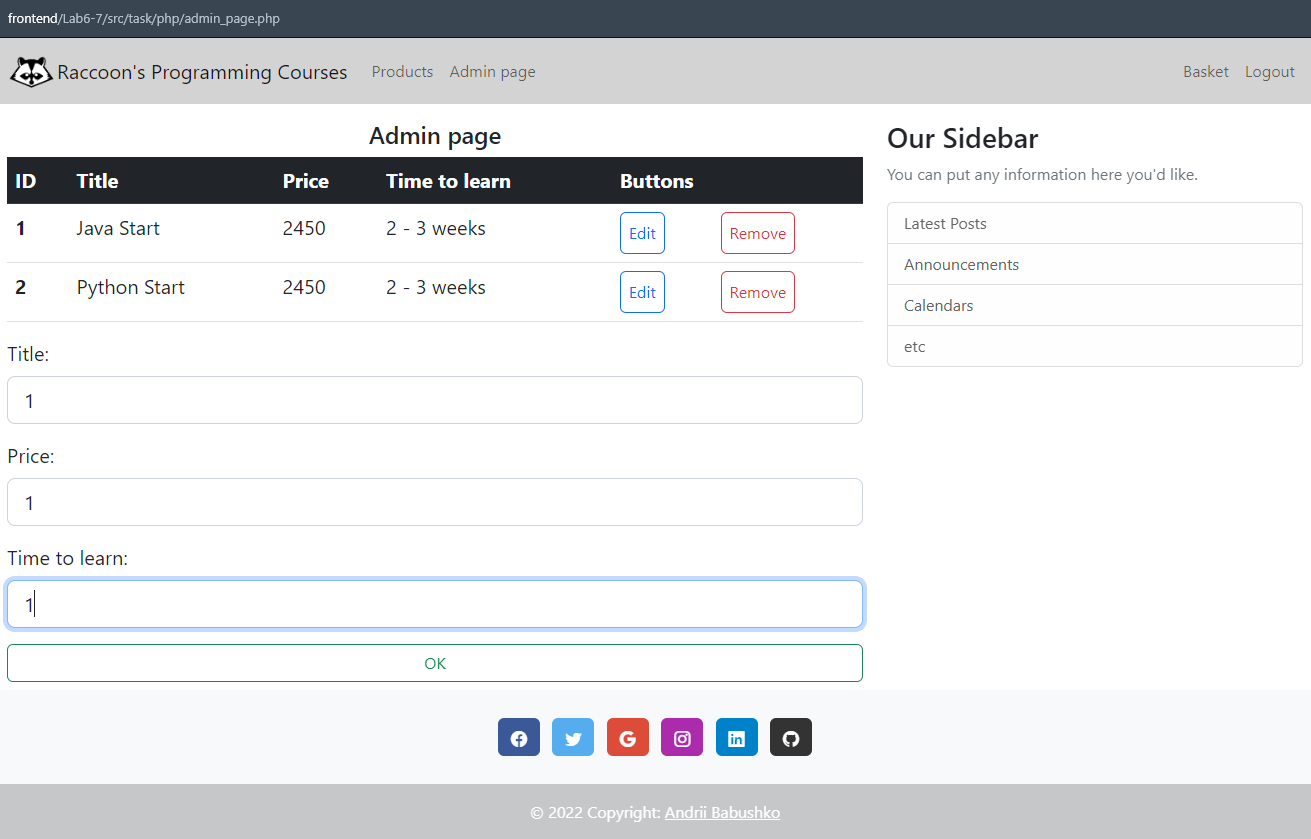
1. ***Видалення доданого коментаря:***

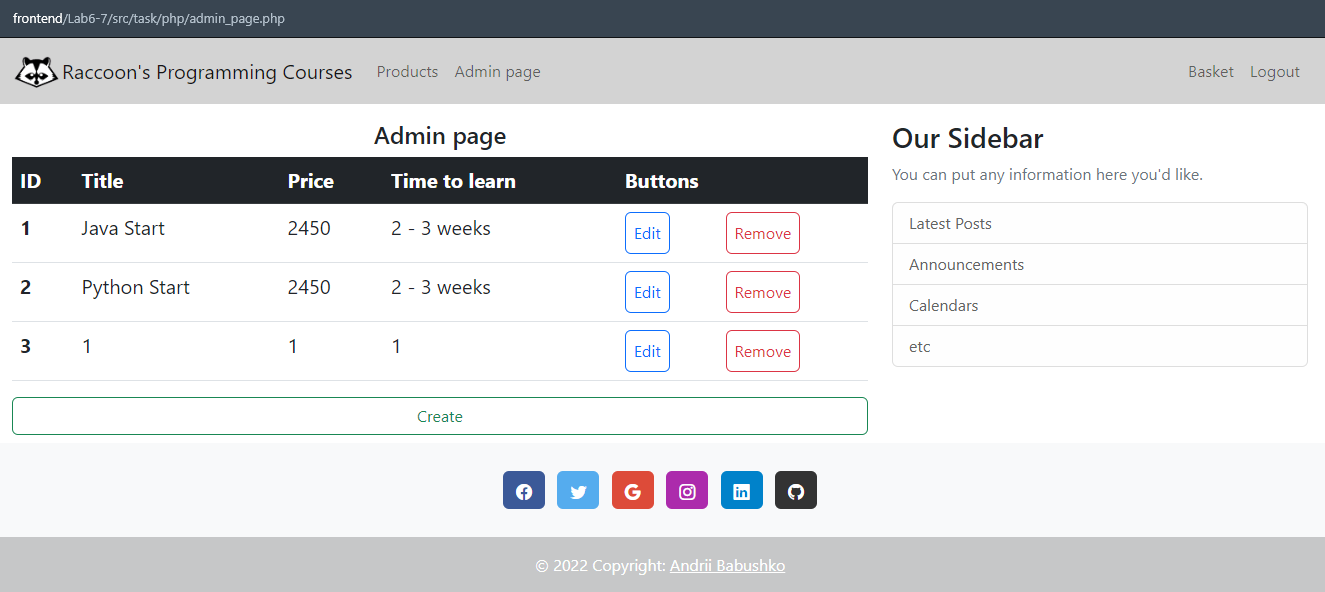
******

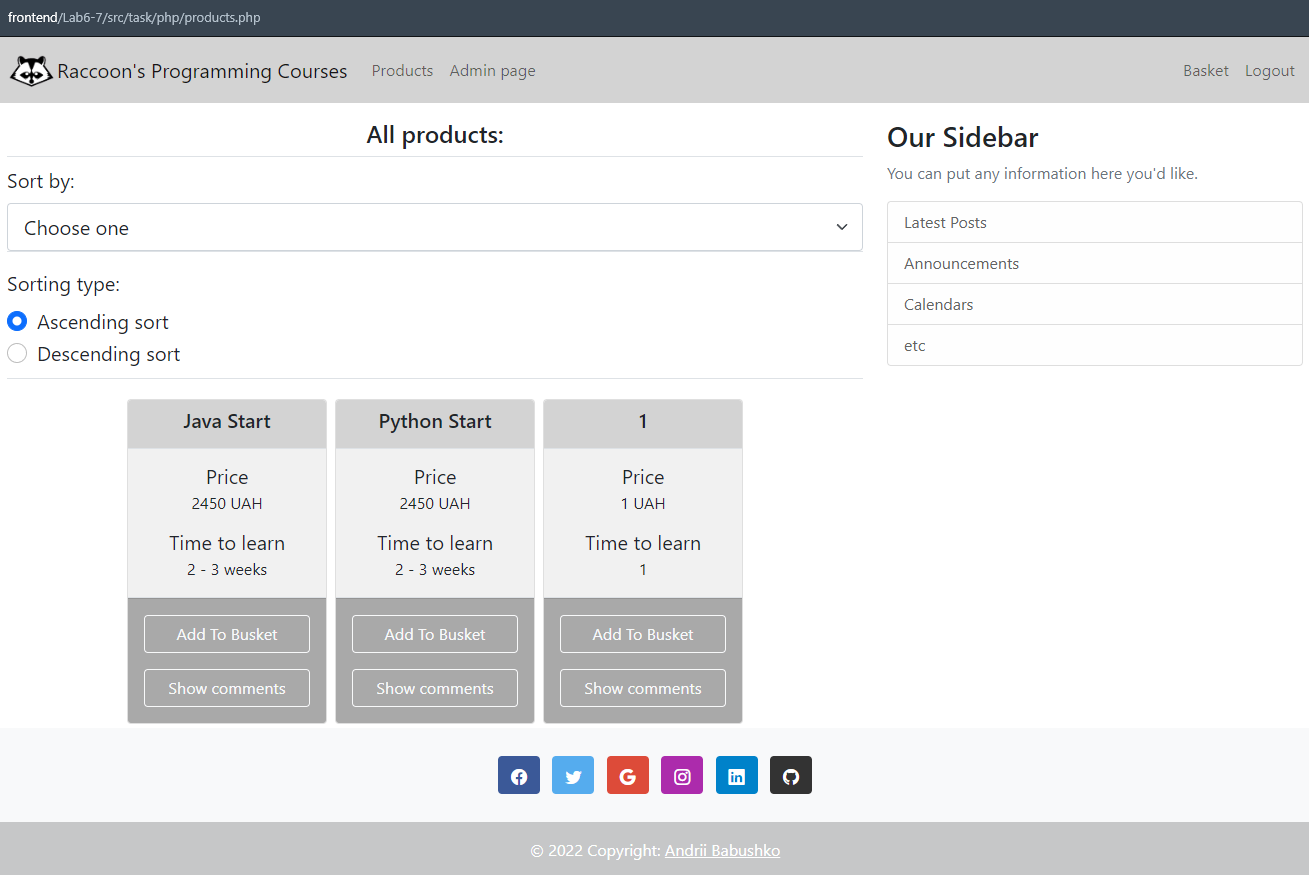
1. ***Вигляд адмін панелі:***

******

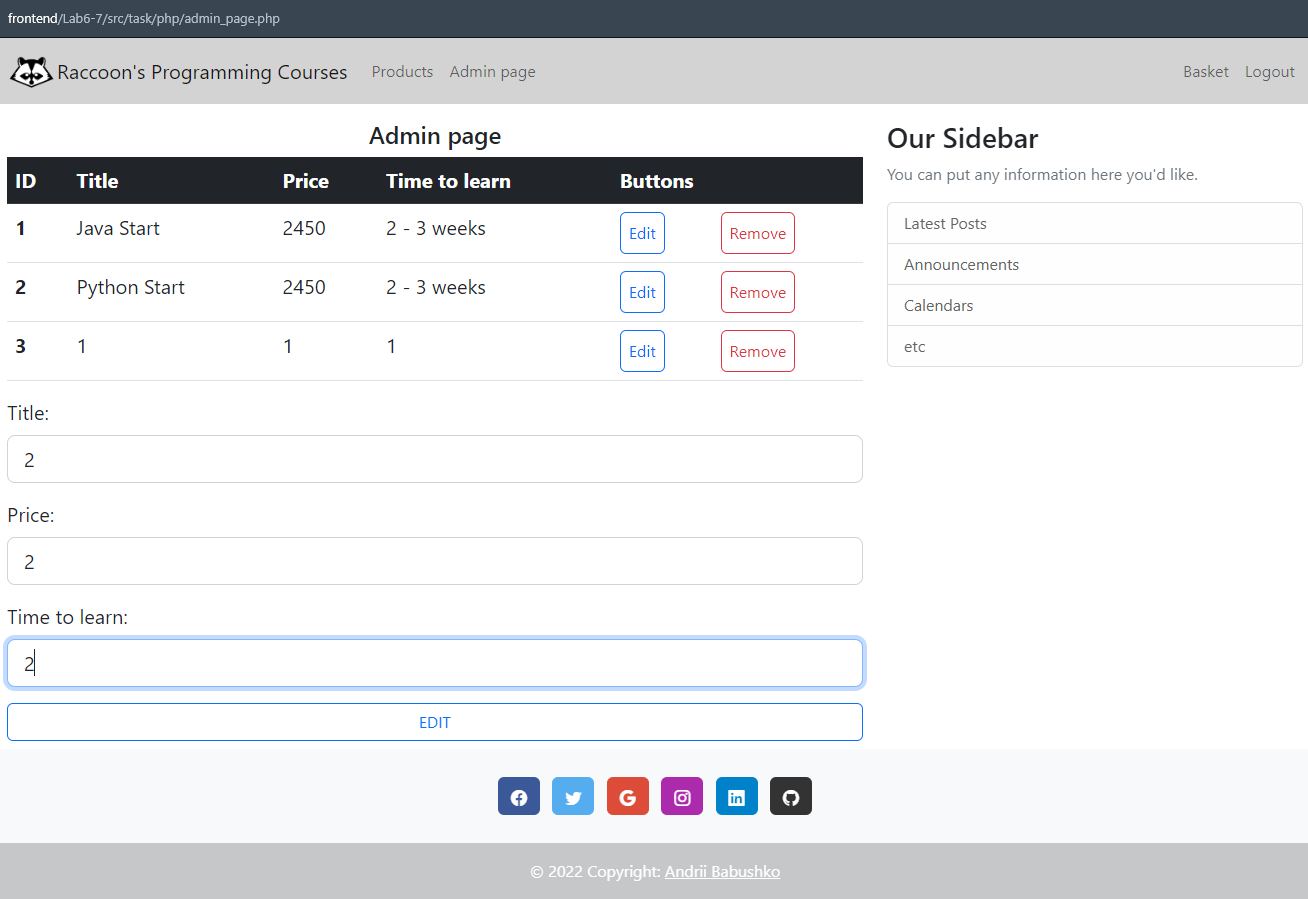
1. ***Додавання продукту:***

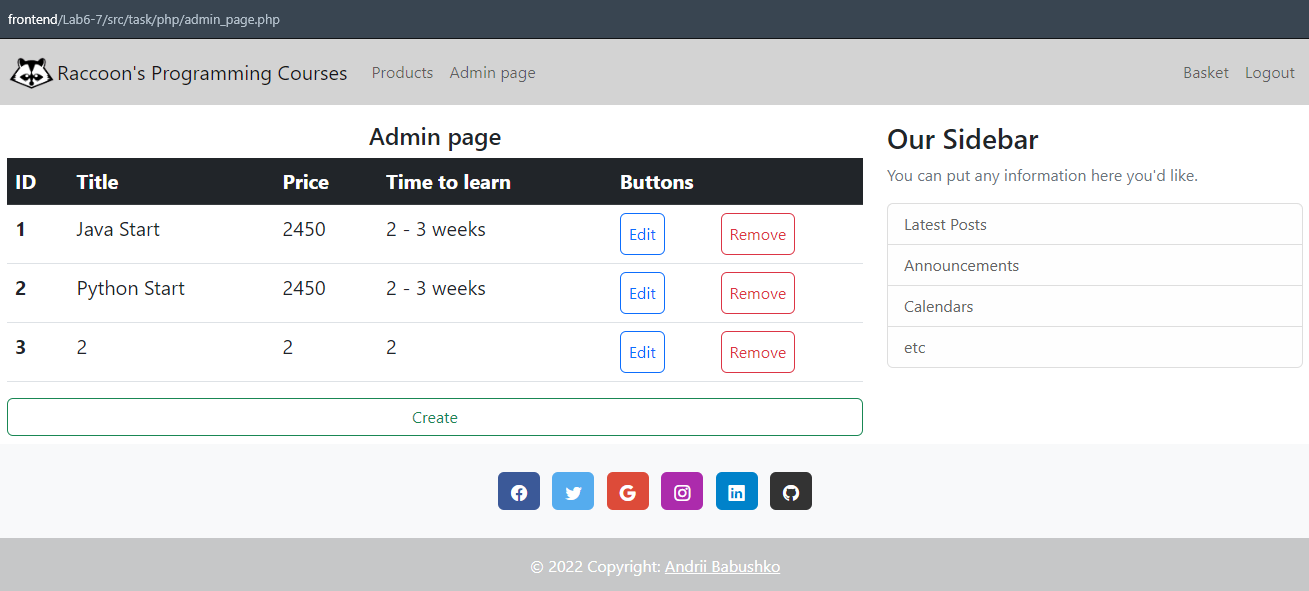
******

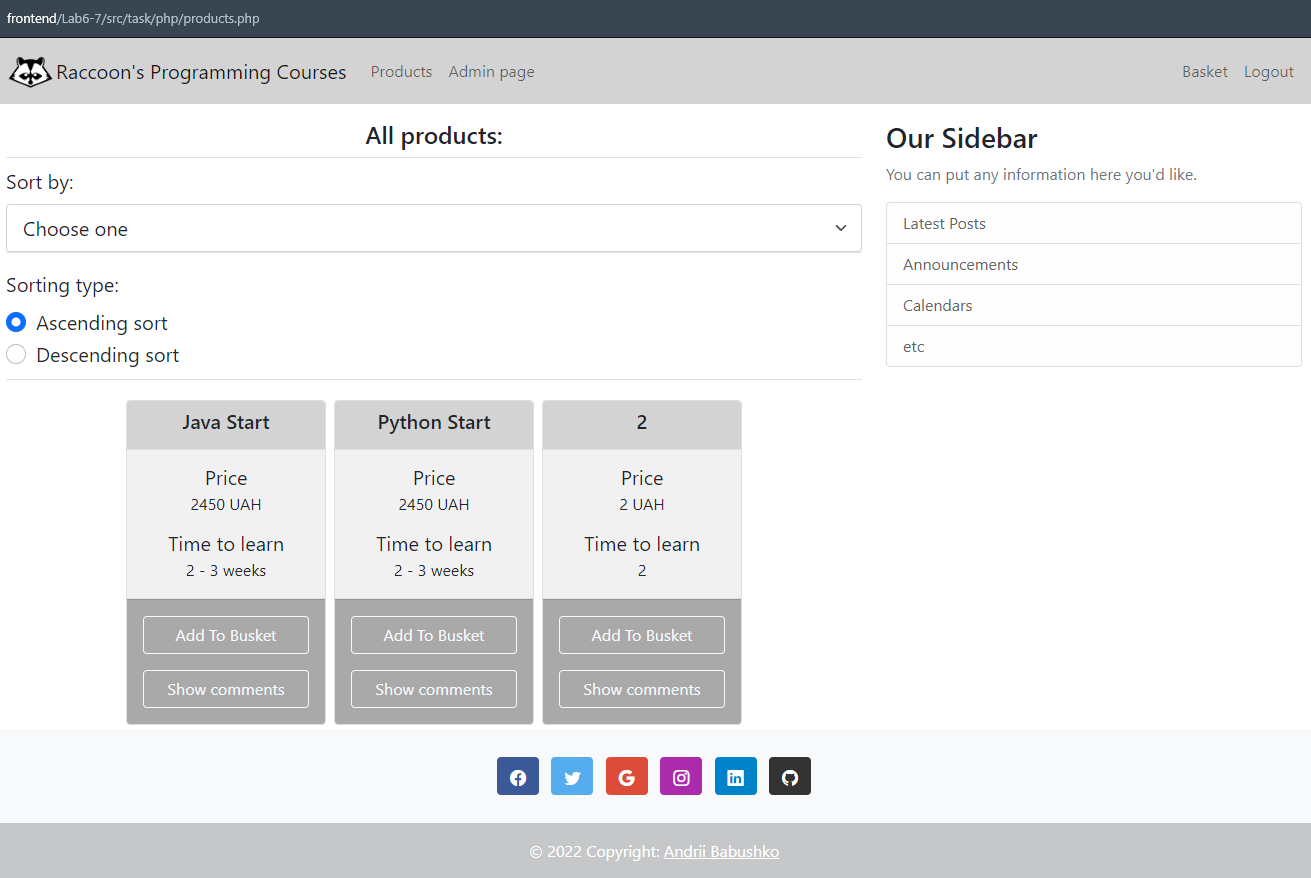
******

******

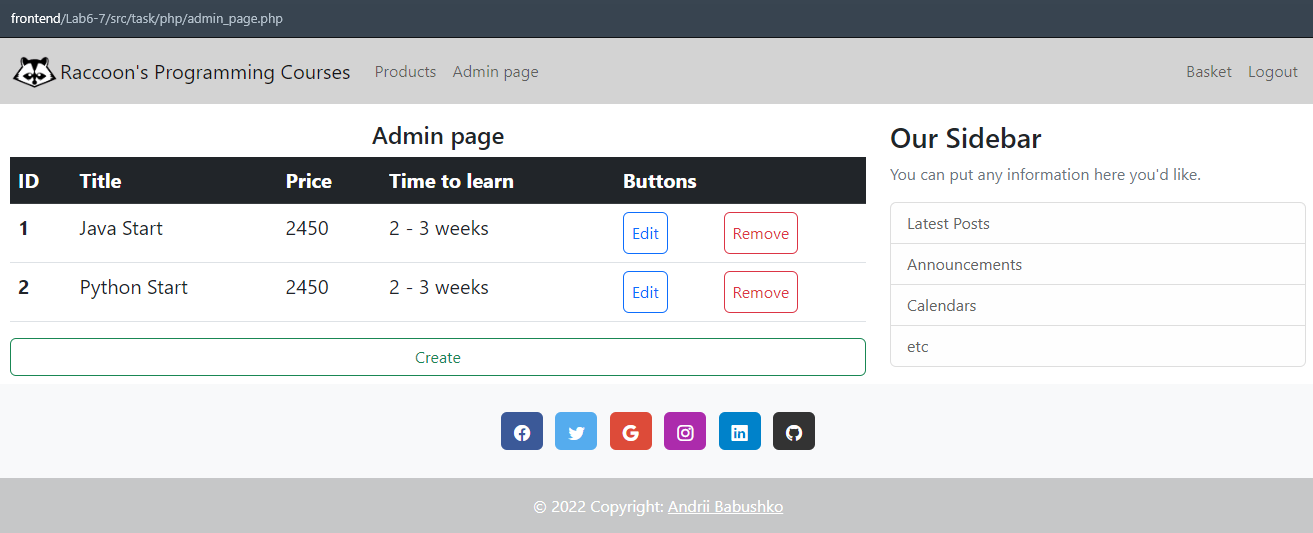
1. ***Редагування доданого продукту:***

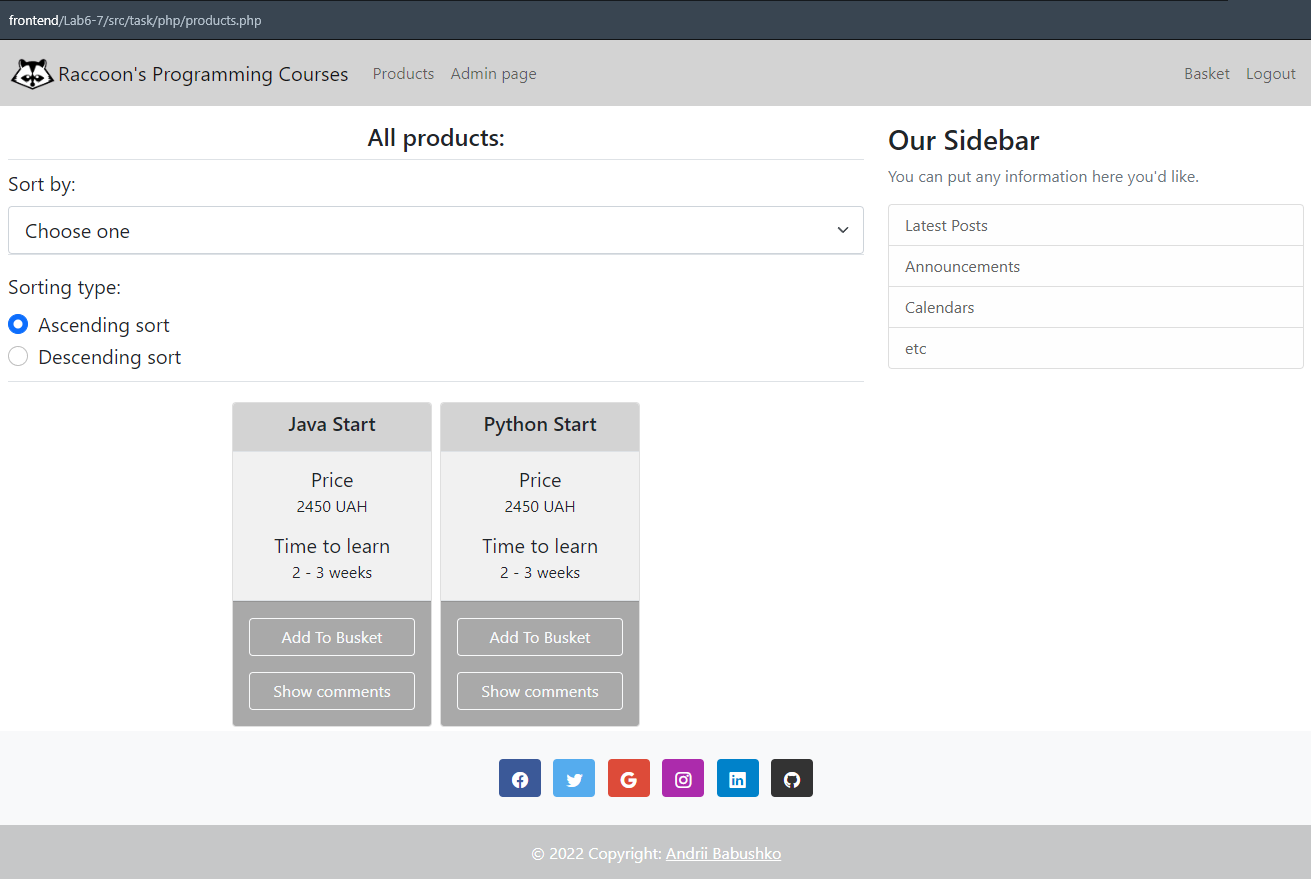
******

******

******

1. ***Видалення доданого продукту:***

******

******

***Репозиторій лабораторної роботи:***

* https://github.com/AndriiBabushko/Frontend

***Висновок:*** під час виконання лабораторної роботи було отримано навички роботи з localstorage, JSON, створення модальних вікон, навігаційної панелі, таблиці та іншого за допомогою бібліотеки Bootstrap 5 в мові JavaScript.