گزارش پروژه

Config.php

1. $serverip="localhost";: در این خط، متغیر $serverip به مقدار localhost تنظیم شده است. این متغیر معمولاً برای نشان دادن آدرس IP یا نام سرور پایگاه داده استفاده می‌شود.

2. $dbname="laptopshop";: در این خط، متغیر $dbname به مقدار laptopshop تنظیم شده است. این متغیر نام پایگاه داده‌ای است که برنامه به آن متصل می‌شود.

3. $username="root";: در این خط، متغیر $username به مقدار root تنظیم شده است. این متغیر نام کاربر پایگاه داده است که برای ورود به پایگاه داده استفاده می‌شود.

4. $password="";: در این خط، متغیر $password به مقدار خالی تنظیم شده است. این متغیر رمز عبور کاربر پایگاه داده است که برای ورود به پایگاه داده استفاده می‌شود.

5. $conn = mysqli\_connect($serverip,$username,$password,$dbname);: در این خط، ارتباط با پایگاه داده برقرار می‌شود. تابع mysqli\_connect() برای اتصال به پایگاه داده MySQL استفاده می‌شود و آرگومان‌های ورودی به ترتیب شامل آدرس IP یا نام سرور، نام کاربر، رمز عبور و نام پایگاه داده هستند.

6. mysqli\_query($conn,"SET NAMES 'utf8'");, mysqli\_query($conn,"SET CHARACTER SET 'utf8'");, mysqli\_query($conn,"SET character\_set\_connection = 'utf8'");: این سه خط کد برای تنظیم کدگذاری UTF-8 برای ارتباط با پایگاه داده استفاده می‌شود. این کدها اطمینان حاصل می‌کنند که داده‌های فارسی یا دیگر زبان‌ها به درستی ذخیره و بازیابی شوند.

به طور کلی، این بخش کد مسئول برقراری ارتباط با پایگاه داده MySQL و تنظیمات مربوط به کدگذاری UTF-8 استفاده شده در فروشگاه آنلاین مورد نظر است.

Cart.php

1. session\_start(): این دستور به PHP می‌گوید که از جلسات استفاده کند. جلسات برای نگهداری اطلاعات کاربر (مانند سبد خرید) بین صفحات وب استفاده می‌شود.

2. require\_once("config.php"): این دستور فایل config.php را وارد می‌کند. این فایل معمولاً شامل تنظیمات و اطلاعات پایگاه داده است و برای برقراری ارتباط با پایگاه داده استفاده می‌شود.

3. if(isset($\_GET['deleteId'])) { ... }: این بخش کد چک می‌کند که آیا پارامتر deleteId در URL وجود دارد یا خیر. اگر وجود داشته باشد، محصول مربوط به deleteId از سبد خرید حذف می‌شود و تعداد محصولات در سبد نیز کاهش می‌یابد.

4. if(isset($\_GET['emptyCart'])) { ... }: این بخش کد چک می‌کند که آیا پارامتر emptyCart در URL وجود دارد یا خیر. اگر وجود داشته باشد، تمام محصولات از سبد خرید حذف شده و تعداد محصولات در سبد صفر می‌شود.

به طور کلی، این بخش کد مسئول حذف محصولات از سبد خرید یا خالی کردن سبد خرید است.

Header.php

1. if (!isset($\_SESSION)){ session\_start(); }: در این خط، یک شرط اولیه برای بررسی اینکه آیا متغیر $\_SESSION تنظیم شده است یا خیر، قرار دارد. اگر $\_SESSION تنظیم نشده باشد، تابع session\_start() فراخوانی می‌شود تا یک نشست جدید برای کاربر شروع شود.

2. require\_once("config.php");: این خط کد، فایل config.php را وارد می‌کند. این فایل معمولاً شامل تنظیمات پایگاه داده و توابع کمکی دیگر است که برای اجرای صحیح برنامه لازم است.

3. if(isset($\_POST['email'])){...}: در این بخش، یک شرط اولیه برای بررسی اینکه آیا فرم ورود کاربر پر شده است یا خیر، قرار دارد. اگر فرم پر شده باشد، اطلاعات وارد شده توسط کاربر (ایمیل و رمز عبور) گرفته و سپس در پایگاه داده جستجو می‌شود.

4. $query = "SELECT \* FROM user WHERE email='$email' AND password='$password'";: در این خط، یک کوئری SQL برای جستجوی کاربر بر اساس ایمیل و رمز عبور وارد شده توسط کاربر ایجاد می‌شود.

5. $result = mysqli\_query($conn,$query);: در این خط، کوئری SQL به پایگاه داده ارسال شده و نتایج بازگشتی در متغیر $result ذخیره می‌شود.

6. if(mysqli\_num\_rows($result) > 0){...}else{...}: در این بخش، نتایج بازگشتی از پایگاه داده بررسی می‌شود. اگر کاربر با اطلاعات وارد شده یافت شود، اطلاعات کاربر (ایمیل و نام) در نشست ذخیره می‌شود و کاربر به صفحه اصلی فروشگاه هدایت می‌شود. در غیر اینصورت، کاربر به صفحه ورود با پیام خطا هدایت می‌شود.

7. if(isset($\_GET['id\_cart'])){...}: در این بخش، چک می‌شود که آیا پارامتر id\_cart در URL وجود دارد یا خیر. اگر وجود داشته باشد، محصول مورد نظر به سبد خرید کاربر اضافه می‌شود.

به طور کلی، این قسمت از کد PHP مسئول بخش‌های مختلف از فروشگاه آنلاین است، شامل ورود کاربران، ثبت سفارشات در سبد خرید و ... می‌باشد، که در اول بقیه فایل ها include کردیم و از آن استفاده کردیم.

Login.php

1. session\_start();: در این خط، یک نشست جدید برای کاربر شروع می‌شود. اگر نشست قبلاً شروع شده باشد، از آن استفاده می‌شود.

2. require\_once("config.php");: این خط کد، فایل config.php را وارد می‌کند. این فایل معمولاً شامل تنظیمات پایگاه داده و توابع کمکی دیگر است که برای اجرای صحیح برنامه لازم است.

3. if(isset($\_POST['email'])){...}: در این بخش، یک شرط اولیه برای بررسی اینکه آیا فرم ورود کاربر پر شده است یا خیر، قرار دارد. اگر فرم پر شده باشد، اطلاعات وارد شده توسط کاربر (ایمیل و رمز عبور) گرفته و سپس در پایگاه داده جستجو می‌شود.

4. $query = "SELECT \* FROM user WHERE email='$email' AND password='$password'";: در این خط، یک کوئری SQL برای جستجوی کاربر بر اساس ایمیل و رمز عبور وارد شده توسط کاربر ایجاد می‌شود.

5. $result = mysqli\_query($conn,$query);: در این خط، کوئری SQL به پایگاه داده ارسال شده و نتایج بازگشتی در متغیر $result ذخیره می‌شود.

6. $row = mysqli\_fetch\_assoc($result);: در این خط، نتیجه بازگشتی از پایگاه داده به صورت یک ردیف (row) ذخیره می‌شود.

7. if(mysqli\_num\_rows($result) > 0){...}else{...}: در این بخش، نتایج بازگشتی از پایگاه داده بررسی می‌شود. اگر کاربر با اطلاعات وارد شده یافت شود، اطلاعات کاربر (ایمیل و نام) در نشست ذخیره می‌شود و کاربر به صفحه اصلی فروشگاه هدایت می‌شود. در غیر اینصورت، کاربر به صفحه ورود با پیام خطا هدایت می‌شود.

8. if(isset($\_GET['email'])){...}: در این بخش، چک می‌شود که آیا پارامتر email در URL وجود دارد یا خیر. اگر وجود داشته باشد، این پارامتر در متغیر $emilGet ذخیره می‌شود.

به طور کلی، این قسمت از کد PHP مسئول بخش‌های مختلف از فروشگاه آنلاین است، شامل ورود کاربران، ثبت سفارشات در سبد خرید و ... می‌باشد.

Logout.php

1. session\_start();: در این خط، یک نشست جدید برای کاربر شروع می‌شود. اگر نشست قبلاً شروع شده باشد، از آن استفاده می‌شود.

2. session\_destroy();: این دستور باعث حذف تمامی متغیرهای مربوط به نشست فعلی کاربر می‌شود و نشست را تماماً پاک می‌کند.

3. header("location: index.php");: این دستور به مرورگر می‌گوید که به آدرس index.php هدایت شود. به این ترتیب، پس از حذف نشست، کاربر به صفحه اصلی وبسایت فروشگاه هدایت می‌شود.

به طور کلی، این کد PHP در فایل logout.php وظیفه حذف نشست کاربر (خروج از حساب کاربری) و هدایت کاربر به صفحه اصلی وبسایت را بر عهده دارد.

Register.php

1. session\_start();: در این خط، یک نشست جدید برای کاربر شروع می‌شود. اگر نشست قبلاً شروع شده باشد، از آن استفاده می‌شود.

2. require\_once ('config.php');: این دستور به وبسایت ماژول config.php را وارد می‌کند که شامل تنظیمات اتصال به پایگاه داده است.

3. if(isset($\_POST['email'])){...}: این شرط بررسی می‌کند که آیا اطلاعات فرم ثبت نام وارد شده است یا خیر.

4. در خطوط 4 تا 7، اطلاعات وارد شده از فرم ثبت نام (ایمیل، رمز عبور، نام، نام خانوادگی، آدرس و زمان ثبت نام) در متغیرهای مختلف ذخیره می‌شوند.

5. $selectEmail = "SELECT \* FROM user WHERE email='$email'";: در این خط، یک کوئری SQL برای جستجوی ایمیل وارد شده در جدول کاربران انجام می‌شود.

6. $exist = mysqli\_query($conn,$selectEmail);: این دستور کوئری را اجرا کرده و نتیجه آن را در متغیر $exist ذخیره می‌کند.

7. if(mysqli\_num\_rows($exist) > 0){...}: این شرط بررسی می‌کند که آیا ایمیل وارد شده قبلاً در پایگاه داده وجود دارد یا خیر. اگر وجود داشته باشد، کاربر به صفحه ثبت نام با پیام خطا هدایت می‌شود.

8. $query = "INSERT INTO user(email,password,firstname,lastname,address,registerTime) VALUES ('$email','$password','$firstname','$lastname','$address','$registerTime')";: در این خط، یک کوئری SQL برای ثبت اطلاعات جدید کاربر در جدول کاربران ایجاد می‌شود.

9. if(mysqli\_query($conn,$query)){...}: این شرط بررسی می‌کند که آیا عملیات ثبت نام با موفقیت انجام شده است یا خیر. اگر موفق بود، اطلاعات کاربر در نشست ذخیره شده و کاربر به صفحه اصلی هدایت می‌شود.

10. header("location: register.php?error=1");: اگر هر گونه خطایی در عملیات ثبت نام رخ داد، کاربر به صفحه ثبت نام با پیام خطا هدایت می‌شود.

به طور کلی، این کد PHP در فایل register.php وظیفه ثبت نام کاربران جدید را بر عهده دارد و با استفاده از پایگاه داده، اطلاعات جدید کاربر را ثبت می‌کند و کاربر را به صفحه مناسب هدایت می‌کند.

Single.php

1. session\_start();: در این خط، یک نشست جدید برای کاربر شروع می‌شود. اگر نشست قبلاً شروع شده باشد، از آن استفاده می‌شود.

2. require\_once("config.php");: این دستور به وبسایت ماژول config.php را وارد می‌کند که شامل تنظیمات اتصال به پایگاه داده است.

3. if(isset($\_GET['p\_id'])){...}: این شرط بررسی می‌کند که آیا شناسه محصول (p\_id) در URL وجود دارد یا خیر.

4. $query = "SELECT \* FROM products WHERE product\_id=".$\_GET['p\_id'];: در این خط، یک کوئری SQL برای بازیابی اطلاعات محصول مورد نظر بر اساس شناسه محصول (p\_id) از جدول محصولات انجام می‌شود.

5. $result = mysqli\_query($conn,$query);: این دستور کوئری را اجرا کرده و نتیجه آن را در متغیر $result ذخیره می‌کند.

6. $row = mysqli\_fetch\_assoc($result);: در این خط، اطلاعات بازیابی شده از پایگاه داده به صورت یک ردیف (آرایه انجمنی) در متغیر $row ذخیره می‌شود.

به طور کلی، این کد PHP در فایل single.php وظیفه نمایش اطلاعات محصول مورد نظر را بر عهده دارد. ابتدا با استفاده از شناسه محصول، اطلاعات محصول را از پایگاه داده بازیابی کرده و سپس در قالب یک ردیف (آرایه انجمنی) آن را به صفحه نمایش محصول نمایش می‌دهد.