

In class Exercise

Developing a WEBP3001 - E commerce website

Using XAMPP, MYSQL, PHP

Part 2

Showing Product Details

We would like to show additional product details, like description and different features. All the information about the product—including its code, name, price, description, and so on—is stored in the products table. The product features are stored in a separate table called *productfeatures*. The *itemdetails.php* script, shown below.

```
<?php
include('topmenu.php');
$connect = mysqli_connect("localhost", "root", "", "shopping") or
die("Please, check your server connection.");
$code=$_REQUEST['itemcode'];
$query = "SELECT item_code, item_name, description, imagename, price FROM
products " .
"where item_code like '$code'";
$results = mysqli_query($connect, $query) or die(mysqli_error($connect)); // #1
$row = mysqli_fetch_array($results, MYSQLI_ASSOC);
extract($row);
echo "<table width=\"100%\" cellpadding=\"5\">";
echo "<tr><td style=\"padding: 3px;\" rowspan=\"6\">";
echo "<img src=images/'. $imagename . ' style=\"max-width:200px;max-height:260px;
width:auto;height:auto;\"></img></td>";
echo "<td colspan=\"2\" align=\"center\" style=\"font-family:verdana;
font-size:150%;\"><b>";
echo $item_name;
echo "</b></td></tr><tr><td colspan=\"2\"><table><tr><td>";
$itemname=urlencode($item_name);
$itemprice=$price;
$itemdescription=$description;
$pfquery = "SELECT feature1, feature2, feature3, feature4, feature5,
feature6 FROM productfeatures " .
"where item_code like '$code'"; // #2
$pfresults = mysqli_query($connect, $pfquery) or die(mysqli_error($connect));
$pfrow = mysqli_fetch_array($pfresults, MYSQLI_ASSOC);
extract($pfrow);
echo $feature1;
echo "</td><td>";
```


type of object along with the custom objects. Session data is stored separately for every client.

HTTP is a stateless protocol, to keep track of the user, the session ID needs to be continually correlated with the user. This correlation is done through the following methods:

Cookies—When the client connects to the server for the first time, the server drops the cookies on the client's machine. In every subsequent request, the server uses the cookies to recognize the user and his settings.

URL Rewriting—The session ID is often appended to the query string of the URL,

<http://www.webp3001.com/productdetails.php;jsessionid=2243781FG55544K1>

Hidden fields—A hidden field is embedded into the session ID within a web form. The session ID is extracted from the hidden field to recognize the user.

Functions Used in Session Handling

session is a combination of a server-side file containing all the data you want to store and a client-side cookie containing a reference to the server data. The file and the client-side cookie are created using the `session_start()` function.

session_start()

`session_start()` function initializes the session data. It creates a session or resumes the current one based on the current session ID. The syntax for using this function is: `bool session_start (void)` This function always returns true. If you want to use a named session, you must call the `session_name()` function before calling `session_start()`. When `session_start()` is called, PHP checks to see whether the visitor sent a session cookie. If it did, PHP loads the session data. Otherwise, PHP creates a new session file on the server and sends an ID back to the visitor to associate the visitor with the new file. Because each visitor has her own data locked away in a unique session file, you need to call `session_start()` every time before you try to read session variables. As `session_start()` needs to send the reference cookie to the user's computer.

session_id()

The `session_id()` function gets or sets the session ID for the current session. More precisely, the `session_id()` function returns either the session ID for the current session

or an empty string ("") if there is no current session (no current session ID exists). The syntax is as follows:

`string session_id ([string $id])` When the id parameter is not supplied, the function returns the session ID of the current session. If the id parameter is supplied, the function replaces the current session with the session whose ID is given. In that case, the `session_id()` function must be called before `session_start()` is invoked.

sessionscript1.php

```
<?php
if (session_status() == PHP_SESSION_NONE) { // #1
    session_start();
}
?>
<!DOCTYPE html>
<html>
<body>
<?php
$_SESSION["username"] = "John"; // #2
$_SESSION["cartquantity"] = 3;
$_SESSION["cartprice"] = 19.99;
?>
Finished with shopping? <br>
Click <a href="sessionscript2.php"> View Cart </a> link to view the quantity
and amount of the products selected in the cart
</body>
</html>
```

- #1 ensures that if the session is not yet started, it invokes the `session_start()` function to create a new session.
- #2 sets the username variable to the value John. Similarly, the following two statements set the cartquantity and cartprice variables to 3 and 19.99, respectively.
- After setting the session variables, the script shows a hyperlink called View Cart, which will navigate the user to another script, called sessionscript2.php.

sessionscript2.php script completes the following tasks:

- Checks whether a session exists. If not, it starts a new session.
- Checks if the username variable is set and displays a welcome message.
- Retrieves the values in the session variables, cartquantity and cartprice, which were set in the sessionscript1.php script, and displays them on the screen.

```

<?php
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
?>
<html>
    <body>
        <?php
            if (isset($_SESSION['username']))
                $username=$_SESSION["username"];
            else
                $username="Sir/Ma'm";
            $cartquantity=$_SESSION["cartquantity"];
            $cartprice=$_SESSION["cartprice"];
            echo "Session is On and the session id is " . session_id() . "<br>";
            echo "Welcome $username. <br>";
            echo "There are $cartquantity products chosen in the cart worth
            $$cartprice";
        ?>
    </body>
</html>

```

- Display the username of the user who is signed in or display the Login link that will enable a user to sign in.
- Display the Signup link, which will display a form that enables the user to register on your site. To do these tasks, modify the topmenu.php

topmenu.php

```

<!DOCTYPE html>
<head>
<meta charset="utf-8">
<title>Bintu Online Bazar</title>
<link rel="stylesheet" href="css/style.css">
</head>
<body>
<table width="100%" cellpadding="2">
<tr>
<td style="width:30%">
<td style="width:40%">
<td style="width:20%">
<tr>
<td style="background-color:cyan;color:Blue;"><td><td>
style="background-color:cyan;color:Blue;"><td><td>
style="background-color:cyan;color:Blue;">
<?php
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
echo "<span id='userinfo'><a href='signin.php'>Login</a>&nbsp;&nbsp;&nbsp;
&nbsp;&nbsp;&nbsp;<a href='validatesignup.php'>Signup</a></span></td></tr>";
?>
<tr><td style="font-size: 35px;color:blue;background-color:cyan;">
<b>Bintu Online Bazar</b></font></td>
<td bgcolor="cyan">
<form method="post" action="searchitems.php">

```

```

<input size="50" type="text" name="tosearch">
<input type="submit" name="submit" value="Search">
</form></td>
<td bgcolor="cyan" ><a href="cart.php"></img>
<span id="cartcountinfo"></span></a></td></tr>
</table>
<div class="container">
<nav>
<ul class="nav">
<li><a href="index.php">Home</a></li>
<li class="dropdown"><a href="index.php">Electronics</a>
<ul>
<li><a href="itemlist.php?category=CellPhone">Smart Phones</a></li>
<li><a href="itemlist.php?category=Laptop">Laptops</a></li>
<li><a href="index.php">Cameras </a></li>
<li><a href="index.php">Televisions</a></li>
</ul>
</li>
<li class="dropdown">
<a href="index.php">Home & Furniture</a>
<ul class="large">
<li><a href="index.php">Kitchen Essentials</a></li>
<li><a href="index.php">Bath Essentials</a></li>
<li><a href="index.php">Furniture</a></li>
<li><a href="index.php">Dining & Serving</a></li>
<li><a href="index.php">Cookware</a></li>
</ul>
</li>
<li><a href="index.php">Books</a></li>
</ul>
</nav>
</div>
<p>

```

Signing In and Out

signin.php script displays a sign-in page along with the header and drop-down menu.

```

<?php
include('topmenu.php');
?>
<html>
<head>
</head>
<body>
<form action="validateuser.php" method="post">
<table border="0" cellspacing="1" cellpadding="3">
<tr><td>Email Address:</td><td><input type="text" name="emailaddress">
</td></tr>
<tr><td>Password:</td><td><input type="password" name="password"></td></tr>
<tr><td colspan="2" align="center"><input type="submit" name="submit"
value="Login"></td></tr>
</table>
</form>
</body>

```

```
</html>
```

Update validateuser.php script to display information about the signed-in user in the site's header

```
<html>  
<head>  
<script language="JavaScript" type="text/JavaScript">  
function updateUser(username){ // #1  
var ajaxUser = document.getElementById("userinfo"); // #2  
ajaxUser.innerHTML = "Welcome " + username + "&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&nbsp;&";  
<a href="\logout.php\">Log Out</a>;  
}  
</script>  
</head>  
<body>  
<?php  
include('topmenu.php');  
if (session_status() == PHP_SESSION_NONE) {  
session_start();  
}  
$connect = mysqli_connect("localhost", "root", "gold", "shopping") or  
die("Please, check your server connection.");  
$query = "SELECT email_address, password, complete_name FROM customers WHERE  
email_address like '" . $_POST['emailaddress'] . "' " .  
"AND password like (PASSWORD('" . $_POST['password'] . "'))";  
$result = mysqli_query($connect, $query) or die(mysqli_error()); // #3  
if (mysqli_num_rows($result) == 1) {  
while ($row = mysqli_fetch_array($result, MYSQLI_ASSOC)) {  
extract($row);  
echo "Welcome " . $complete_name . " to our Shopping Mall <br>"; // #4  
$_SESSION['emailaddress'] = $_POST['emailaddress'];  
$_SESSION['password'] = $_POST['password'];  
echo "<SCRIPT LANGUAGE=\"JavaScript\">updateUser('$complete_name')";  
</SCRIPT>"; // #5  
}  
}  
else {  
?>  
Invalid Email address and/or Password<br> // #6  
Not registered?  
<a href="validatesignup.php">Click here</a> to register.<br><br><br>  
Want to Try again<br>  
<a href="signin.php">Click here</a> to try login again.<br>  
<?php  
}  
?>  
</body>  
</html>
```

#1 defines a JavaScript function called `updateUser` that is invoked when users enter a valid e-mail address and password.

#2 displays a welcome message along with the user's name in the span ID called `userinfo`. It also displays a link, called `Log Out`, which navigates the users to the `logout.php` script and enables them to log out of your site.

#3 executes a SQL query that determines whether any row exists in the `customers` table that contains the e-mail address and password that was entered by the user in the `signin.php` script.

#4 displays a welcome message along with the user's name if such a row exists.

#5 invokes the JavaScript function, `updateUser`, to display a welcome message in the header of the site.

#6 and on display an error message and links that enable the user to try to sign in again or create a new account.

`logout.php` script that destroys users' sessions so they can log out.

Home Page of the Site

The home page of this e-commerce site displays the site header, a drop-down menu, and images of some of the products that the site sells. The product images being displayed will fade out after a few seconds and will be replaced with others. The process continues to display different images. Upon clicking any image, users will be navigated to the page that shows the complete list of that type of product. For example, when users click on any laptop image, they will be navigated to the page that shows the complete list of all the laptops that are available for sale.

```
<?php
include('topmenu.php');
?>
<span id="crossfade">
<a href="itemlist.php?category=CellPhone">

</a>
</span>
<span id="crossfade">
<a href="itemlist.php?category=CellPhone">
<img class="bottom" src="images/MicrosoftLumia640XL.jpg"
```



```

style="max-width:350px;max-height:350px;width:auto; height: auto;" />
</a>
</span>
<span id="crossfade">
<a href="itemlist.php?category=Laptop">

</a>
</span>
</body>
</html>

```

Update style.css by adding the following

```

#crossfade {
    position:relative;
    height:350px;
    width:350px;
    margin-right:250px;
}
#crossfade img {
    position:absolute;
    left:0;
    -webkit-transition: opacity 1s ease-in-out;
    -moz-transition: opacity 1s ease-in-out;
    -o-transition: opacity 1s ease-in-out;
    transition: opacity 1s ease-in-out;
}
@keyframes crossfadeFadeInOut {
    0% {
        opacity:1;
    }
    45% {
        opacity:1;
    }
    55% {
        opacity:0;
    }
    100% {
        opacity:0;
    }
}
#crossfade img.top {
    animation-name: crossfadeFadeInOut;
    animation-timing-function: ease-in-out;
    animation-iteration-count: infinite;
    animation-duration: 5s;
    animation-direction: alternate;
}

```

Shopping Cart

Shopping cart is a software module that shows the products chosen by the visitor for purchase. The items in the cart can be revised if required, i.e., after adding the products to the cart, users can remove them from the cart, add more items to the cart, and increase the quantity of the products already selected in the cart.

A cart is a database table that temporarily keeps the user's items selected for purchase. To store and manage items in the cart, a database table called cart was created with the structure.

Saving Selections in the Cart

itemdetails.php script displays the detailed information about the selected product and shows the following two buttons beneath the information:

- Buy Now – Adds the item to the cart and proceeds to checkout.
- Add To Cart – Adds the item to the cart and remains there, this enabling the user to choose more products.

```
<?php
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
$connect = mysqli_connect("localhost", "root", "gold", "shopping") or
die("Please, check your server connection.");
$message = "";
$quantity="";
$action="";
$query="";
if (isset($_POST['quantity'])) {
    $quantity = trim($_POST['quantity']);
}
if ($quantity=='')
{
    $quantity=1;
}
if($quantity <=0)
{
    echo "Quantity value is invalid ";
    echo "Go Back and enter a valid value";
}
else
{
    if (isset($_REQUEST['icode'])) {
```

```

$itemcode = $_REQUEST['icode'];
}
if (isset($_REQUEST['iname'])) {
$item_name = $_REQUEST['iname'];
}
if (isset($_REQUEST['iprice'])) {
$price = $_REQUEST['iprice'];
}
if (isset($_POST['modified_quantity'])) {
$modified_quantity = $_POST['modified_quantity'];
}
$sess = session_id();
if (isset($_REQUEST['action'])) {
$action = $_REQUEST['action'];
}

switch ($action) {
case "add":
$query="select * from cart where cart_sess = '$sess' and cart_itemcode like '$itemcode'";
$result = mysqli_query($connect, $query) or die(mysql_error());
if(mysqli_num_rows($result)==1)
{
$row=mysqli_fetch_array($result, MYSQLI_ASSOC);
$qt=$row['cart_quantity'];
$qt=$qt + $quantity;
$query="UPDATE cart set cart_quantity=$qt where cart_sess = '$sess' and cart_itemcode like '$itemcode'";
$result = mysqli_query($connect, $query) or die(mysql_error());
}
else
{
$query = "INSERT INTO cart (cart_sess, cart_quantity, cart_itemcode, cart_item_name, cart_price) VALUES ('$sess', $quantity, '$itemcode', '$item_name', $price)";
$message = "<div align=\"center\"><strong>Item added.</strong></div>";
}
break;
case "change":
if($modified_quantity==0)
{
$query = "DELETE FROM cart WHERE cart_sess = '$sess' and cart_itemcode like '$itemcode'";
$message = "<div style=\"width:200px; margin:auto;\">Item deleted</div>";
}
else
{
if($modified_quantity <0)
{
echo "Invalid quantity entered";
}
else
{
$query = "UPDATE cart SET cart_quantity = $modified_quantity WHERE cart_sess = '$sess' and cart_itemcode like '$itemcode'";
$message = "<div style=\"width:200px; margin:auto;\">Quantity changed</div>";
}
}
break;
case "delete":

```

```

$query = "DELETE FROM cart WHERE cart_sess = '$sess' and cart_itemcode like
'$itemcode'";

$message = "<div style=\"width:200px; margin:auto;\">Item deleted</div>";
break;
case "empty":
$query = "DELETE FROM cart WHERE cart_sess = '$sess'";
break;
}
if($query != "")
{
    $results = mysqli_query($connect, $query) or die(mysql_error());
    echo $message;
}
include("showcart.php");
echo "<SCRIPT LANGUAGE=\"JavaScript\">updateCart();</SCRIPT>";
}
?>

```

session ID will be used to remember the items selected in the cart by the particular user. Remember that the status of the session is equal to PHP_SESSION_NONE if the session is enabled and none exists

Connection to the MySQL server is established and the shopping database file is selected.

The quantity of the item purchased (stored in the quantity variable) is retrieved from the \$_POST array. If the user has not specified a quantity, the default is one. It also verifies that the value of the quantity entered is a non-negative number. The item code, item name, and price (stored in the icode, iname, and iprice variables) sent from the itemdetails.

php script are retrieved using the \$_REQUEST array and are assigned to the \$itemcode, \$item_name, and \$price variables.

Maintaining the Cart

showcart.php Script Displays the Content in the Cart Table and Maintains It

```

<?php
if ( ! isset($totalamount)) {
$totalamount=0;
}
$totalquantity=0;
if (!session_id()) {
session_start();
}
$connect = mysqli_connect("localhost", "root", "", "shopping") or
die("Please, check your server connection.");
$sessid = session_id();
$query = "SELECT * FROM cart WHERE cart_sess = '$sessid'";

```

```

$results = mysqli_query($connect, $query) or die (mysqli_query($connect, $query));
if(mysqli_num_rows($results)==0)
{
echo "<div style=\"width:200px; margin:auto;\">Your Cart is empty</div> ";
}
else
{
?>
<table border="1" align="center" cellpadding="5">
<tr><td> Item Code</td><td>Quantity</td><td>Item Name</td><td>Price</td><td>Total Price</td>
<?php
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
extract($row);
echo "<tr><td>";
echo $cart_itemcode;
echo "</td>";
echo "<td><form method=\"POST\" action=\"cart.php?action=change&icode=
$cart_itemcode\"><input type=\"text\" name=\"modified_quantity\" size=\"2\"
value=\"$cart_quantity\">";
echo "</td><td>";
echo $cart_item_name;
echo "</td><td>";
echo $cart_price;
echo "</td><td>";
$totalquantity = $totalquantity + $cart_quantity;
$totalprice = number_format($cart_price * $cart_quantity, 2);
$totalamount=$totalamount + ($cart_price * $cart_quantity);
echo $totalprice;
echo "</td><td>";
echo "<input type=\"submit\" name=\"Submit\" value=\"Change quantity\">
</form></td>";
echo "<td>";
echo "<form method=\"POST\" action=\"cart.php?action=delete&icode=$cart_
itemcode\">";
echo "<input type=\"submit\" name=\"Submit\" value=\"Delete Item\"></form>
</td></tr>";
}
echo "<tr><td>Total</td><td>$totalquantity</td><td></td><td></td><td>";
$totalamount = number_format($totalamount, 2);
echo $totalamount;
echo "</td></tr>";
echo "</table><br>";

echo "<div style=\"width:400px; margin:auto;\">You currently have " .
$totalquantity . " product(s) selected in your cart</div> ";
?>
<table border="0" style="margin:auto;">
<tr><td style="padding: 10px;">
<form method="POST" action="cart.php?action=empty">
<input type="submit" name="Submit" value="Empty Cart"
style="font-family:verdana; font-size:150%;" >
</form>
</td><td>
<form method="POST" action="checklogin.php">
<input id="cartamount" name="cartamount" type="hidden" value= "<?php echo
$totalamount ; ?>">
<input type="submit" name="Submit" value="Checkout"
style="font-family:verdana; font-size:150%;" >
</form>
</td></tr></table>

```

```
<?php
}
?>
</body>
</html>
```

Displaying the Cart Count in the Site Header Using AJAX

To display the count of the items selected in the cart and the visitor information in the site header, you'll use AJAX.

AJAX stands for Asynchronous JavaScript and XML. It is an umbrella term used for creating dynamic and highly responsive web pages. Usually, in a web application, partial updating of a web page is not possible.

Example,

When the user prompts for some information from the database on the server, the entire web page is refreshed to display the fetched information. Even if the fetched information is meant to be displayed in a small region of the web page, the entire page is reloaded.

AJAX,

- Only the region that is supposed to display the fetched information is refreshed, which makes it highly responsive.
- The data is sent to and accessed from the server in the background. This makes it much faster at displaying responses.

AJAX uses the XMLHttpRequest object and JavaScript to communicate with the server and to display data on web pages, respectively.

Revise `topmenu.php` to indicate the cart content and user signed in status.

```
!DOCTYPE html>
<head>
<meta charset="utf-8">
<title>Bintu Online Bazar</title>
<link rel="stylesheet" href="css/style.css">
<script language="JavaScript" type="text/JavaScript">
function makeRequestObject(){
var xmlhttp=false;
try {
xmlhttp = new XMLHttpRequest('Msxml2.XMLHTTP'); // #1
```

```

} catch (e) {
try {
xmlhttp = new
ActiveXObject('Microsoft.XMLHTTP'); // #2
} catch (E) {
xmlhttp = false;
}
}
if (!xmlhttp && typeof XMLHttpRequest!='undefined') {
xmlhttp = new XMLHttpRequest(); // #3
}
return xmlhttp;
}
function updateCart(){ // #4
var xmlhttp=makeRequestObject();
xmlhttp.open('GET', 'countcart.php', true); // #5
xmlhttp.onreadystatechange = function(){ // #6
if (xmlhttp.readyState == 4 && xmlhttp.status == 200) { // #7
var ajaxCart = document.getElementById("cartcountinfo"); // #8
ajaxCart.innerHTML = xmlhttp.responseText;
}
}
xmlhttp.send(null);
}
</script>
</head>
<body>
<table width="100%" cellpadding="2">
<tr>
<td style="width:30%">
<td style="width:40%">
<td style="width:20%">
<tr>
<td style="background-color:cyan;color:Blue;"></td>
<td style="background-color:cyan;color:Blue;"></td>
<td style="background-color:cyan;color:Blue;">
<?php
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
if (isset($_SESSION['emailaddress']))
{
    echo "Welcome " . $_SESSION['emailaddress'] . "&nbsp;&nbsp;&nbsp;"; // #9
    echo "<a href='\"logout.php\"'>Log Out</a></td></tr>";
}
else
{
    echo "<a href='\"signin.php\"'>Login</a>&nbsp;&nbsp;&nbsp;";
    echo "<a href='\"validatesignup.php\"'>Signup</a></td></tr>";
}
?>
<tr>
<td style="font-size: 35px;color:blue;background-color:cyan;">
<b> Bintu Online Bazar</b></td>
<td bgcolor="cyan">
<form method="post" action="searchitems.php">
<input size="50" type="text" name="tosearch">
<input type="submit" name="submit" value="Search">
</form></td>
<td bgcolor="cyan" ><a href="cart.php"></img>
<span id="cartcountinfo"></span></a>
</td></tr>
</table>

```

```

<div class="container">
  <nav>
    <ul class="nav">
      <li><a href="index.php">Home</a></li>
      <li class="dropdown">
        <a href="index.php">Electronics</a>
        <ul>
          <li><a href="itemlist.php?category=CellPhone">Smart Phones</a></li>
          <li><a href="itemlist.php?category=Laptop">Laptops</a></li>
          <li><a href="index.php">Cameras </a></li>
          <li><a href="index.php">Televisions</a></li>
        </ul>
      </li>
      <li class="dropdown">
        <a href="index.php">Home & Furniture</a>
        <ul class="large">
          <li><a href="index.php">Kitchen Essentials</a></li>
          <li><a href="index.php">Bath Essentials</a></li>
          <li><a href="index.php">Furniture</a></li>
          <li><a href="index.php">Dining & Serving</a></li>
          <li><a href="index.php">Cookware</a></li>
        </ul>
      </li>
      <li><a href="index.php">Books</a></li>
    </ul>
  </nav>
</div>
<p>

```

#1 creates an XMLHttpRequest object. The XMLHttpRequest object enables the JavaScript code to make asynchronous HTTP server requests. It is through using this object that you can make HTTP requests, receive responses, and update a region of the page completely in the background.

#2 executes ActiveXObject("Microsoft.XMLHTTP"), assuming the browser is IE5.0. If all these fail, the built-in function XMLHttpRequest() in statement

#3 is used to create an XMLHttpRequest object.

#4 defines a JavaScript function that displays the count of the items in the cart. This function is called whenever the user makes changes to the cart, i.e., whenever any item is added, modified, or deleted from the cart

#5 makes a request to the server with the GET method and passes the file name, countcart.php, to be executed on the server

countcart.php File Counts and Returns the Number of Items Selected in the Cart

```
<?php
$totalquantity=0;
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
$connect = mysqli_connect("localhost", "root", "gold", "shopping") or
die("Please, check your server connection.");
$sess = session_id();
$query="select * from cart where cart_sess = '$sess'";
$results = mysqli_query($connect, $query) or die(mysql_error());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
    extract($row);
    $totalquantity = $totalquantity + $cart_quantity;
}
echo $totalquantity;
?>
```

Check Out

checklogin.php Script Checks the Login Status of the Visitor

```
<?php
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
$connect = mysqli_connect("localhost", "root", "", "shopping") or
die("Please, check your server connection.");
$cartamount=0;
$cartamount = $_POST['cartamount'];
$_SESSION['cartamount']=$cartamount;
if (isset($_SESSION['emailaddress']))
{
    $email_address=$_SESSION['emailaddress'];
    echo "Welcome " . $email_address . ". <br/>";
}
if (isset($_SESSION['password']))
{
    $password=$_SESSION['password'];
}
if ((isset($_SESSION['emailaddress']) && $_SESSION['emailaddress'] != "") ||
(isset($_SESSION['password']) && $_SESSION['password'] != "")) {
    $sess = session_id();
    $query="select * from cart where cart_sess = '$sess'";
    $result = mysqli_query($connect, $query) or die(mysqli_error($connect));
    if(mysqli_num_rows($result)>=1)
    {
        echo "If you have finished Shopping ";
        echo "<a href=shipping_info.php>Click Here</a> to supply Shipping
        Information";
        echo " Or You can do more purchasing by selecting items from the menu ";
    }
    else
```

```

{
echo "You can do purchasing by selecting items from the menu on left side";
}
}
else
{
?>
<html>
<head>
</head>
<body>
<h3>Not Logged in yet</h1>
<p>
You are currently not logged into our system.<br>
You can do purchasing only if you are logged in.<br>
If you have already registered,
<a href="signin.php">click here</a> to login, or if would like to create an
account, <a href="create_account.php">click here</a> to register.
</p>
</body>
</html>
<?php
}
?>

```

Shipping Information

shipping_info.php Script Enables Entering of Shipping Information for Product Delivery

```

<?php
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) {
session_start();
}
if (isset($_SESSION['cartamount']))
{
$cartamount=$_SESSION['cartamount'];
}
$connect = mysqli_connect("localhost", "root", "", "shopping") or
die("Please, check your server connection.");
$email_address="";
if (isset($_SESSION['emailaddress']))
{
$email_address=$_SESSION['emailaddress'];
}
if (isset($_SESSION['password']))
{
$password=$_SESSION['password'];
}
if ((isset($_SESSION['emailaddress']) && $_SESSION['emailaddress'] != "") ||
(isset($_SESSION['password']) && $_SESSION['password'] != "")) {
$query = "SELECT * FROM customers where email_address like '$email_address'
and password like (PASSWORD('$password'))";
$results = mysqli_query($connect, $query) or die(mysql_error());
$row = mysqli_fetch_array($results, MYSQLI_ASSOC);
extract($row);
}

```

```

?>
<form action="purchase.php" method="post">
<table border="0" cellspacing="1" cellpadding="3">
<tr><td colspan="2" align="center">Your information available with us:</td></tr>
<tr><td>Email Address:</td><td><input size="20" type="text" name="email_address" value="<?php echo $email_address; ?>"></td></tr>
<tr><td>Complete Name: </td><td><input size="50" type="text" name="complete_name" value="<?php echo $complete_name; ?>"></td></tr>
<tr><td>Address: </td><td><input size="80" type="text" name="address1" value="<?php echo $address_line1; ?>"></td></tr>
<tr><td></td><td><input size="80" type="text" name="address2" value="<?php echo $address_line2; ?>"></td></tr>
<tr><td>City: </td><td><input size="30" type="text" name="city" value="<?php echo $city; ?>"></td></tr>
<tr><td>State: </td><td><input size="30" type="text" name="state" value="<?php echo $state; ?>"></td></tr>
<tr><td>Country: </td><td><input size="30" type="text" name="country" value="<?php echo $country; ?>"></td></tr>
<tr><td>Zip Code: </td><td><input size="20" type="text" name="zipcode" value="<?php echo $zipcode; ?>"></td></tr>
<tr><td>Phone No: </td><td><input size="30" type="text" name="phone_no" value="<?php echo $cellphone_no; ?>"></td></tr>
<tr><td colspan="2" align="center">Please update shipping information if different from the shown below: </td></tr>
<tr><td> Address to deliver at: </td><td><input type="text" size="80" name="shipping_address_line1" value="<?php echo $address_line1; ?>"></td></tr>
<tr><td></td><td><input type="text" size="80" name="shipping_address_line2" value="<?php echo $address_line2; ?>"></td></tr>
<tr><td> City: </td><td><input size="30" type="text" name="shipping_city" value="<?php echo $city; ?>"></td></tr>
<tr><td> State: </td><td><input size="30" type="text" name="shipping_state" value="<?php echo $state; ?>"></td></tr>
<tr><td> Country: </td><td><input size="30" type="text" name="shipping_country" value="<?php echo $country; ?>"></td></tr>
<tr><td> Zip Code: </td><td><input type="text" size="20" name="shipping_zipcode" value="<?php echo $zipcode; ?>"></td></tr>
<tr><td><input type="submit" name="submit" value="Supply Payment Information"></td><td><input type="reset" value="Cancel"></td></tr>
</table>
</form>
<?php
}
?>
</body>
</html>

```

Different Payment Modes

Credit Cards, Debit Cards, Smart Cards, E-Money, Electronic Fund Transfer, Cash on Delivery Transactions (COD) and internet banking.

-commerce sites collect money from consumers through a service provider known as a payment gateway provider. A payment gateway is an e-commerce application service provider that processes credit card and other card payments. It acts

as a bridge between the consumer who is making purchases and the bank that issued the credit card.

Payment gateways provide a safer platform for money exchanges to take place. In cash on delivery (COD) transactions, payment gateway is not involved because the payments are made directly from the customer to the seller.

Making Payments

JQUERY

jQuery is a lightweight, feature-rich JavaScript library. It makes it easier to apply JavaScript on your site. In fact, many complex tasks like traversing HTML elements, implementing animation, handling events, etc. can be easily applied to your site just by using jQuery methods. Here are a few of jQuery's features:

- jQuery makes it quite easy to select DOM elements and manipulate them as required.
- jQuery is capable of sensing different events on HTML elements and then taking actions accordingly.
- jQuery enables you to apply AJAX to your site, thus making it highly responsive.
- jQuery includes several built-in animation effects that can be directly applied to your site.
- jQuery is supported by most major browsers

The purchase.php script prompts the visitor to supply payment information.

```
HTML>
<HEAD>
<script type="text/javascript" src="https://www.2checkout.com/checkout/
api/2co.min.js"></script>
<script src="//ajax.googleapis.com/ajax/libs/jquery/1.11.0/jquery.min.js"></script>
<script>
var successCallback = function(data) {
var myForm = document.getElementById('payment-form');
// Set the token as the value for the token input
myForm.token.value = data.response.token.token;
myForm.submit();
};
// Called when token creation fails.
var errorCallback = function(data) {
// Retry the token request if ajax call fails
if (data.errorCode === 200) {
call tokenRequest();
} else {
alert(data.errorMsg);
}
}
```

```

};
var tokenRequest = function() {
// Setup token request arguments
var args = {
sellerId: "102626791",
publishableKey: "E0F6517A-CFCF-11E3-8295-A7DD28100996",
ccNo: $("#card-number").val(),
cvv: $("#card-cvc").val(),
expMonth: $("#card-expiry-month").val(),
expYear: $("#card-expiry-year").val()
};
// Make the token request
TCO.requestToken(successCallback, errorCallback, args);
};
$(function() {
// Pull in the public encryption key for our environment
TCO.loadPubKey('production');
$("#payment-form").submit(function(e) {
// Call our token request function
tokenRequest();
// Prevent form from submitting
return false;
});
});
</SCRIPT>
</HEAD>
<BODY>
<?php
if (session_status() == PHP_SESSION_NONE) {
session_start();
}
if (isset($_SESSION['cartamount']))
{
$cartamount=$_SESSION['cartamount'];
}
$complete_name=$_POST['complete_name'];
$address1 = $_POST['address1'];
$city = $_POST['city'];
$state = $_POST['state'];
$zipcode = $_POST['zipcode'];
$country = $_POST['country'];
$shipping_address_line1 = $_POST['shipping_address_line1'];
$shipping_address_line2 = $_POST['shipping_address_line2'];
$shipping_city = $_POST['shipping_city'];
$shipping_state = $_POST['shipping_state'];
$shipping_country = $_POST['shipping_country'];
$shipping_zipcode = $_POST['shipping_zipcode'];
$phone_no= $_POST['phone_no'] ;
$_SESSION['complete_name'] =$complete_name;
$_SESSION['address1'] =$address1;
$_SESSION['city'] =$city;
$_SESSION['state'] =$state;
$_SESSION['zipcode'] =$zipcode;
$_SESSION['country'] =$country;
$_SESSION['shipping_address_line1'] =$shipping_address_line1;
$_SESSION['shipping_address_line2'] =$shipping_address_line2;
$_SESSION['shipping_city'] =$shipping_city;
$_SESSION['shipping_state'] =$shipping_state;
$_SESSION['shipping_country'] =$shipping_country;
$_SESSION['shipping_zipcode'] =$shipping_zipcode;
$_SESSION['phone_no'] =$phone_no;

```

```

?>
<span class="payment-errors"></span>
<form action="placeorder.php" method="POST" id="payment-form" >
<input id="token" name="token" type="hidden" value="">
<table border="0" cellspacing="1" cellpadding="3">
<tr><td colspan="2" align="center">Online Payment Form</td></tr>
<tr><td>Card Number</td><td><input type="text" size="20" autocomplete="off"
id="card-number"></td></tr>
<tr><td>CVC</td><td><input type="text" size="4" autocomplete="off"
id="card-cvc"></td></tr>
<tr><td>Full Name</td><td><input type="text" id="name" size="80"
autocomplete="on"></td></tr>
<tr><td>Expiration (MM/YYYY)</td><td><input type="text" size="2"
id="card-expiry-month"><input type="text" size="4" id="card-expiry-year"></td></tr>
<tr><td>Amount to Pay: </td><td><?php echo $cartamount; ?></td></tr>
<tr><td colspan="2" align="center"><input type="submit" name="submit"
value="Submit" onclick="formSubmit()">
</td></tr>
</table>
</form>
</BODY>
</HTML>

```

placeorder.php Script Places the Order

```

<?php
require_once("lib/Twocheckout.php");
Twocheckout::privateKey('E0F6517A-CFCF-11E3-8295-A7DD28100996');
Twocheckout::sellerId('102626791');
include('topmenu.php');
if (session_status() == PHP_SESSION_NONE) {
    session_start();
}
if (isset($_SESSION['cartamount']))
{
    $cartamount=$_SESSION['cartamount'];
}
$connect = mysqli_connect("localhost", "root", "gold", "shopping") or
die("Please, check your server connection.");
$complete_name=$_SESSION['complete_name'];
$address1 = $_SESSION['address1'];
$city = $_SESSION['city'];
$state = $_SESSION['state'];
$zipcode = $_SESSION['zipcode'];
$country = $_SESSION['country'];
$shipping_address_line1 = $_SESSION['shipping_address_line1'];
$shipping_address_line2 = $_SESSION['shipping_address_line2'];
$shipping_city = $_SESSION['shipping_city'];
$shipping_state = $_SESSION['shipping_state'];
$shipping_country = $_SESSION['shipping_country'];
$shipping_zipcode = $_SESSION['shipping_zipcode'];
$phone_no= $_SESSION['phone_no'] ;
$email_address= $_SESSION['emailaddress'] ;
$today = date("Y-m-d");
$session_id = session_id();
$sql = "INSERT INTO orders (order_date, email_address,
shipping_address_line1, shipping_line_2, shipping_city, shipping_state,
shipping_country, shipping_zipcode)
VALUES ('$today','$email_address','$shipping_address_line1',

```

```

'$shipping_address_line2', '$shipping_city', '$shipping_state',
'$shipping_country', '$shipping_zipcode'"));
$result = mysqli_query($connect, $sql) or die(mysql_error());
$orderid = mysql_insert_id();
try {
$charge = Twocheckout_Charge::auth(array(
"merchantOrderId" => "$orderid",
"token" => $_POST['token'],
"currency" => 'USD',
"total" => '$cartamount',
"billingAddr" => array(
"name" => '$complete_name',
"addrLine1" => '$address1',
"city" => '$city',
"state" => '$state',
"zipCode" => '$zipcode',
"country" => '$country',
"email" => '$email_address',
"phoneNumber" => '$phone_no'
),
"shippingAddr" => array(
"name" => '$complete_name',
"addrLine1" => '$shipping_address_line1',
"city" => '$shipping_city',
"state" => '$shipping_state',
"zipCode" => '$shipping_zipcode',
"country" => '$shipping_country',
"email" => '$email_address',
"phoneNumber" => '$phone_no'
)
), 'array');
if ($charge['response']['responseCode'] == 'APPROVED') {
echo "Thanks for your Order!";
echo "Please, remember your Order number is $orderid<br>";
echo "<h3>Return Parameters:</h3>";
echo "<pre>";
print_r($charge);
echo "</pre>";
$query = "SELECT * FROM cart WHERE cart_sess='$sessid'";
$results = mysqli_query($connect, $query) or die(mysql_error());
while ($row = mysqli_fetch_array($results, MYSQLI_ASSOC)) {
extract($row);
$totalamount=$totalamount + ($cart_price * $cart_quantity);
$sql2 = "INSERT INTO orders_details (order_no, item_code, item_name,
quantity, price)
VALUES ($orderid,$cart_itemcode,'$cart_item_name',
$cart_quantity,$cart_price)";
$insert = mysqli_query($connect, $sql2) or die(mysql_error());
}
$sql2 = "INSERT INTO payment_details (order_no, email_address,
customer_name, payment_type, name_on_card, card_number, expiration_date)
VALUES ($orderid,$cart_itemcode,'$cart_item_name',
$cart_quantity,$cart_price)";
$insertpayment = mysqli_query($connect, $sql2) or die(mysql_error());
$query = "DELETE FROM cart WHERE cart_sess='$sessid'";
$delete = mysqli_query($connect, $query) or die(mysql_error());
session_destroy();
}
} catch (Twocheckout_Error $e) {
print_r($e->getMessage());
}
}

```

?>