**A Project Report**

On

**MOBILE PLAZA**

By

**Name**:Prachi P. Haldekar **Registration no**:2014BCS071

**Name**:Kajol D. Tale **Registration no**:2014BCS073

**Name**:Manisha G. Yedke **Registration no**:2014BCS039

**Guided By**

Miss. Priya B. Bhosle

(Department Of Computer Science And Engineering)



**Department Of Computer Science And Engineering**

**Shri Guru Gobind Singhji Of Institute Of Engineering And Technology,Nanded**

**Year - 2016-2017**

**Shri Guru Gobind Singhji Institute Of Engineering And Technology, Nanded**

**Department Of Computer Science and Engineering**



**CERTIFICATE**

This is to certify that

**Name**:Prachi P. Haldekar **Registration no**:2014BCS071

**Name**:Kajol D. Tale **Registration no**:2014BCS073

**Name**:Manisha G. Yedke **Registration no**:2014BCS039

Has successfully completed her project on the topic

“ Mobile Plaza”

At Shri Guru Gobind Singhji Institute Of Engineering And Technology,Nanded int the partial fulfillment of the Poat Graduate Degree course in Third Year at Department Of Computer Science And Engineering in the Academic Year 2016-2017 Semester-V—prescribed by the S.R.T.M. U. Nanded

Miss. Priya B. Bhosle Prof.P.S.Nalawade

**Guide Head Of Department**

(Department of CSE) (Department of

i

**ACKNOWLEDGEMENT**

I take this opportunity to express my heartfelt gratitude to my guide, Miss. Priya B. Bhosle, Department of Computer Engineering, SGGSIE&T, Nanded, for her kind cooperation and capable guidance during the course of the seminar, without which the seminar would have been difficult to proceed with.

I thankful to the Prof.P.S.Nalwade, Head of Department, SGGSIE &T for his constant encouragement and suggestions.This would help me to work and completing this seminar.

I would also like to thank our, principal and management for providing the entire lab and other facility.

**NAME: Kajol D. Tale**

**Prachi P .Haldekar**

**Manisha G.Yedke**

ii

**ABSTRACT**

The project **MOBILE PLAZA** is to develop an Internet based sale and purchase system for mobile. As mobile market is increasing very sharply and will increase with boom in near future. Visualizing the huge opportunity this is an effort to internationalize the business through Internet. By developing of this portal and keeping the site searchable on Google hence increasing the customer base from a local market to all around the globe.

My ongoing project is a web application because nowadays Internet is a prominent tool of marketing mantra. With the advent of the Internet technologies, world has become a global village. Every year, millions of people around the world use the Internet to interact in more ways then ever before till date. People, who surf on Internet, are thinking about the online purchase because it is very easy and comfortable to order online than going to market and purchase their required goods. So considering a big hike in the revenue in this booming sector and one of the successful businesses through Internet one should be proud to have such a technical deal.

With the fast growing of communications network like optical fiber, making way to broadband it will be now easy and economical also to access the Internet, whether in monetary or information retrieval term.

iii

**INDEX**

|  |
| --- |
| **Certificate i** |
| **Acknowledgment ii** |
| **Abstract iii** |

|  |  |  |
| --- | --- | --- |
| **Chapter No** | **Title** | **Page No** |
| 1 | **Introduction** | 1 |
| 2 | **Existing System** | 3 |
| 3 | **Proposed System** | 9 |
| 4 | **Modules** | 11 |
| 5 | **Requirements** | 15 |
| 6 | **Result** | 17 |
| 7 | **Conclusion** | 22 |
| 8 | **References** | 23 |

|  |
| --- |
|  |
|  |
|  |
|  |

**Chapter 1**

**INTRODUCTION**

The main goal of this project was to create a shopping cart, whic allows customers to shop and purchase the products online. Moreover, the project is also designed in such a way it lets managers manage the products information online.

This online store sells wine cellars. Customers can orders products, and they will be contacted to further process the orders.

A shopping cart is an important part of every e-Commerce project. It helps the user to select multiple items for purchase and view total price before submitting the order. in this PHP project it is made easy to understand the shopping cart concept. In this script, I am using PHP session with MYSQL to store the products information in the cart. Once the order is submitted by the user, the products information would be inserted into the database using PHP and MySQL.

**Technological Background:**

The tools used to develop the shopping cart include:

- PHP scripting language

- MySQL database server

- Apache server

- HTML

- Text Editor

PHP is a server side scripting language designed specially for the Web. Within an HTML page, PHP code can be embedded and it will be executed each time the page is visited. PHP is an Open Source product. PHP source code can be accessed, used, altered, and redistributed without any charges.

MySQL is a very fast, robust, relational database management system (RDBMS). A database enables the users to efficiently store, search, sort, and retrieve data.

Apache server interprets the PHP code at the Web server and generates HTML or other output that the visitor will see.

Text editor such as Notepad is used to write the coding in plain text that the Web server can recognize and translate into binary code.

There are advantages of using these tools to develop and run the software.

***Forms:***

1. Login
2. Home page
3. Admin Home Page
4. Customer Home page
5. View Profile
6. view cart

Project Objective:

The objective of the project “**MOBILE PLAZA”** is to develop a viable e-marketing system for the purchase and sale of mobile phones.

**The major objectives of this project are as follows: -**

1. Providing an interactive interface for the customer for online sale and purchase of Mobiles, purchase details input, sales details input, search of the MOBILE type available on the given criteria, online booking of the MOBILE to be purchased, with the facility of the shopping cart using the flavors of H.T.M.L, X.M.L. blended with the power of php under the framework of php After all interface has a lot to do when proper interaction and attraction is necessary.
2. Online booking of Mobiles with the delivery status notification system.
3. Retrieval of data from database using easy buttons, links and interfaces in a very fast and efficient manner.
4. On-line updating of all types of data. Updating Data using web based interfaces and with the help of buttons & links in an easy & convenient way.

Developing a viable candidate system that fulfills the following requirements:

* + - Faster and accurate information retrieval.
    - Easy registration.
    - Login dependent information retrieval and updating.
    - Correct and consistent maintenance of data and its

quick retrieving.

Minimizing the page download time and hence less Investment.

**1**.Improving the switches button and hyperlink so that the candidate shall find all the queries they need at first sight.

**2**.Maintaining the fast download of the page by using the flavors of cookies.

**Chapter 2**

**Existing System**

➢ In the current system all work are done manually.

➢ Wastage of time to find out how many order are placed.

➢ In manual system generating reports for particular types of data based on some condition istime consuming task.

➢ Searching facility is not provided in a System.

➢ Time & Money is not save in a System

**Chapter 3**

**Proposed System**

The undergoing project falls under **INTERNET TECHNOLOGIES**&**RDBMS (Relational Database Management System)** category. Since the project is mainly responsible for creation of the portal with the online database at backend. As we know that, the Internet is huge client server architecture. The client is the web browser, it is requesting a web based data, a file, or whatever, from some computer somewhere – anywhere – in the world. The server is that computer that holds the information you want.PHP is a rich programming framework for building web-based applications. It offers outstanding support for both developers and administrators, providing improved ease-of-use, tool support, reliability, scalability, administration and security.

* **ENHANCED PERFORMANCE: -** PHP is compiled common language runtime code running on the server. Unlike its interpreted predecessors, PHP can take advantage of early binding, just-in-time compilation, native optimization, and caching services right out of the box.
* **POWER AND FLEXIBILITY: -** Because PHP is based on the common language runtime, the power and flexibility of that entire platform is available to Web application developers. PHP is also language-independent, so you can choose the language that best applies to your application or partition your application across many languages. Further, common language runtime
* **SIMPLICITY:** - PHP makes it easy to perform common tasks, from simple form submission and client authentication to deployment and Site configuration. For example, the PHP page framework allows you to build user interfaces that cleanly separate application logic from presentation code and to handle events in a simple, Visual Basic- like forms processing model.
* **MANAGEABILITY: -** PHP employs a text-based, hierarchical configuration system, which simplifies applying settings to your server environment and Web applications, because configuration information is stored as plain text, new settings may be applied without the aid of local administration tools.
* **SCALABILITY AND AVAILABILITY: -** PHP has been designed with scalability in mind, with features specifically tailored to improve performance in clustered and multiprocessor environments. Further, processes are closely monitored and managed by the PHP runtime, so that if one misbehaves (leaks, deadlocks), a new process can be created in its place, which helps keep your application constantly available to handle requests.

In database design specific objectives are considered: -

1. **Ease of learning and use.**
2. **More information at low cost.**
3. **Accuracy and integrity.**
4. **Recovery from failure.**
5. **Performance.**

**Chapter 4**

**Modules Description**

**PROJECT MODULES:**

The proposed website will have its main page and will be mainly divided into partially dependent and partially independent modules as: -

* 1. **LOGIN SYSTEM**
  2. **THE SEARCH MASTER**
  3. **ITEM MASTER**
  4. **ORDER MASTER**
  5. **SHOPPING CART**
  6. **PRODUCT SALE PURCHASE MAILER**

These modules with their sub module are described below: -

1. **LOGIN MASTER: -**

Login master is the module which checks for a valid candidate when the user enters his user-id, password and link to the correct page or denies and link to the registration page.

As user id rules the system so a person is known by his uniqueness of his user id. As it is to provide the viable candidate system .so the user id is being validated with password in different cases to validate the genuine ness of the candidate.

This module keeps the status of who and when logged in and for which purpose and how much time.

**2. SEARCH MASTER: -** This module is the main module of the project. An enhanced search tool is made to search item of different categories on the basis of item type, item name or the difficult one on the basis of description and criteria given by the visitor of the site.

**3. ITEM MASTER:-** This module specially relates with item being searched on the site and are not available, such item and their description are being sent to all the members of the site, which is being handled by product sale purchase mailer module. A special database is maintained for such items which when uploaded on the site being informed to the party through e-mail.

**4. ORDER MASTER: -** It deals with orders being booked and transaction related to a candidate that first books the MOBILE that he wants to purchase and he is asked to send the amount through suitable media to the company account if the transaction succeeds, the booking detail is being deleted from the database and the data is being transferred to the purchase detail or after a limited period the booking detail expires.

**5.SHOPPING CART SYSTEM:-** If a user navigates through the site and WANTS TO BOOK MORE THEN ONE MOBILE, THEN this SHOPPING CART IS VERY MUCH HELPFUL FOR SUCH CASES. This shopping cart simply keeps his entire purchased MOBILE rate maintained on the server so that there should be no ambiguity of what the user bought and what he left. This is done through maintaining virtual table for each user and a unique session id generated by the system that remains same with the current user session based on that uniqueness is maintained and whatever the person purchase is updated in to the shopping cart, which is being displayed with the amount entered.

**8. PRODUCT SALE PURCHASE MAILER:-** This module basically deals with mailing of data to all the members whenever and wherever the module gets called to take the data and mail it to specific suggested or all the members of the portal by using their e-mail addresses from the database.

# **Site summary**

1. MAIN PAGE: - home page of the site

2. Login page: - used for login to use the service provided by the site.

3. MOBILE details display

4. Shopping cart frame

5. Payment details pages and forms

6. Delivery status information pages.

**Chapter 5**

**Implementation Requirement**

**For the undergoing project, following tools are used:**

Platform : Windows

The Operating System : Windows 8.1

Framework : PHP FRAMEWORK

Front-End Tool : PHP With HTML

Editing tools :NOTEPAD

Scripting : VB.NET, JAVASCRIPT, JAVASCRIPT.NET

Interface : H.T.M.L, X.M.L (STYLE SHEET (.XSL))

Server : INTERNET INFORMATION SERVER

Back-End Tool : MYSQL

hardware: -

Processor : INTEL CORE I5 GHZ

Memory : 4 GB, D.D.R. RAM

Network Adaptor : Ethernet Adaptor

Modem : 56 KBPS VOICE FAX DATA

Secondary Storage : SAMSUNG / SEAGATE HARD DISK

(80 GB)

**Chapter 6**

**Result**

**CODE:**

Index.php

<?php

// include database configuration file

include 'dbConfig.php';

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>PHP Shopping Cart Tutorial</title>

<meta charset="utf-8">

<style>

.container{padding: 10px;}

.cart-link{

width: 100%;

text-align: right;

display: block;

font-size: 22px;

}

</style>

</head>

<body bgcolor="#34495E">

<marquee direction="right" bgcolor="#c70039"><img src="c.jpg" height="150" width="250"/></marquee>

<h1><center><font color="#D1F2EB">MOBILEPLAZA</font></center></h1>

<hr>

<div class="container">

<h4><font color="#D1F2EB">PRODUCTS</font></h4>

<hr>

<a href="viewCart.php" class="cart-link" title="View Cart"><i class="glyphicon glyphicon-shopping-cart"></i></a>

<div id="products" class="row list-group">

<?php

$query = $db->query("SELECT \* FROM products ORDER BY id DESC LIMIT 10");

if($query->num\_rows > 0){

while($row = $query->fetch\_assoc()){ ?><div class="item col-lg-4">

<div class="thumbnail">

<div class="caption">

<h4 class="list-group-item-heading">

<?php echo $row["name"]; ?></h4>

<p class="list-group-item-text"><?php echo "Color:".$row["color"]; ?></p>

<div class="row">

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

<p class="lead"><?php echo 'Price:Rs.'.$row["price"]; ?></p>

<a href="image.html"><font color="#D1F2EB">Image</font></a>

</div>

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

<a class="btn btn-success" href="cartAction.php?action=addToCart&id=<?php echo $row["id"]; ?>"><font color="#D1F2EB"><h3>Add to cart</h3></font></a>

</div></div></div></div></div>

<?php } }else{ ?>

<p>Product(s) not found.....</p>

<?php } ?>

</div></div>

</body></html>

Index1.php

<?php

include('login.php'); // Includes Login Script

if(isset($\_SESSION['login\_user'])){

header("location: profile.php");

}

?>

<!DOCTYPE html>

<html>

<head>

<title>Login Form in PHP with Session</title>

<link href="style.css" rel="stylesheet" type="text/css">

</head>

<body>

<div id="main">

<h1>PHP Login Session Example</h1>

<div id="login">

<h2>Login Form</h2>

<form action="" method="post">

<label>UserName :</label>

<input id="name" name="username" placeholder="username" type="text">

<label>Password :</label>

<input id="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*\*" type="password"><br><br>

<input name="submit" type="submit" value=" Login ">

<span><?php echo $error; ?></span>

<p>Don't have account?<a href="registration.php">Register</a></p>

</form></div></div></body>

</html>

viewCart.php

<?php

// initializ shopping cart class

include 'Cart.php';

$cart = new Cart;

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>View Cart - PHP Shopping Cart Tutorial</title>

<meta charset="utf-8">

<style>

.container{padding: 100px;}

input[type="number"]{width: 20%;}

</style>

<script>

function updateCartItem(obj,id){

$.get("cartAction.php", {action:"updateCartItem", id:id, qty:obj.value},

function(data){

if(data == 'ok'){

location.reload();

}else{

alert('Cart update failed, please try again.');

}

});

}

</script>

</head>

</head>

<body bgcolor="#73C6B6">

<div class="container">

<h1><font color="white">SHOPPING CART</font></h1>

<table class="table">

<thead>

<tr>

<th>Product</th>

<th>Price</th>

<th>Quantity</th>

<th>Subtotal</th>

<th>&nbsp;</th>

</tr>

</thead>

<tbody>

<?php

if($cart->total\_items() > 0){

//get cart items from session

$cartItems = $cart->contents();

foreach($cartItems as $item){

?>

<tr>

<td>

<?php echo $item["name"];

?></td>

<td><?php echo '$'.$item["price"].' USD'; ?></td>

<td><input type="number" class="form-control text-center" value="<?php echo $item["qty"];

?>

" onchange="updateCartItem(this, '<?php echo $item["rowid"]; ?>')"></td>

<td><?php echo '$'.$item["subtotal"].' USD'; ?></td>

<td>

<!--<a href="cartAction.php?action=updateCartItem&id=" class="btn btn-info">

<i class="glyphicon glyphicon-refresh"></i></a>-->

<a href="cartAction.php?action=removeCartItem&id=<?php echo $item["rowid"];

?>" class="btn btn-danger" onclick="return confirm('Are you sure?')">

<i class="glyphicon glyphicon-trash"></i></a>

</td>

</tr>

<?php } }

else{ ?>

<tr><td colspan="5"><p>Your cart is empty.....</p></td>

<?php } ?>

</tbody>

<tfoot>

<tr>

<td><a href="index.php" class="btn btn-warning"><i class="glyphicon glyphicon-menu-left">

</i><font color="#D1F2EB"><h3>Continue Shopping</h3></font></a></td>

<td colspan="2"></td>

<?php if($cart->total\_items() > 0){ ?>

<td class="text-center"><strong>Total

<?php echo '$'.$cart->total().' USD'; ?></strong></td>

<td><a href="index1.php" class="login-link" title="Login"><font color="#D1F2EB"><h3>Buy Now</h3></font><i class="glyphicon glyphicon-shopping-login"></i></a></td>

<?php } ?>

</tr>

</tfoot>

</table>

</div>

</body>

</html>

Cart.php

<?php session\_start();

class Cart {

protected $cart\_contents = array();

public function \_\_construct(){

// get the shopping cart array from the session

$this->cart\_contents = !empty($\_SESSION['cart\_contents'])?$\_SESSION['cart\_contents']:NULL;

if ($this->cart\_contents === NULL){

// set some base values

$this->cart\_contents = array('cart\_total' => 0, 'total\_items' => 0);

}

}

/\*\*

\* Cart Contents: Returns the entire cart array

\* @param bool

\* @return array

\*/

public function contents(){

// rearrange the newest first

$cart = array\_reverse($this->cart\_contents);

// remove these so they don't create a problem when showing the cart table

unset($cart['total\_items']);

unset($cart['cart\_total']);

return $cart;

}

/\*\*

\* Get cart item: Returns a specific cart item details

\* @param string $row\_id

\* @return array

\*/

public function get\_item($row\_id){

return (in\_array($row\_id, array('total\_items', 'cart\_total'), TRUE) OR ! isset($this->cart\_contents[$row\_id]))

? FALSE

: $this->cart\_contents[$row\_id];

}

/\*\*

\* Total Items: Returns the total item count

\* @return int

\*/

public function total\_items(){

return $this->cart\_contents['total\_items'];

}

/\*\*

\* Cart Total: Returns the total price

\* @return int

\*/

public function total(){

return $this->cart\_contents['cart\_total'];

}

/\*\*

\* Insert items into the cart and save it to the session

\* @param array

\* @return bool

\*/

public function insert($item = array()){

if(!is\_array($item) OR count($item) === 0){

return FALSE;

}else{

if(!isset($item['id'], $item['name'], $item['price'], $item['qty'])){

return FALSE;

}else{

/\*

\* Insert Item

\*/

// prep the quantity

$item['qty'] = (float) $item['qty'];

if($item['qty'] == 0){

return FALSE;

}

// prep the price

$item['price'] = (float) $item['price'];

// create a unique identifier for the item being inserted into the cart

$rowid = md5($item['id']);

// get quantity if it's already there and add it on

$old\_qty = isset($this->cart\_contents[$rowid]['qty']) ? (int) $this->cart\_contents[$rowid]['qty'] : 0;

// re-create the entry with unique identifier and updated quantity

$item['rowid'] = $rowid;

$item['qty'] += $old\_qty;

$this->cart\_contents[$rowid] = $item;

// save Cart Item

if($this->save\_cart()){

return isset($rowid) ? $rowid : TRUE;

}else{

return FALSE;

}

}

}

}

/\*\*

\* Update the cart

\* @param array

\* @return bool

\*/

public function update($item = array()){

if (!is\_array($item) OR count($item) === 0){

return FALSE;

}else{

if (!isset($item['rowid'], $this->cart\_contents[$item['rowid']])){

return FALSE;

}else{

// prep the quantity

if(isset($item['qty'])){

$item['qty'] = (float) $item['qty'];

// remove the item from the cart, if quantity is zero

if ($item['qty'] == 0){

unset($this->cart\_contents[$item['rowid']]);

return TRUE;

}

}

// find updatable keys

$keys = array\_intersect(array\_keys($this->cart\_contents[$item['rowid']]), array\_keys($item));

// prep the price

if(isset($item['price'])){

$item['price'] = (float) $item['price'];

}

// product id & name shouldn't be changed

foreach(array\_diff($keys, array('id', 'name')) as $key){

$this->cart\_contents[$item['rowid']][$key] = $item[$key];

}

// save cart data

$this->save\_cart();

return TRUE;

}

}

}

/\*\*

\* Save the cart array to the session

\* @return bool

\*/

protected function save\_cart(){

$this->cart\_contents['total\_items'] = $this->cart\_contents['cart\_total'] = 0;

foreach ($this->cart\_contents as $key => $val){

// make sure the array contains the proper indexes

if(!is\_array($val) OR !isset($val['price'], $val['qty'])){

continue;

}

$this->cart\_contents['cart\_total'] += ($val['price'] \* $val['qty']);

$this->cart\_contents['total\_items'] += $val['qty'];

$this->cart\_contents[$key]['subtotal'] = ($this->cart\_contents[$key]['price'] \* $this->cart\_contents[$key]['qty']);

}

// if cart empty, delete it from the session

if(count($this->cart\_contents) <= 2){

unset($\_SESSION['cart\_contents']);

return FALSE;

}else{

$\_SESSION['cart\_contents'] = $this->cart\_contents;

return TRUE;

}

}

/\*\*

\* Remove Item: Removes an item from the cart

\* @param int

\* @return bool

\*/

public function remove($row\_id){

// unset & save

unset($this->cart\_contents[$row\_id]);

$this->save\_cart();

return TRUE;

}

/\*\*

\* Destroy the cart: Empties the cart and destroy the session

\* @return void

\*/

public function destroy(){

$this->cart\_contents = array('cart\_total' => 0, 'total\_items' => 0);

unset($\_SESSION['cart\_contents']);

}

}

cartAction.php

<?php

// initialize shopping cart class

include 'Cart.php';

$cart = new Cart;

// include database configuration file

include 'dbConfig.php';

if(isset($\_REQUEST['action']) && !empty($\_REQUEST['action']))

{

if($\_REQUEST['action'] == 'addToCart' && !empty($\_REQUEST['id']))

{

$productID = $\_REQUEST['id'];

// get product details

$query = $db->query("SELECT \* FROM products WHERE id = ".$productID);

$row = $query->fetch\_assoc();

$itemData = array(

'id' => $row['id'],

'name' => $row['name'],

'price' => $row['price'],

'qty' => 1

);

$insertItem = $cart->insert($itemData);

$redirectLoc = $insertItem?'viewCart.php':'index.php';

header("Location: ".$redirectLoc);

}

elseif($\_REQUEST['action'] == 'updateCartItem' && !empty($\_REQUEST['id']))

{

$itemData = array(

'rowid' => $\_REQUEST['id'],

'qty' => $\_REQUEST['qty']

);

$updateItem = $cart->update($itemData);

echo $updateItem?'ok':'err';die;

}

elseif($\_REQUEST['action'] == 'removeCartItem' && !empty($\_REQUEST['id']))

{

$deleteItem = $cart->remove($\_REQUEST['id']);

header("Location: viewCart.php");

}

elseif($\_REQUEST['action'] == 'placeOrder' && $cart->total\_items() > 0 && !empty($\_SESSION['sessCustomerID']))

{

// insert order details into database

$insertOrder = $db->query("INSERT INTO orders (customer\_id, total\_price, created, modified)

VALUES ('".$\_SESSION['sessCustomerID']."', '".$cart->total()."', '".date("Y-m-d H:i:s")."', '".date("Y-m-d H:i:s")."')");

if($insertOrder){

$orderID = $db->insert\_id;

$sql = '';

// get cart items

$cartItems = $cart->contents();

foreach($cartItems as $item){

$sql .= "INSERT INTO order\_items (order\_id, product\_id, quantity) VALUES ('".$orderID."', '".$item['id']."', '".$item['qty']."');";

}

// insert order items into database

$insertOrderItems = $db->multi\_query($sql);

if($insertOrderItems){

$cart->destroy();

header("Location: orderSuccess.php?id=$orderID");

}else{

header("Location: checkout.php");

}

}else{

header("Location: checkout.php");

}

}

else{

header("Location: index.php");

}

}

else{

header("Location: index.php");

}

Checkout.php

<?php

// include database configuration file

include 'dbConfig.php';

// initializ shopping cart class

include 'Cart.php';

$cart = new Cart;

// redirect to home if cart is empty

if($cart->total\_items() <= 0){

header("Location: index.php");

}

// set customer ID in session

$\_SESSION['sessCustomerID'] = 1;

// get customer details by session customer ID

$query = $db->query("SELECT \* FROM customers WHERE id = ".$\_SESSION['sessCustomerID']);

$custRow = $query->fetch\_assoc();

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Checkout - PHP Shopping Cart Tutorial</title>

<meta charset="utf-8">

<link rel="stylesheet" href="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/css/bootstrap.min.css">

<script src="https://ajax.googleapis.com/ajax/libs/jquery/1.12.4/jquery.min.js"></script>

<script src="http://maxcdn.bootstrapcdn.com/bootstrap/3.3.7/js/bootstrap.min.js"></script>

<style>

.container{width: 100%;padding: 50px;}

.table{width: 65%;float: left;}

.shipAddr{width: 30%;float: left;margin-left: 30px;}

.footBtn{width: 95%;float: left;}

.orderBtn {float: right;}

</style>

</head>

<body>

<div class="container">

<h1>Order Preview</h1>

<table class="table">

<thead>

<tr>

<th>Product</th>

<th>Price</th>

<th>Quantity</th>

<th>Subtotal</th>

</tr>

</thead>

<tbody>

<?php

if($cart->total\_items() > 0){

//get cart items from session

$cartItems = $cart->contents();

foreach($cartItems as $item){

?>

<tr>

<td><?php echo $item["name"]; ?></td>

<td><?php echo '$'.$item["price"].' USD'; ?></td>

<td><?php echo $item["qty"]; ?></td>

<td><?php echo '$'.$item["subtotal"].' USD'; ?></td>

</tr>

<?php } }else{ ?>

<tr><td colspan="4"><p>No items in your cart......</p></td>

<?php } ?>

</tbody>

<tfoot>

<tr>

<td colspan="3"></td>

<?php if($cart->total\_items() > 0){ ?>

<td class="text-center"><strong>Total <?php echo '$'.$cart->total().' USD'; ?></strong></td>

<?php } ?>

</tr>

</tfoot>

</table>

<div class="shipAddr">

<h4>Shipping Details</h4>

<p><?php echo $custRow['name']; ?></p>

<p><?php echo $custRow['email']; ?></p>

<p><?php echo $custRow['phone']; ?></p>

<p>

<?php echo $custRow['address']; ?></p>

</div>

<div class="footBtn">

<a href="index.php" class="btn btn-warning">

<i class="glyphicon glyphicon-menu-left"></i> Continue Shopping</a>

<a href="cartAction.php?action=placeOrder" class="btn btn-success orderBtn">Place Order <i class="glyphicon glyphicon-menu-right"></i></a>

</div>

</div>

</body>

</html>

Dbconfig.php

<?php

//DB details

$dbHost = 'localhost';

$dbUsername = 'root';

$dbPassword = 'kajol';

$dbName = 'shopping';

//Create connection and select DB

$db = new mysqli($dbHost, $dbUsername, $dbPassword, $dbName);

if ($db->connect\_error) {

die("Unable to connect database: " . $db->connect\_error);

}

?>

Login.php

<?php

session\_start(); // Starting Session

$error=''; // Variable To Store Error Message

if (isset($\_POST['submit'])) {

if (empty($\_POST['username']) || empty($\_POST['password'])) {

$error = "Username or Password is invalid";

}

else

{

// Define $username and $password

$username=$\_POST['username'];

$password=$\_POST['password'];

// Establishing Connection with Server by passing server\_name, user\_id and password as a parameter

$connection = mysql\_connect("localhost", "root", "kajol");

// To protect MySQL injection for Security purpose

$username = stripslashes($username);

$password = stripslashes($password);

$username = mysql\_real\_escape\_string($username);

$password = mysql\_real\_escape\_string($password);

// Selecting Database

$db = mysql\_select\_db("shopping", $connection);

// SQL query to fetch information of registerd users and finds user match.

$query = mysql\_query("select \* from login where password='$password' AND username='$username'", $connection);

$rows = mysql\_num\_rows($query);

if ($rows == 1) {

$\_SESSION['login\_user']=$username; // Initializing Session

header("location: profile.php"); // Redirecting To Other Page

} else {

$error = "Username or Password is invalid";

}

mysql\_close($connection); // Closing Connection

}

}

?>

Logout.php

<?php

session\_start();

if(session\_destroy()) // Destroying All Sessions

{

header("Location: index.php"); // Redirecting To Home Page

}

?>

orderSucess.php

<?php

if(!isset($\_REQUEST['id'])){

header("Location: index.php");

}

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>Order Success - PHP Shopping Cart Tutorial</title>

<meta charset="utf-8">

<style>

.container{width: 100%;padding: 50px;}

p{color: #34a853;font-size: 18px;}

</style>

</head>

</head>

<body>

<div class="container">

<h1>Order Status</h1>

<p>Your order has submitted successfully. Order ID is #<?php echo $\_GET['id']; ?></p>

</div>

</body>

</html>

Profile.php

<?php

include('session.php');

?>

<!DOCTYPE html>

<html lang="en">

<html>

<head>

<title>Your Home Page</title><meta charset="utf-8">

<link href="style.css" rel="stylesheet" type="text/css">

<style>

.container{width: 100%;padding: 50px;}

.table{width: 65%;float: left;}

.shipAddr{width: 30%;float: left;margin-left: 30px;}

.footBtn{width: 95%;float: left;}

.orderBtn {float: right;}

</style>

</head>

<body bgcolor="#E6B0AA">

<div id="profile">

<b id="welcome">Welcome : <i><?php echo $login\_session; ?></i></b>

<b id="logout"><a href="logout.php">Log Out</a></b>

</div>

<?php

// include database configuration file

include 'dbConfig.php';

// initializ shopping cart class

include 'Cart.php';

$cart = new Cart;

// redirect to home if cart is empty

if($cart->total\_items() <= 0){

header("Location: index.php");

}

// set customer ID in session

$\_SESSION['sessCustomerID'] = 1;

// get customer details by session customer ID

$query = $db->query("SELECT \* FROM customers WHERE id = ".$\_SESSION['sessCustomerID']);

$custRow = $query->fetch\_assoc();

?>

<div class="container">

<h1><font color="#D1F2EB">ORDER PREVIEW</font></h1>

<table class="table">

<thead>

<tr>

<th>Product</th>

<th>Price</th>

<th>Quantity</th>

<th>Subtotal</th>

</tr>

</thead>

<tbody>

<?php

if($cart->total\_items() > 0){

//get cart items from session

$cartItems = $cart->contents();

foreach($cartItems as $item){

?>

<tr>

<td><?php echo $item["name"]; ?></td>

<td>

<?php echo '$'.$item["price"].' USD'; ?></td>

<td>

<?php echo $item["qty"]; ?></td>

<td><?php echo '$'.$item["subtotal"].' USD'; ?>

</td>

</tr>

<?php } }else{ ?>

<tr><td colspan="4"><p>No items in your cart......</p></td>

<?php } ?>

</tbody>

<tfoot>

<tr>

<td colspan="3"></td>

<?php if($cart->total\_items() > 0){ ?>

<td class="text-center"><strong>Total

<?php echo '$'.$cart->total().' USD'; ?></strong></td>

<?php } ?>

</tr>

</tfoot>

</table>

<div class="shipAddr">

<h4>Shipping Details</h4>

<p>

<?php echo $custRow['name']; ?></p>

<p><?php echo $custRow['email']; ?></p>

<p>

<?php echo $custRow['phone']; ?></p>

<p>

<?php echo $custRow['address']; ?></p>

</div>

<div class="footBtn">

<a href="index.php" class="btn btn-warning">

<i class="glyphicon glyphicon-menu-left"></i><font color="#D1F2EB"><h3>Continue Shopping</h3></font></a>

<a href="cartAction.php?action=placeOrder" class="btn btn-success orderBtn"><font color="#D1F2EB"><h3>Place Order</h3></font><i class="glyphicon glyphicon-menu-right"></i></a>

</div>

</div>

</body>

</html>

Registration.php

<?php

include('login.php'); // Includes Login Script

if(isset($\_SESSION['login\_user'])){

header("location: profile.php");

}

?>

<!DOCTYPE html>

<html>

<head>

<title>Registration</title>

<link href="style.css" rel="stylesheet" type="text/css">

</head>

<body>

<div id="main">

<h1>Registration</h1>

<div id="userAccount">

<h2>Registration Form</h2>

<form action="" method="post">

<input id="name" name="name" placeholder="name" type="text">

<input id="phone" name="phone" placeholder="phone" type="text">

<input id="address" name="address" placeholder="address" type="text">

<input id="email" name="email" placeholder="email" type="text">

<input id="name" name="username" placeholder="username" type="text">

<input id="password" name="password" placeholder="\*\*\*\*\*\*\*\*\*\*" type="password"><br><br>

<input name="submit" type="submit" value=" Login ">

</form>

</div>

</div>

</body>

</html>

Session.php

<?php

// Establishing Connection with Server by passing server\_name, user\_id and password as a parameter

$connection = mysql\_connect("localhost", "root", "kajol");

// Selecting Database

$db = mysql\_select\_db("shopping", $connection);

session\_start();// Starting Session

// Storing Session

$user\_check=$\_SESSION['login\_user'];

// SQL Query To Fetch Complete Information Of User

$ses\_sql=mysql\_query("select username from login where username='$user\_check'", $connection);

$row = mysql\_fetch\_assoc($ses\_sql);

$login\_session =$row['username'];

if(!isset($login\_session)){

mysql\_close($connection); // Closing Connection

header('Location: index.php'); // Redirecting To Home Page

}

?>

Viewcart.php

<?php

// initializ shopping cart class

include 'Cart.php';

$cart = new Cart;

?>

<!DOCTYPE html>

<html lang="en">

<head>

<title>View Cart - PHP Shopping Cart Tutorial</title>

<meta charset="utf-8">

<style>

.container{padding: 100px;}

input[type="number"]{width: 20%;}

</style>

<script>

function updateCartItem(obj,id){

$.get("cartAction.php", {action:"updateCartItem", id:id, qty:obj.value},

function(data){

if(data == 'ok'){

location.reload();

}else{

alert('Cart update failed, please try again.');

}

});

}

</script>

</head>

</head>

<body bgcolor="#73C6B6">

<div class="container">

<h1><font color="white">SHOPPING CART</font></h1>

<table class="table">

<thead>

<tr>

<th>Product</th>

<th>Price</th>

<th>Quantity</th>

<th>Subtotal</th>

<th>&nbsp;</th>

</tr>

</thead>

<tbody>

<?php

if($cart->total\_items() > 0){

//get cart items from session

$cartItems = $cart->contents();

foreach($cartItems as $item){

?>

<tr>

<td>

<?php echo $item["name"];

?></td>

<td><?php echo '$'.$item["price"].' USD'; ?></td>

<td><input type="number" class="form-control text-center" value="<?php echo $item["qty"];

?>

" onchange="updateCartItem(this, '<?php echo $item["rowid"]; ?>')"></td>

<td><?php echo '$'.$item["subtotal"].' USD'; ?></td>

<td>

<!--<a href="cartAction.php?action=updateCartItem&id=" class="btn btn-info">

<i class="glyphicon glyphicon-refresh"></i></a>-->

<a href="cartAction.php?action=removeCartItem&id=<?php echo $item["rowid"];

?>" class="btn btn-danger" onclick="return confirm('Are you sure?')">

<i class="glyphicon glyphicon-trash"></i></a>

</td>

</tr>

<?php } }

else{ ?>

<tr><td colspan="5"><p>Your cart is empty.....</p></td>

<?php } ?>

</tbody>

<tfoot>

<tr>

<td><a href="index.php" class="btn btn-warning"><i class="glyphicon glyphicon-menu-left">

</i><font color="#D1F2EB"><h3>Continue Shopping</h3></font></a></td>

<td colspan="2"></td>

<?php if($cart->total\_items() > 0){ ?>

<td class="text-center"><strong>Total

<?php echo '$'.$cart->total().' USD'; ?></strong></td>

<td><a href="index1.php" class="login-link" title="Login"><font color="#D1F2EB"><h3>Buy Now</h3></font><i class="glyphicon glyphicon-shopping-login"></i></a></td>

<?php } ?>

</tr>

</tfoot>

</table>

</div>

</body>

</html>

Database: `shopping`

Table structure for table `customers`

CREATE TABLE IF NOT EXISTS `customers` (

`id` int(11) NOT NULL,

`name` varchar(100) COLLATE utf8\_unicode\_ci NOT NULL,

`email` varchar(100) COLLATE utf8\_unicode\_ci NOT NULL,

`phone` varchar(15) COLLATE utf8\_unicode\_ci NOT NULL,

`address` text COLLATE utf8\_unicode\_ci NOT NULL,

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

Dumping data for table `customers`

INSERT INTO `customers` (`id`, `name`, `email`, `phone`, `address`, `created`, `modified`, `status`) VALUES

(1, 'Test User', 'testuser@gmail.com', '9999999999', 'New York, NY, USA', '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1');

Table structure for table `orders`

CREATE TABLE IF NOT EXISTS `orders` (

`id` int(11) NOT NULL,

`customer\_id` int(11) NOT NULL,

`total\_price` float(10,2) NOT NULL,

`created` datetime NOT NULL,

`modified` datetime NOT NULL,

`status` enum('1','0') COLLATE utf8\_unicode\_ci NOT NULL DEFAULT '1'

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

Table structure for table `order\_items`

CREATE TABLE IF NOT EXISTS `order\_items` (

`id` int(11) NOT NULL,

`order\_id` int(11) NOT NULL,

`product\_id` int(11) NOT NULL,

`quantity` int(5) NOT NULL

) ENGINE=InnoDB DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

Table structure for table `products`

CREATE TABLE IF NOT EXISTS `products` (

`id` int(11) NOT NULL,

`name` varchar(200) COLLATE utf8\_unicode\_ci NOT NULL,

`description` text COLLATE utf8\_unicode\_ci NOT NULL,

`price` float(10,2) NOT NULL,

`created` datetime NOT NULL,

`modified` datetime NOT NULL,

`status` enum('1','0') COLLATE utf8\_unicode\_ci NOT NULL DEFAULT '1'

) ENGINE=InnoDB AUTO\_INCREMENT=3 DEFAULT CHARSET=utf8 COLLATE=utf8\_unicode\_ci;

Dumping data for table `products`

INSERT INTO `products` (`id`, `name`, `description`, `price`, `created`, `modified`, `status`) VALUES

(1, 'Product 1', 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.', 15.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1'),

(2, 'Product 2', 'Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.', 25.00, '2016-08-17 08:21:25', '2016-08-17 08:21:25', '1');

Indexes for dumped tables

Indexes for table `customers`

ALTER TABLE `customers`

ADD PRIMARY KEY (`id`);

Indexes for table `orders`

ALTER TABLE `orders`

ADD PRIMARY KEY (`id`), ADD KEY `customer\_id` (`customer\_id`);

Indexes for table `order\_items`

ALTER TABLE `order\_items`

ADD PRIMARY KEY (`id`), ADD KEY `order\_id` (`order\_id`);

Indexes for table `products`

ALTER TABLE `products`

ADD PRIMARY KEY (`id`);

AUTO\_INCREMENT for dumped tables

AUTO\_INCREMENT for table `customers`

ALTER TABLE `customers`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT;

AUTO\_INCREMENT for table `orders`

ALTER TABLE `orders`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT;

AUTO\_INCREMENT for table `order\_items`

ALTER TABLE `order\_items`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT;

AUTO\_INCREMENT for table `products`

ALTER TABLE `products`

MODIFY `id` int(11) NOT NULL AUTO\_INCREMENT,AUTO\_INCREMENT=3;

Constraints for table `orders`

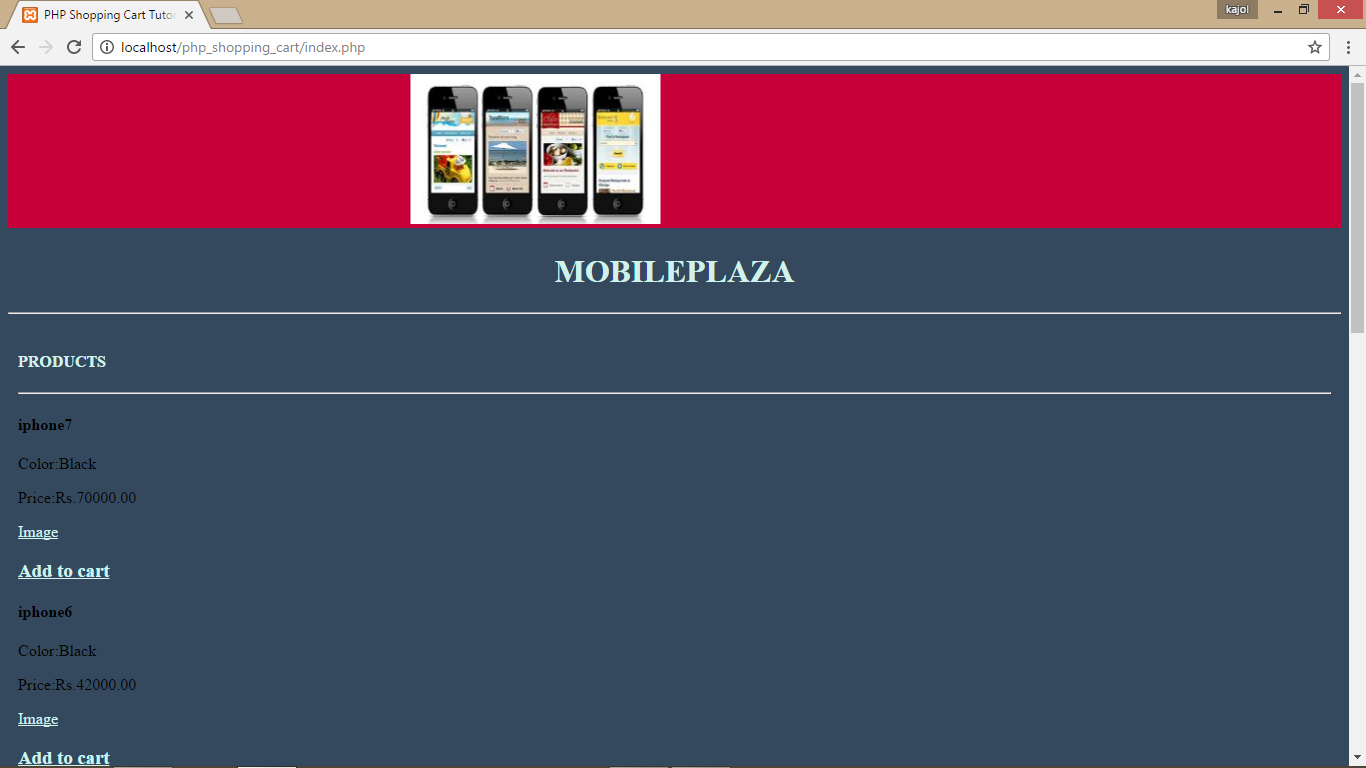
ALTER TABLE `orders`

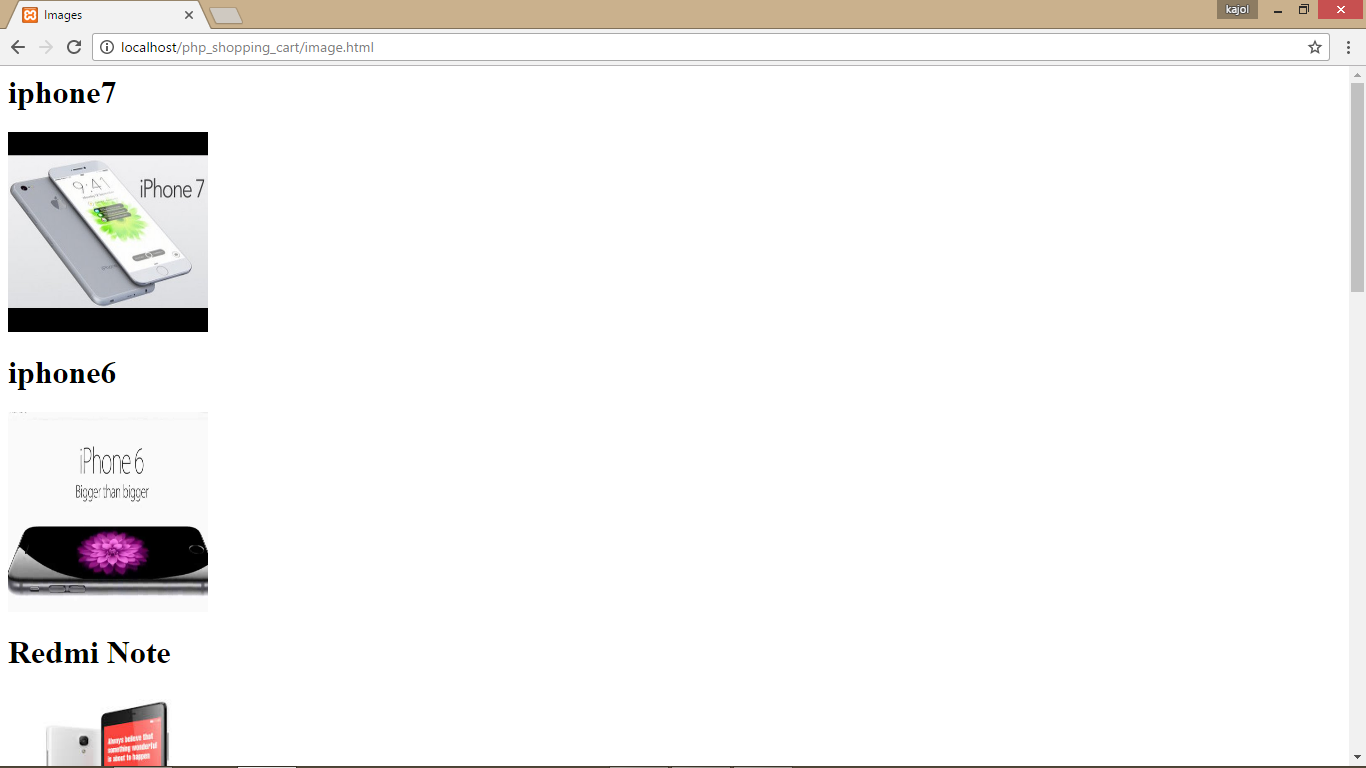
ADD CONSTRAINT `orders\_ibfk\_1` FOREIGN KEY (`customer\_id`) REFERENCES `customers` (`id`) ON DELETE CASCADE ON UPDATE NO ACTION;

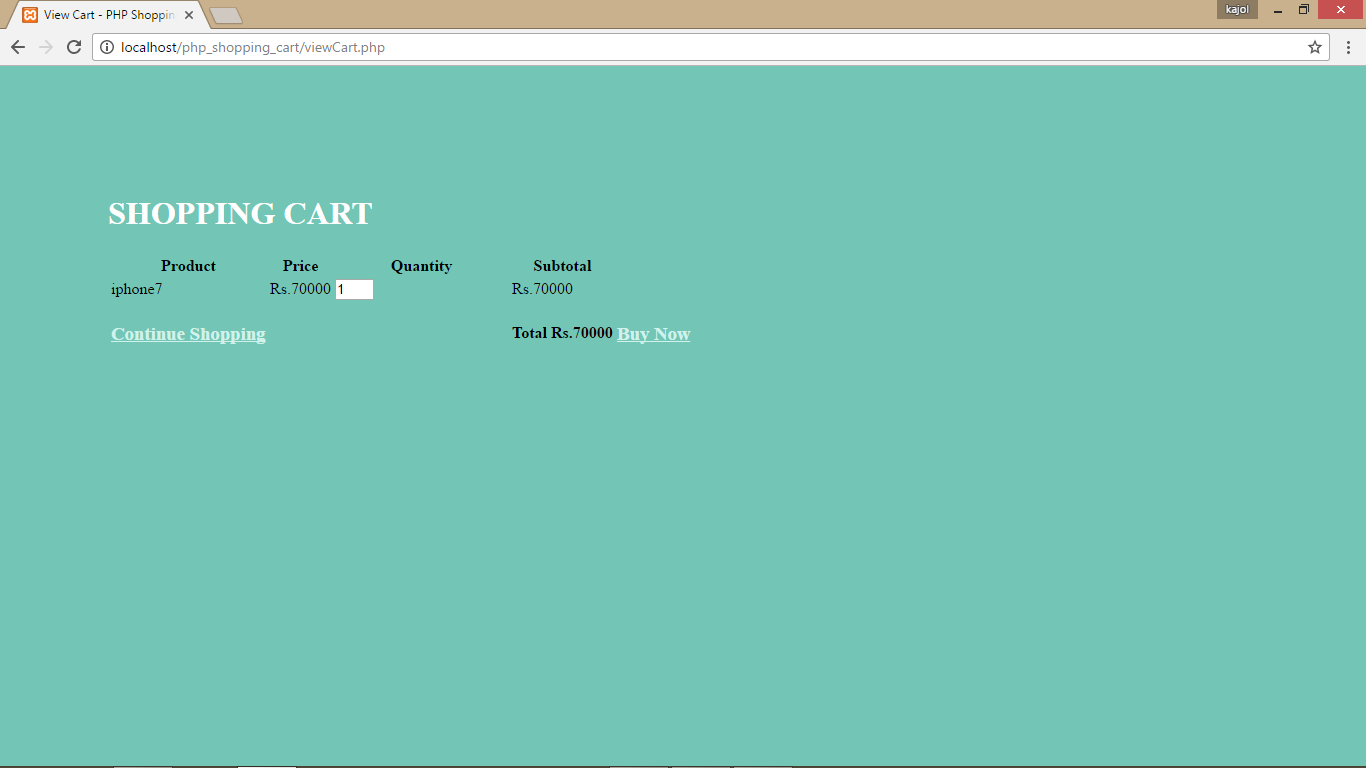
Constraints for table `order\_items`

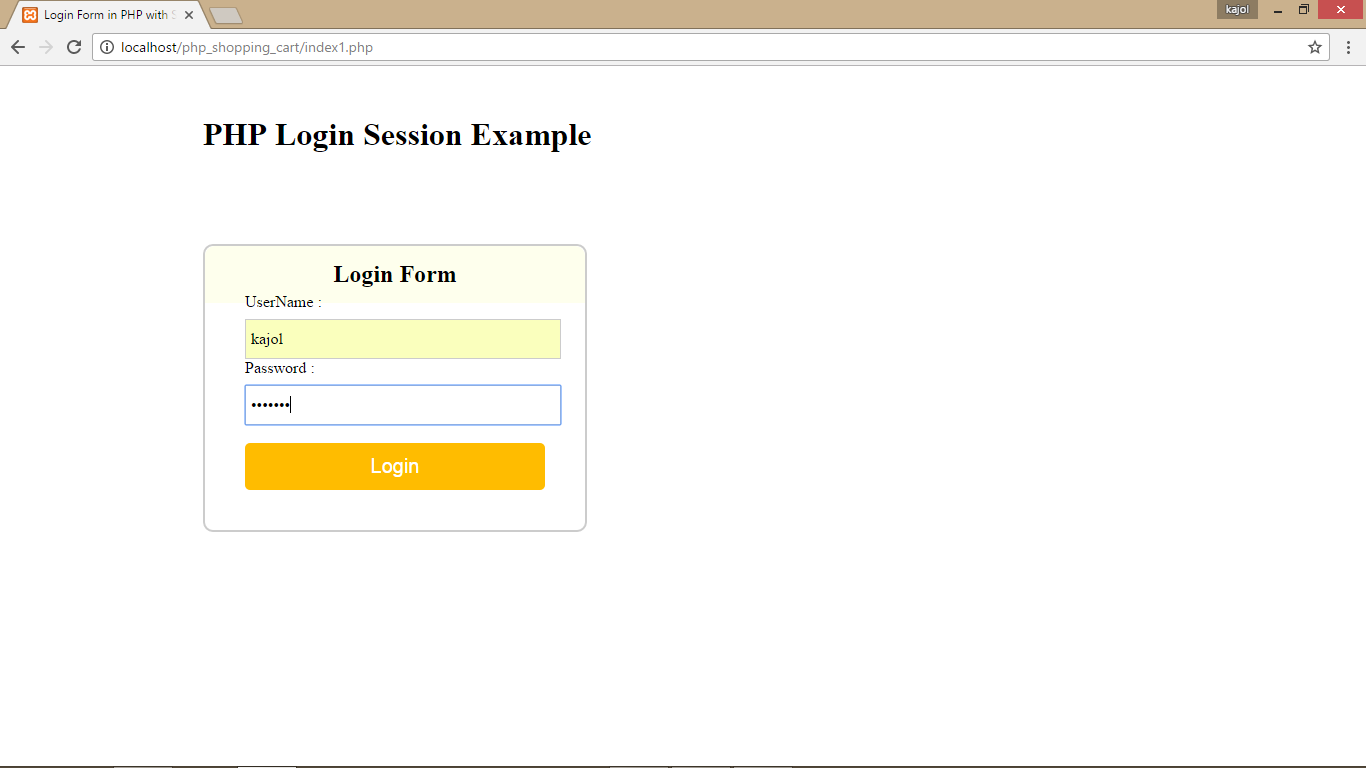
ALTER TABLE `order\_items`

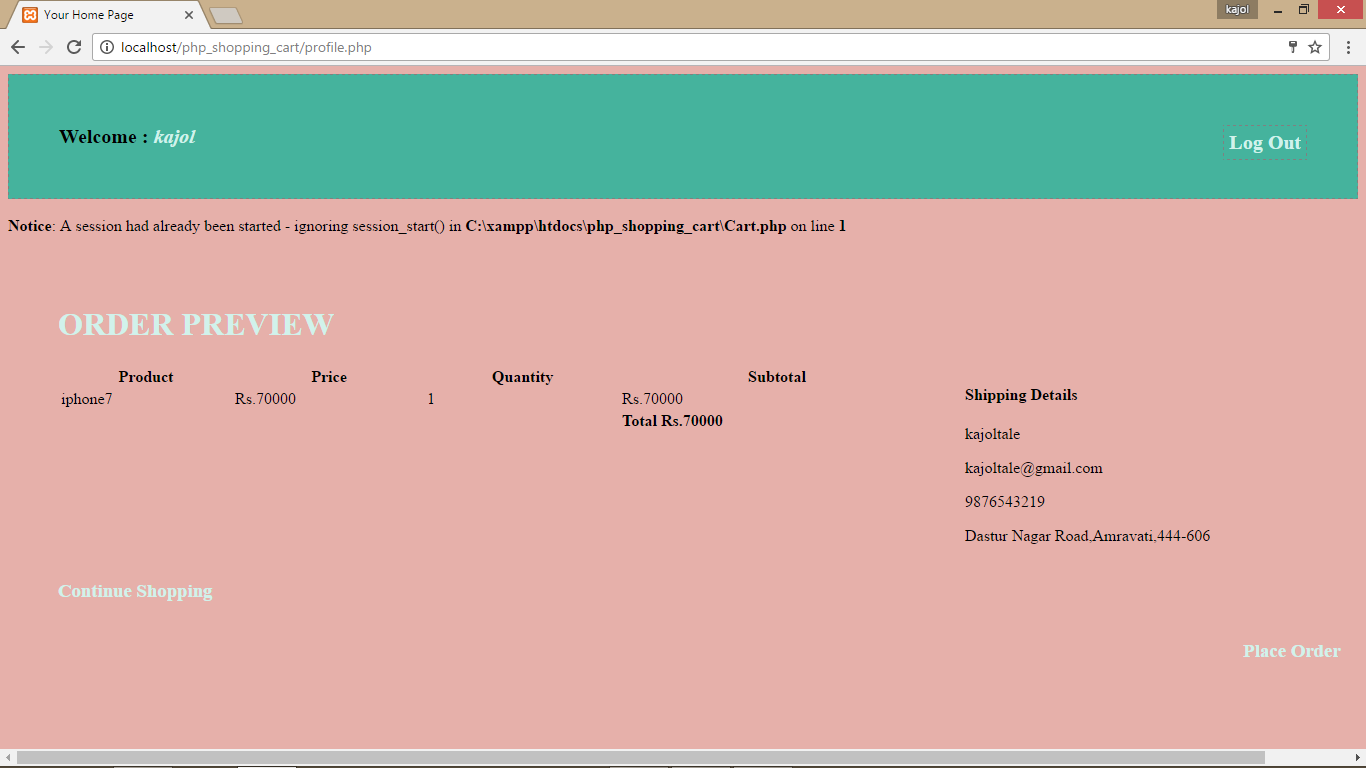
ADD CONSTRAINT `order\_items\_ibfk\_1` FOREIGN KEY (`order\_id`) REFERENCES `orders` (`id`) ON DELETE CASCADE ON UPDATE NO ACTION;

****

****

****

****

****

**Chapter 7**

**Conclusion**

The Internet has become a major resource in modern business, thus electronic shopping has gained significance not only from the entrepreneur’s but also from the customer’s point of view. For the entrepreneur, electronic shopping generates new business opportunities and for the customer, it makes comparative shopping possible.

As per a survey, most consumers of online stores are impulsive and usually make a decision to stay on a site within the first few seconds. “Website design is like a shop interior. If the shop looks poor or like hundreds of other shops the customer is most likely to skip to the other site. Hence we have designed the project to provide the user with easy navigation, retrieval of data and necessary feedback as much as possible. In this project, the user is provided with an ecommerce web site that can be used to buy mobiles online. To implement this as a web application we used PHP and MYSQL as the Technology. PHP has several advantages such asenhanced performance, scalability, built-in security and simplicity.

**Chapter 8**

**Referances**

1 [www.google.com](http://www.google.com)

2 www.filpcart.com