1.INTRODUCTION

UR cart is a online retail store platform that provides customers to buy their favourite apparels and accessories online. UR cart provides a wide range of apparels for men, women and children of all ages.

Online shopping is a form of electronic shopping store where the buyer is directly online to the seller's computer usually via the internet. There is no intermediary service. The sale and purchase transaction is completed electronically and interactively in real-time. The development of this new system contains the following activities, which try to develop on-line application by keeping the entire process in the view of database integration approach. User gets its eamil id and password to acces their account.

Uses of ur cart shopping cart system

Shopping Cart System is the Simple shopping Solution. It's a full-featured website and shopping cart system that bends over backwards to give you the flexibility you need to run your online store. The basic concept of the application is to allow the customer to shop virtually using the Internet and allow customers to buy the items and articles of their desire from the store. The information pertaining to the products are stores on an RDBMS at the server side (store). The Server process the customers and the items are shipped to the address submitted by them. The details of the items are brought forward from the database for the customer view based on the selection through the menu and the database of all the products are updated at the end of each transaction.

UR cart is developed using HTML(hypertext markup language ) and php using xammp and phpmyadmin

2.HTML(Hyper Text Markup Language)

Hypertext Markup Language (HTML) is the standard [markup language](https://en.wikipedia.org/wiki/Markup_language) for documents designed to be displayed in a [web browser](https://en.wikipedia.org/wiki/Web_browser). It can be assisted by technologies such as [Cascading Style Sheets](https://en.wikipedia.org/wiki/Cascading_Style_Sheets) (CSS) and [scripting languages](https://en.wikipedia.org/wiki/Scripting_language) such as [JavaScript](https://en.wikipedia.org/wiki/JavaScript).

[Web browsers](https://en.wikipedia.org/wiki/Web_browser) receive HTML documents from a [web server](https://en.wikipedia.org/wiki/Web_server) or from local storage and [render](https://en.wikipedia.org/wiki/Browser_engine) the documents into multimedia web pages. HTML describes the structure of a [web page](https://en.wikipedia.org/wiki/Web_page) [semantically](https://en.wikipedia.org/wiki/Semantic_Web) and originally included cues for the appearance of the document.

2.1 HISTORY OF HTML

In 1980, physicist [Tim Berners-Lee](https://en.wikipedia.org/wiki/Tim_Berners-Lee), a contractor at [CERN](https://en.wikipedia.org/wiki/CERN), proposed and prototyped [ENQUIRE](https://en.wikipedia.org/wiki/ENQUIRE), a system for CERN researchers to use and share documents. In 1989, Berners-Lee wrote a memo proposing an [Internet](https://en.wikipedia.org/wiki/Internet)-based [hypertext](https://en.wikipedia.org/wiki/Hypertext) system. Berners-Lee specified HTML and wrote the browser and server software in late 1990. That year, Berners-Lee and CERN data systems engineer [Robert Cailliau](https://en.wikipedia.org/wiki/Robert_Cailliau) collaborated on a joint request for funding, but the project was not formally adopted by CERN. In his personal notes from 1990 he listed "some of the many areas in which hypertext is used" and put an encyclopedia first.

The first publicly available description of HTML was a document called ["HTML Tags"](http://info.cern.ch/hypertext/WWW/MarkUp/Tags.html), first mentioned on the Internet by Tim Berners-Lee in late 1991. It describes 18 elements comprising the initial, relatively simple design of HTML. Except for the hyperlink tag, these were strongly influenced by [SGMLguid](https://en.wikipedia.org/wiki/SGMLguid), an in-house [Standard Generalized Markup Language](https://en.wikipedia.org/wiki/Standard_Generalized_Markup_Language) (SGML)-based documentation format at CERN. Eleven of these elements still exist in HTML 4.

HTML is a [markup language](https://en.wikipedia.org/wiki/Markup_language) that [web browsers](https://en.wikipedia.org/wiki/Web_browser) use to interpret and [compose](https://en.wikipedia.org/wiki/Typesetting) text, images, and other material into visual or audible web pages. Default characteristics for every item of HTML markup are defined in the browser, and these characteristics can be altered or enhanced by the web page designer's additional use of [CSS](https://en.wikipedia.org/wiki/Cascading_Style_Sheets). Many of the text elements are found in the 1988 ISO technical report TR 9537 *Techniques for using SGML*, which in turn covers the features of early text formatting languages such as that used by the [RUNOFF command](https://en.wikipedia.org/wiki/TYPSET_and_RUNOFF) developed in the early 1960s for the [CTSS](https://en.wikipedia.org/wiki/Compatible_Time-Sharing_System) (Compatible Time-Sharing System) operating system: these formatting commands were derived from the commands used by typesetters to manually format documents. However, the SGML concept of generalized markup is based on elements (nested annotated ranges with attributes) rather than merely print effects, with also the separation of structure and markup; HTML has been progressively moved in this direction with CSS.

Berners-Lee considered HTML to be an application of SGML. It was formally defined as such by the [Internet Engineering Task Force](https://en.wikipedia.org/wiki/Internet_Engineering_Task_Force) (IETF) with the mid-1993 publication of the first proposal for an HTML specification, the "Hypertext Markup Language (HTML)" Internet Draft by Berners-Lee and [Dan Connolly](https://en.wikipedia.org/wiki/Dan_Connolly_(computer_scientist)), which included an SGML [Document type definition](https://en.wikipedia.org/wiki/Document_type_definition) to define the grammar. The draft expired after six months, but was notable for its acknowledgment of the [NCSA Mosaic](https://en.wikipedia.org/wiki/Mosaic_(web_browser)) browser's custom tag for embedding in-line images, reflecting the IETF's philosophy of basing standards on successful prototypes. Similarly, [Dave Raggett](https://en.wikipedia.org/wiki/Dave_Raggett)'s competing Internet-Draft, "HTML+ (Hypertext Markup Format)", from late 1993, suggested standardizing already-implemented features like tables and fill-out forms.

After the HTML and HTML+ drafts expired in early 1994, the IETF created an HTML Working Group, which in 1995 completed "HTML 2.0", the first HTML specification intended to be treated as a standard against which future implementations should be based.

Further development under the auspices of the IETF was stalled by competing interests. Since 1996, the HTML specifications have been maintained, with input from commercial software vendors, by the [World Wide Web Consortium](https://en.wikipedia.org/wiki/World_Wide_Web_Consortium) (W3C). However, in 2000, HTML also became an international standard ([ISO](https://en.wikipedia.org/wiki/International_Organization_for_Standardization)/[IEC](https://en.wikipedia.org/wiki/International_Electrotechnical_Commission) 15445:2000). HTML 4.01 was published in late 1999, with further errata published through 2001. In 2004, development began on HTML5 in the [Web Hypertext Application Technology Working Group](https://en.wikipedia.org/wiki/Web_Hypertext_Application_Technology_Working_Group) (WHATWG), which became a joint deliverable with the W3C in 2008, and completed and standardized on 28 October 2014.

2.2HTML VERSIONS

**TAGS IN HTML:**

HTML tags most commonly come in pairs like <h1> and </h1>, although some represent empty elements and so are unpaired, for example <img>. The first tag in such a pair is the start tag, and the second is the end tag (they are also called opening tags and closing tags).

Basic tags in html

<!DOCTYPE html>

The Document Type Declaration <!DOCTYPE html> is for HTML5. If a declaration is not included, various browsers will revert to "quirks mode" for rendering

<html> element is the root element of an HTML page. <head> element contains meta information about the document. <title> element specifies a title for the document. <body> element contains the visible page content <h1> element defines a large heading. <p> element defines a paragraph

Tags may also enclose further tag markup between the start and end, including a mixture of tags and text. This indicates further (nested) elements, as children of the parent element.

The start tag may also include attributes within the tag. These indicate other information, such as identifiers for sections within the document, identifiers use to bind style information to the presentation of the document, and for some tags such as the <img> used to embed images, the reference to the image resource.

Some elements, such as the line break <br>, do not permit any embedded content, either text or further tags. These require only a single empty tag (akin to a start tag) and do not use an end tag.

Many tags, particularly the closing end tag for the very commonly used paragraph element <p>, are optional. An HTML browser or other agent can infer the closure for the end of an element from the context and the structural rules defined by the HTML standard. These rules are complex and not widely understood by most HTML coders.

The general form of an HTML element is therefore: **<tag attribute1="value1" attribute2="value2">''content''</tag>.** Some HTML elements are defined as empty elements and take the form **<tag attribute1="value1" attribute2="value2">.** Empty elements may enclose no content, for instance, the <br> tag or the inline <img> tag. The name of an HTML element is the name used in the tags. Note that the end tag's name is preceded by a slash character, /, and that in empty elements the end tag is neither required nor allowed. If attributes are not mentioned, default values are used in each case.

2.3 TAGS USED IN PROJECT

<a>--</a> This defines a hyperlink, which is used to link from one page to another. The most important attribute of the <a> element is the href attribute, which indicates the link's destination.

<div>--</a> tag defines a division or a section in an HTML document. It is often used as a container for other HTML elements to style them with CSS or to perform certain tasks with JavaScript.

<span>---</span> tag is used to group inline-elements in a document. It provides no visual change by itself and also provides a way to add a hook to a part of a text or a part of a document.

<p>--</p> tag defines a paragraph. Browsers automatically add some space (margin) before and after each <p> element. The margins can be modified with CSS (with the margin properties).

<h1>---</h1>

<h1> to <h6> tags are used to define HTML headings.

<h1> defines the most important heading. <h6> defines the least important heading.

<link>---</link> tag defines a link between a document and an external resource. It is used to link to external style sheets.

<meta>---</meta>

Metadata is data (information) about data.<meta> tag provides metadata about the HTML document. Metadata will not be displayed on the page, but will be machine parsable. Meta elements are typically used to specify page description, keywords, author of the document, last modified, and other metadata. It used by browsers (how to display content or reload page), search engines (keywords), or other web services.

<input>---</input> tag specifies an input field where the user can enter data.<input> elements are used within a <form> element to declare input controls that allow users to input data.

<font>---</font> tag specifies the font face, font size, and color of text.It is not used in html 5 we have to use this using css.

<form>---</form> tag is used to create an HTML form for user input. It includes tags such as <input>,<button>,<select>,<output> etc that are used for creating forms.

<p>---</p> tag defines a paragraph. Browsers automatically add some space (margin) before and after each <p> element. The margins can be modified with CSS (with the margin properties).

2.4 HTML ATTRIBUTE

Most of the attributes of an element are name-value pairs, separated by = and written within the start tag of an element after the element's name. The value may be enclosed in single or double quotes, although values consisting of certain characters can be left unquoted in HTML (but not XHTML). Leaving attribute values unquoted is considered unsafe. In contrast with name-value pair attributes, there are some attributes that affect the element simply by their presence in the start tag of the element,like the ismap attribute for the img element.

**There are several common attributes that may appear in many elements :**

* The id attribute provides a document-wide unique identifier for an element. This is used to identify the element so that stylesheets can alter its presentational properties, and scripts may alter, animate or delete its contents or presentation. Appended to the URL of the page, it provides a globally unique identifier for the element, typically a sub-section of the page.
* The class attribute provides a way of classifying similar elements. This can be used for semantic or presentation purposes. For example, an HTML document might semantically use the designation <class="notation"> to indicate that all elements with this class value are subordinate to the main text of the document. In presentation, such elements might be gathered together and presented as footnotes on a page instead of appearing in the place where they occur in the HTML source. Class attributes are used semantically in microformats. Multiple class values may be specified; for example <class="notation important"> puts the element into both the notation and the important classes.
* An author may use the style attribute to assign presentational properties to a particular element. It is considered better practice to use an element's id or class attributes to select the element from within a stylesheet, though sometimes this can be too cumbersome for a simple, specific, or ad hoc styling.
* The title attribute is used to attach subtextual explanation to an element. In most browsers this attribute is displayed as a tooltip.

HTML DATATYPES:

HTML defines several data types for element content, such as script data and stylesheet data, and a plethora of types for attribute values, including IDs, names, URIs, numbers, units of length, languages, media descriptors, colors, character encodings, dates and times, and so on. All of these data types are specializations of character data.

2.5 CSS(CASCADING STYLE SHEETS)

Cascading Style Sheets (CSS) is a style sheet language used for describing

the presentation of a document written in a markup language like HTML. CSS is a cornerstone technology of the World Wide Web, alongside HTML and JavaScript.

CSS is designed to enable the separation of presentation and content, including layout, colors, and fonts.This separation can improve content accessibility, provide more flexibility and control in the specification of presentation characteristics, enable multiple web pages to share formatting by specifying the relevant CSS in a separate .css file, and reduce complexity and repetition in the structural content.

Separation of formatting and content also makes it feasible to present the same markup page in different styles for different rendering methods, such as on-screen, in print, by voice (via speech-based browser or screen reader), and on Braille-based tactile devices. CSS also has rules for alternate formatting if the content is accessed on a mobile device.

The name cascading comes from the specified priority scheme to determine which style rule applies if more than one rule matches a particular element. This cascading priority scheme is predictable.

The CSS specifications are maintained by the World Wide Web Consortium (W3C). Internet media type (MIME type) text/css is registered for use with CSS by RFC 2318 (March 1998). The W3C operates a free CSS validation service for CSS documents. In addition to HTML, other markup languages support the use of CSS including XHTML, plain XML, SVG, and XUL.

3.PHP

PHP is a popular general-purpose scripting language that is especially suited to web development. It was originally created by Rasmus Lerdorf in 1994. the PHP reference implementation is now produced by The PHP Group. PHP originally stood for Personal Home Page, but it now stands for the recursive initialism PHP: Hypertext Pre- processor.

PHP code is usually processed on a web server by a PHP interpreter implemented as a module, a daemon or as a Common Gateway Interface (CGI) executable. On a web server, the result of the interpreted and executed PHP code – which may be any type of data, such as generated HTML or binary image data – would form the whole or part of a HTTP response. Various web template systems, web content management systems, and web frameworks exist which can be employed to orchestrate or facilitate the generation of that response. Additionally, PHP can be used for many programming tasks outside of the web context, such as standalone graphical applications and robotic drone control. Arbitrary PHP code can also be interpreted and executed via command line interface (CLI).

The standard PHP interpreter, powered by the Zend Engine, is free software released under the PHP License. PHP has been widely ported and can be deployed on most web servers on almost every operating system and platform, free of charge.The PHP language evolved without a written formal specification or standard until 2014, with the original implementation acting as the de facto standard which other implementations aimed to follow. Since 2014, work has gone on to create a formal PHP specification.

3.1 VERSIONS IN PHP

Early PHP was not intended to be a new programming language, and grew organically, with Lerdorf noting in retrospect: "I don't know how to stop it, there was never any intent to write a programming language . I have absolutely no idea how to write a programming language, I just kept adding the next logical step on the way." A development team began to form and, after months of work and beta testing, officially released PHP/FI 2 in November 1997.

The fact that PHP was not originally designed, but instead was developed organically has led to inconsistent naming of functions and inconsistent ordering of their parameters. In some cases, the function names were chosen to match the lower-level libraries which PHP was "wrapping", while in some very early versions of PHP the length of the function names was used internally as a hash function, so names were chosen to improve the distribution of hash values.

PHP 3 AND 4

Zeev Suraski and Andi Gutmans rewrote the parser in 1997 and formed the base of PHP 3, changing the language's name to the recursive acronym PHP: Hypertext Preprocessor. Afterwards, public testing of PHP 3 began, and the official launch came in June 1998. Suraski and Gutmans then started a new rewrite of PHP's core, producing the Zend Engine in 1999.They also founded Zend Technologies in Ramat Gan, Israel.

PHP 5

On July 14, 2004, PHP 5 was released, powered by the new Zend Engine II.PHP- 5 included new features such as improved support for object-oriented programming, the PHP Data Objects (PDO) extension (which defines a lightweight and consistent interface for accessing databases), and numerous performance enhancements. In 2008, PHP 5 became the only stable version under development. Late static binding had been missing from PHP and was added in version 5.3.

Many high-profile open-source projects ceased to support PHP 4 in new code as of February 5, 2008, because of the GoPHP5 initiative, provided by a consortium of PHP developers promoting the transition from PHP 4 to PHP 5.

Over time, PHP interpreters became available on most existing 32-bit and 64-bit operating systems, either by building them from the PHP source code, or by using pre-built binaries. For PHP versions 5.3 and 5.4, the only available Microsoft Windows binary distributions were 32-bit IA-32 builds, requiring Windows 32-bit compatibility mode while using Internet Information Services (IIS) on a 64-bit Windows platform. PHP version 5.5 made the 64-bit x86-64 builds available for Microsoft Windows.

Official security support for PHP 5.6 ended on 31 December 2018,but Debian 8.0 Jessie will extend support until June 2020.

On May 22, 2000, PHP 4, powered by the Zend Engine 1.0, was released. As of August 2008 this branch reached version 4.4.9. PHP 4 is no longer under development nor will any security updates be released.

**PHP 6 and Unicode**

PHP received mixed reviews due to lacking native Unicode support at the core language level.In 2005, a project headed by Andrei Zmievski was initiated to bring native Unicode support throughout PHP, by embedding the International Components for Unicode (ICU) library, and representing text strings as UTF-16 internally. Since this would cause major changes both to the internals of the language and to user code, it was planned to release this as version 6.0 of the language, along with other major features then in development.

However, a shortage of developers who understood the necessary changes, and performance problems arising from conversion to and from UTF-16, which is rarely used in a web context, led to delays in the project. As a result, a PHP 5.3 release was created in 2009, with many non-Unicode features back-ported from PHP 6, notably namespaces. In March 2010, the project in its current form was officially abandoned, and a PHP 5.4 release was prepared containing most remaining non-Unicode features from PHP 6, such as traits and closure re-binding.Initial hopes were that a new plan would be formed for Unicode integration, but as of 2014 none had been adopted.

PHP 7

During 2014 and 2015, a new major PHP version was developed, which was numbered PHP 7. The numbering of this version involved some debate. While the PHP 6 Unicode experiment had never been released, several articles and book titles referenced the PHP 6 name, which might have caused confusion if a new release were to reuse the name. After a vote, the name PHP 7 was chosen.

The foundation of PHP 7 is a PHP branch that was originally dubbed PHP next generation (phpng). It was authored by Dmitry Stogov, Xinchen Hui and Nikita Popov, and aimed to optimize PHP performance by refactoring the Zend Engine while retaining near-complete language compatibility. As of 14 July 2014, WordPress-based benchmarks, which served as the main benchmark suite for the phpng project, showed an almost 100% increase in performance. Changes from phpng are also expected to make it easier to improve performance in the future, as more compact data structures and other changes are seen as better suited for a successful migration to a just-in-time (JIT) compiler. Because of the significant changes, the reworked Zend Engine is called Zend Engine 3, succeeding Zend Engine 2 used in PHP 5.

Because of major internal changes in phpng it must receive a new major version number of PHP, rather than a minor PHP 5 release, according to PHP's release process. Major versions of PHP are allowed to break backward-compatibility of code and therefore PHP 7 presented an opportunity for other improvements beyond phpng that require backward-compatibility breaks.

In particular, it involved the following changes:

* Many fatal- or recoverable-level legacy PHP error mechanisms were replaced with modern object-oriented exceptions
* The syntax for variable dereferencing was reworked to be internally more consistent and complete, allowing the use of the operators ->, [], (),{}, and ::, with arbitrary meaningful left-side expressions
* Support for legacy PHP 4-style constructor methods was deprecated
* The behavior of the foreach statement was changed to be more predictable
* Constructors for the few classes built-in to PHP which returned null upon failure were changed to throw an exception instead, for consistency
* Several unmaintained or deprecated server application programming interfaces (SAPIs) and extensions were removed from the PHP core, most notably the legacy mysql extension
* The behavior of the list() operator was changed to remove support for strings
* Support was removed for legacy ASP-style delimiters <% and %> and <script language="php"> ... </script>
* An oversight allowing a switch statement to have multiple default clauses was fixed
* Support for hexadecimal number support in some implicit conversions from strings to number types was removed
* The left-shift and right-shift operators were changed to behave more consistently across platforms
* Conversions between integers and floating point numbers were tightened and implemented more consistently across platforms

PHP 7 also included new language features. Most notably, it introduces return type declarations for functions which complement the existing parameter type declarations, and support for the scalar types (integer, float, string, and boolean) in parameter and return type declarations.

4. XAMPP server :

XAMPP is the most popular PHP development environment(Apache, MySQL, PHP and Perl) Using this link xampp can be downloaded <https://www.apachefriends.org/download.html>

XAMPP is a free and open-source cross-platform web server solution stack package developed by Apache Friends, consisting mainly of the Apache HTTP Server, MariaDB database, and interpreters for scripts written in the PHP and Perl programming languages. Since most actual web server deployments use the same components as XAMPP, it makes transitioning from a local test server to a live server possible.

XAMPP's ease of deployment means a WAMP or LAMP stack can be installed quickly and simply on an operating system by a developer, with the advantage that common add-in applications such as WordPress and Joomla! can also be installed with similar ease using Bitnami.

To write programs in xampp after installation a xampp file is created in c drive in that create a basic folder in htdocs and then save all your php files in that folder.to view the php file open a browser and type http://loacalhost/filename.php

5 STEP BY STEP EXPLANATION OF PROJECT

Technologies used in project are HTML ,PHP and XAMPP server

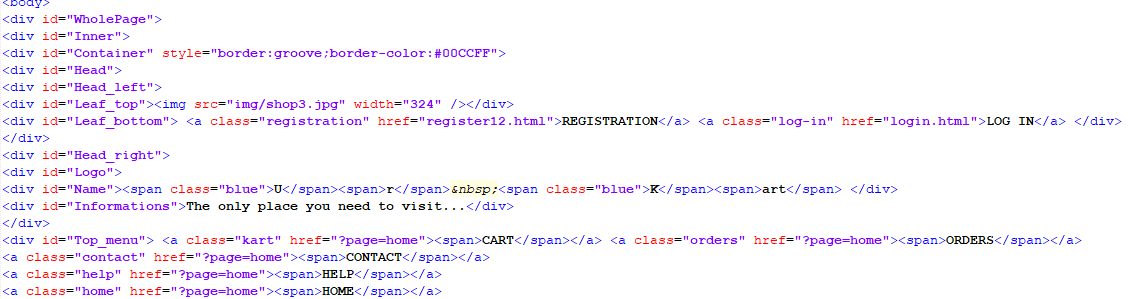
Step 1:install phpmyadmin and xampp server

Step 2: create a database using xampp

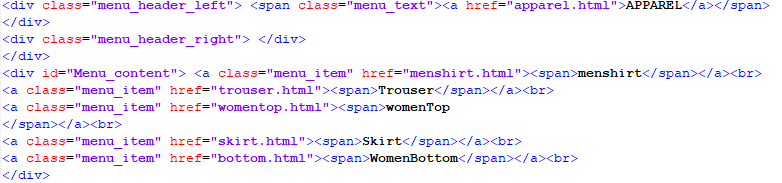
Step 3:create all the html and php files required for website(index file is the main that is using for display and all the remaining files are connected to it)

Step 4: connect all forms and database.

Ur cart online web application index file it :

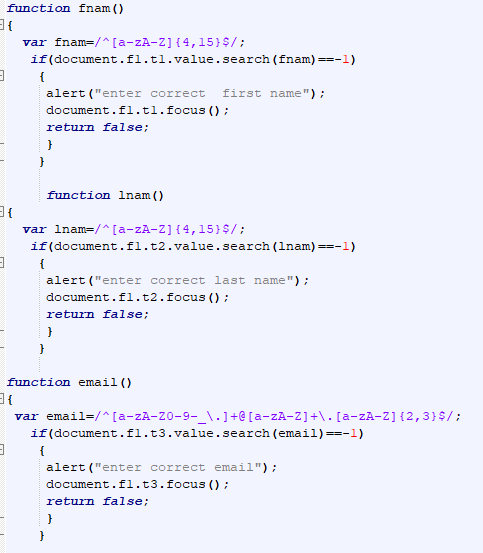


The above code is for creating a registration form and login form here the registration form is attached using href and that is linked to register12.html file and login file is also called using href

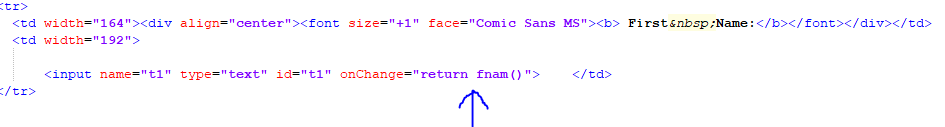
This above code is used for creating a apparel section that consists of men clothing and women clothing and also there are accessories and electronics sections are created.

PHP Connection

For recording the customer details via registration we created a registration form. We created a database named shop in which we created a table in the phpmyadmin through Xampp. Before creating registration form we have written some functions to validate the form on client side it includes name check, email, phone number, address. We created registration form and gave function validation using function call

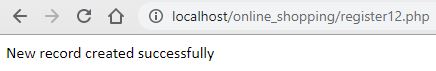


Using onchange we are calling fname() function

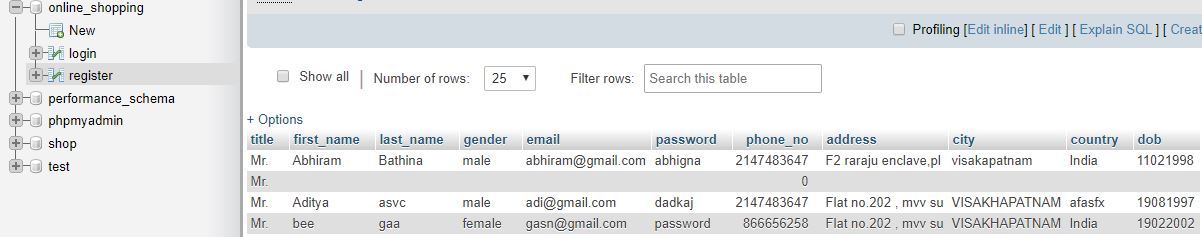


After creating registration form we created a table in database using php and xampp server.

Once a customer successfully enter the details according to fore mentioned credentials the record will be successfully stored in the database.



We can see the data stored whenever we want by opening the table. It looks something like this.



6.CODE of the index file:

<!DOCTYPE html>

<html><head><title>UrKart</title>

<meta http-equiv="Content-language" content="cs">

<meta http-equiv="Content-Type" content="text/html; charset=UTF-8">

<meta name="description" content=" ">

<meta name="keywords" content=" ">

<link rel="shortcut icon" href="favicon.ico" type="image/x-icon">

<link rel="icon" href="favicon.ico" type="image/x-icon">

<meta name="author" content="Klára Frolichová, Sunlight webdesign">

<meta name="Copyright" content="Klára Frolichová, Sunlight webdesign 2007">

<meta name="design" content="Sunlight webdesign - http://www.sunlight.cz, info@sunlight.cz">

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

</head>

<body>

<div id="WholePage">

<div id="Inner">

<div id="Container" style="border:groove;border-color:#00CCFF">

<div id="Head">

<div id="Head\_left">

<div id="Leaf\_top"><img src="img/shop3.jpg" width="324" /></div>

<div id="Leaf\_bottom"> <a class="registration" href="register12.html">REGISTRATION</a> <a class="log-in" href="login.html">LOG IN</a> </div>

</div>

<div id="Head\_right">

<div id="Logo">

<div id="Name"><span class="blue">U</span><span>r</span>&nbsp;<span class="blue">K</span><span>art</span> </div>

<div id="Informations">The only place you need to visit...</div>

</div>

<div id="Top\_menu"> <a class="kart" href="?page=home"><span>CART</span></a> <a class="orders" href="?page=home"><span>ORDERS</span></a>

<a class="contact" href="?page=home"><span>CONTACT</span></a>

<a class="help" href="?page=home"><span>HELP</span></a>

<a class="home" href="?page=home"><span>HOME</span></a>

</div>

</div>

</div>

<div id="CentralPart">

<div id="LeftPart">

<div id="Menu">

<div id="Menu\_header">

<div class="menu\_header\_left"> <span class="menu\_text"><font face="Georgia, Times New Roman, Times, serif">Search</font></span>

</div>

<div class="menu\_header\_right"> </div>

<div id="Menu\_content"> <!--<a class="menu\_item" href="?page=home"><span>--><!-- Start of Page Search -->

<h5>&nbsp;</h5>

<input type="text" name="t1" value="search" onfocus=

"if(this.value=='search')

{this.value='';}

"

onBlur=

"if(this.value=='')

{this.value='search';}

"/>

<input name="sub" type="submit" class="button" id="sub" value="Go" />

<h5>&nbsp;</h5>

<!--</form>-->

<!-- End of Page Search --></span></a><br>

</div>

<div class="menu\_header\_left"> <span class="menu\_text"><a href="apparel.html">APPAREL</a></span>

</div>

<div class="menu\_header\_right"> </div>

</div>

<div id="Menu\_content"> <a class="menu\_item" href="menshirt.html"><span>menshirt</span></a><br>

<a class="menu\_item" href="trouser.html"><span>Trouser</span></a><br>

<a class="menu\_item" href="womentop.html"><span>womenTop

</span></a><br>

<a class="menu\_item" href="skirt.html"><span>Skirt</span></a><br>

<a class="menu\_item" href="bottom.html"><span>WomenBottom</span></a><br>

</div>

<div class="menu\_header\_left"> <span class="menu\_text"><a href="accessories.html">ACCESSORIES</a></span>

</div>

<div class="menu\_header\_right"> </div>

<div id="Menu\_content"> <a class="menu\_item" href="popsocket.html"><span>Popsocket</span></a><br>

<a class="menu\_item" href="earrings.html"><span>Earrings</span></a><br>

<a class="menu\_item" href="phonecase.html"><span>Phonecase</span></a><br>

<a class="menu\_item" href="belt.html"><span>MenBelt</span></a><br>

<a class="menu\_item" href="footwear.html"><span>Foot Wear</span></a><br>

</div>

<div class="menu\_header\_left"> <span class="menu\_text"><a href="kids1.html">ELECTRONICS</a></span>

</div>

<div class="menu\_header\_right"> </div>

<div id="Menu\_content"> <a class="menu\_item" href="laptop.html"><span>Laptop</span></a><br>

<a class="menu\_item" href="tablet.html"><span>Tablet</span></a><br>

<a class="menu\_item" href="pendrive.html"><span>Pendrive</span></a><br>

<a class="menu\_item" href="keyboard.html"><span>Keyboard</span></a><br>

<a class="menu\_item" href="phone.html"><span>Phone</span></a><br>

<a class="menu\_item" href="earphones.html"><span>Earphones</span></a><br>-->

</div>

</div>

<img src="usepics/4.jpg" width="228" height="183" /></div>

<div id="Poll">

<div id="Poll\_header">

<div class="menu\_header\_left"> <span class="menu\_text">Other</span>

<div id="Poll\_header"> <img src="usepics/pay.gif" alt="payment image" width="220" height="90"> </div>

</div>

<div class="menu\_header\_right"> </div>

</div>

<div id="Poll\_content"> <span class="poll\_question">Which payment method would you prefer?</span><br>

<a class="poll\_unswer" href="?page=home"><span>Online

Payment</span></a><br>

<a class="poll\_unswer" href="?page=home"><span>Cash on

delivery</span></a><br>

<a class="poll\_unswer" href="?page=home"><span>Creditcard/

Debitcard</span></a><br>

</div>

</div>

<div id="Banner"> <img src="usepics/3.jpg" alt="illustration image" width="180" height="250"> </div>

</div>-

<div id="RightPart">

<div id="Page"><img src="usepics/akkriti-banner.jpg" alt="" width="669" height="210" name="img"/>

<form name="f1">

<input type="hidden" name="h1" value="0" />

</form>

<div id="Page\_top">

<p> <marquee behavior="scroll" direction="left" onmouseover="stop()" onmouseout="start()"><font color="#FFFF99"><h3>New Trends added...Hurry!!!</h3></font><font color="#00CC99"><h4>Keep shopping for unlimited happiness...</h4></font></marquee><br>

</p>

</div>

<div id="Page\_center">

</div>

</div></div>

<div class="cleaner"></div>

</div>

<div id="Bottom">

</div>

</div>

</div>

</div>

</div>

</body></html>