

DEPARTMENT OF COMPUTER SCIENCE, IT AND ANIMATION



**Project Report
On**

“BOOK RESALE”

Submitted by

Mr. Vishal Kailash Aher (Roll No: 2)

Mr. Karan Prakash Bohare (Roll No: 15)

Mr. Sachin Manbahal Chaudhari (Roll No:16)

Mr. Prashant Ganesh Deshmukh (RollNo:28)

**BCA (Sci.) III Year VI Semester Academic Year
2024-2025**

Guided by

Ms. Minal S. Darekar

Submitted To

**Department of Computer Science, IT and Animation
Deogiri College, Chh. Sambhajinagar 431005.**

DEPARTMENT OF COMPUTER SCIENCE, IT AND ANIMATION



CERTIFICATE

*It is the certified work of candidates of **BCA(Sci.) III Year VI Sem** who have satisfactorily completed the project entitle "**Book Resale**" for the partial fulfillment of the Bachelor Degree from Dr. Babasaheb Ambedkar Marathwada University, Chh. Sambhajinagar for the academic year 2024-25.*

Submitted By

Mr. Vishal Kailas Aher (Roll No: 02)

Mr. Karan Prakash Bohare (Roll No: 15)

Mr. Sachin Manbahal Chaudhari (Roll No:16)

Mr. Prashant Ganesh Deshmukh (RollNo:28)

Project Guide

Ms. Minal S. Darekar

In-charge

Dr. Rajashri A. Joshi

HOD

Dr. Ritesh A. Magre

External Examiner

ACKNOWLEDGEMENT

Many people have contributed in some or the other forms to this project, although our name appears on the cover of this project. We could not have done this Case Study without the assistance or support of each of the following. We thank all of you.

We wish to place on record our deep sense of gratitude towards our project **Ms. Minal S. Darekar** and in-charge **Dr. Rajashri A. Joshi** for the constant motivation and valuable help through the project. We also want to express our gratitude towards our Head of the department **Dr. Ritesh A, Magre**, for his invaluable guidance, insightful advice and continuous encouragement. We also extend our thanks to other faculty members for their co-operation.

We also thank those who have directly as well as indirectly supported us for the completion of this Project.

Mr. Vishal Kailas Aher (Roll No:02)

Mr. Karan Prakash Bohare (Roll No: 15)

Mr. Sachin Manbahal Chaudhari (Roll No:16)

Mr. Prashant Ganesh Deshmukh (RollNo:28)

INDEX

SR. NO.	CHAPTR NAME	PAGE NO.
1.	INTRODUCTION TO HTML	1
2.	INTRODUCTION TO CSS	4
3.	INTRODUCTION TO JAVASCRIPT	7
4.	INTRODUCTION TO PHP	10
5.	INTRODUCTION TO SQL	12
6.	INTRODUCTION TO PROJECT	14
7.	SYSTEM ANALYSIS STUDY REPORT	17
8.	SCOPE OF SYSTEM	18
9.	HARDWARE & SOFTWARE REQUIREMENT	19
10.	DFD (DATA FLOW DIAGRAM)	20
11.	ENTITY REALATIONSHIP (ER) DIAGRAM	22
12.	FEASIBILITY STUDY	25
13.	TESTING	26
14.	DATA DICTIONARY	27
15.	SOURCE CODE	28
16.	OUTPUT	108
17.	CONCLUSION	113
18.	BIBLIOGRAPHY	114
19.	WEB REFERENCES	115

INTRODUCTION OF HTML

INTRODUCTION TO HTML

HTML is a short form of Hyper Text Mark-up Language. HTML was discovered by Tim Berners-Lee in 1991. It is used to design web pages using mark-up language. HTML is the combination of Hypertext and Mark-up language. Hypertext defines the link between the webpages. Mark-up language is used to define the text document within tag which defines the structure of web pages. This language is used to annotate (make notes for the computer) text so that a machine can understand it and manipulate text accordingly. Most of mark-up (e.g. HTML) languages are human readable. Most of mark-up (e.g. HTML) languages are human readable. Language uses tags to define what manipulation has to be done on the text.

HTML is used by the browser to manipulate text, images and other content to display it in required format. The first ever version of HTML was HTML 1.0 but the first standard version was HTML 2.0 which was published in 1999.

1.1 HTML page structure

It contains some elements like head, title, body, etc. These elements are used to build the block of web pages.

The Basic structure of HTML page is given below:

<DOCTYPE! Html>: This tag is used to tell the HTML version. This currently tells that the version is HTML 5.

<Html>: This is called HTML root element and used to wrap all the code.

<Head>: Head tag contains metadata, title, page CSS etc. HTML elements that are used inside the <head> tag are:

<Style>

<Title>

<Base>

<No script>

<Script>

<Meta>

<Title>

<Body>:

Body tag is used to enclose all the data which a web page has from texts to links. All of the content that you see rendered in the browser is contained within this element. Example: HTMLpage can be created using any text editor (notepad). Then save that file using .html or html extension and open that file in browser. It will get the HTML page response.

1.2 Features of HTML are:

- It is easy to learn and easy to use.
- It is platform independent.
- Images, video and audio can be added to a web page.
- Hypertext can be added to text.
- It is a markup language.

1.3 Why learn HTML?

- It is a simple mark-up language. Its implementation is easy.
- It is used to create a website.
- Helps in developing fundamentals about web programming.
- Boost professional career.

1.4 Advantages:

- HTML is used to build a websites.
- It is supported by all browsers.
- It can be integrated with other languages like CSS, JavaScript etc.

1.5 Disadvantages:

- HTML can create only static webpages so for dynamic web page other languages have to be used.
- Large amount of code has to be written to create a simple web page.

Security feature is not good.

Syntax:

```
<Html>  
<head>  
<title>  
</title>  
</head>  
<body>  
</body>  
</html>
```


INTRODUCTION TO CSS

INTRODUCTION TO CSS

Cascading Style Sheets is referred as CSS. It is a simple design language used for process of making web pages presentable. CSS handles the look and feel part of a web page. Using CSS, you can control the colour of the text, the style of fonts, the spacing between paragraphs, how columns are sized and laid out, what background images or colours are used, layout designs, and variations in display for different devices and screen sizes as well as a variety of other effects. CSS is easy to learn and understand but it provides powerful control over the presentation of an HTML document. Most commonly, CSS is combined with the mark-up languages HTML or XHTML.

2.1. What are Style sheets?

In the late '90s, HTML coders noticed that they were retyping the same old tags again and again on the same page, leading to bigger HTML files and above all, time consumption and frustration. You may have found yourself in the same situation, adding in mountains of tags, despite wanting them all the same; or using tricks like invisible gifs for spacing. Then, someone had a great idea: have one file that defines all the values that those piles of tags would have done, and then have all your pages checking this file and formatting your pages accordingly. You can therefore leave out most of the formatting tags in HTML and use only nice structural elements (like headings, paragraphs and links) — separating structure and presentation. In late 1996 CSS (Cascading Style Sheets) became a reality, forged by our good friends the » World Wide Web Consortium (W3C). Your style sheet acts as a partner to your HTML code; taking care of all the layout, fonts, colours and overall *look* of your site. If you ever decide to change the look of your site, you modify that one CSS file (your *style sheet*) and all the HTML pages reading from that file will display differently. This makes maintenance of your design much easier.

Another of CSS's boons is that you define things *once*, making it far more **efficient** than defining everything in HTML on every page.

2.2 Benefits:

- Pages download faster, sometimes by as much as 50%.
- You have to type less code, and your pages are shorter and neater.

2.3 Types of CSS:

1. Inline CSS

2. Internal CSS

3. External CSS

I. Inline CSS

In Inline CSS every style content is in HTML elements. It is used for a limited section. Whenever our requirements are very small we can use inline CSS. It will affect only single elements. In HTML we require that various HTML tags. Views are different so then we use inline Cascading Style Sheets.

There is a disadvantage of inline Cascading Style Sheets. It must be specified on every HTML tag. There is a lot of time consumed by that and it is not the best practice for a good programmer and the code will be quite large and very complex.

II. Internal CSS

In Internal CSS the style of CSS is specified in the <head> section. This is internal CSS, it affects all the elements in the body section. Internal CSS is used in the condition when we want a style to be used in the complete HTML body. For that, we can use the style in the head tag.

III. External CSS

In External CSS we create a .css file and use it in our HTML page as per our requirements. Generally, external Cascading Style Sheets are used whenever we have many HTML attributes and we can use them as required; there is no need to rewrite the CSS style again and again in a complete body of HTML that inherits the property of the CSS file.

Implementation:

CSS files are termed “cascading” style sheet because of two reasons: one style sheet can cascade, or have influence over, multiple pages. Similarly, many CSS files can define a single page.

There are 3 ways to implement css commands into your site:

- Use one CSS file for all your pages. This is the best way to do it.
- Integrate CSS commands into the head of each of your documents. Use the style attribute to put CSS code directly into a HTML element.

Advantages of CSS:

- CSS saves time – you can write CSS once and then reuse same sheet in multiple HTML pages. You can define a style for each HTML element and apply it to as many Web pages as you want.
- Pages load faster – If you are using CSS, you do not need to write HTML tag attributes every time. Just write one CSS rule of a tag and apply it to all the occurrences of that tag. So less code means faster download times.
- Easy maintenance – To make a global change, simply change the style, and all elements in all the web pages will be updated automatically.

INTRODUCTION TO JAVASCRIPT

INTRODUCTION TO JAVASCRIPT

JavaScript is a lightweight, interpreted programming language. It is designed for creating network-centric applications. It is complimentary to and integrated with Java. JavaScript is very easy to implement because it is integrated with HTML. It is open and cross-platform.

1.6 Why to Learn JavaScript?

JavaScript is a MUST for students and working professionals to become a great Software Engineer especially when they are working in Web Development Domain. JavaScript is the most popular programming language in the world and that makes it a programmer's great choice. Once you learnt JavaScript, it helps you developing great front-end as well as back-end software using different JavaScript based frameworks like query, Node.JS etc.

JavaScript is everywhere, it comes installed on every modern web browser and so to learn JavaScript you really do not need any special environment setup. For example Chrome, Mozilla Firefox, Safari and every browser you know as of today, supports JavaScript. JavaScript helps you create really beautiful and crazy fast websites. You can develop your website with a console like look and feel and give your users the best Graphical User Experience. JavaScript usage has now extended to mobile app development, desktop app development, and game development. This opens many opportunities for you as JavaScript Programmer.

Due to high demand, there is tons of job growth and high pay for those who know JavaScript. You can navigate over to different job sites to see what having JavaScript skills look like in the job market.

There are many useful JavaScript frameworks and libraries available:

1. Angular is
2. React
3. JQuery
4. Vue.js
5. Ext.js
6. Ember.js

7. Meteor
8. Mithril
9. Node.js
10. Polymer
11. Aurelia
12. Backbone.js

Applications of JavaScript Programming :

JavaScript is one of the most widely used programming languages (Front-end as well as Back-end). It has its presence in almost every area of software development.

There are some applications of JavaScript:

- 1. Client side validation** - This is really important to verify any user input before submitting it to the server and JavaScript plays an important role in validating those inputs at front-end itself.
- 2. Manipulating HTML Pages** - JavaScript helps in manipulating HTML page on the fly. This helps in adding and deleting any HTML tag very easily using JavaScript and modify your HTML to change its look and feel based on different devices and requirements.
- 3. User Notifications** - You can use JavaScript to raise dynamic pop-ups on the webpages to give different types of notifications to your website visitors.
- 4. Back-end Data Loading** - JavaScript provides Ajax library which helps in loading back- end data while you are doing some other processing. This really gives an amazing experience to your website visitors.
- 5. Presentations** - JavaScript also provides the facility of creating presentations which gives website look and feel. JavaScript provides Reveal JS and Bespoke JS libraries to build a web-based slide Presentations.
- 6. Server Applications** - Node JS is built on Chrome's JavaScript runtime for building fast and scalable network applications. This is an event Based library which helps in developing very sophisticated server applications including Web Servers.

This list goes on, there are various areas where millions of software developers are happily using JavaScript to develop great websites and others software's.

Syntax

```
<script type="text/JavaScript">
```


INTRODUCTION TO PHP

INTRODUCTION TO PHP

The PHP Hypertext Pre-processor (PHP) is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web based software applications.

1.7 Why to Learn PHP?

PHP started out as a small open source project that evolved as more and more people found out how useful it was. Rasmus Lerdorf unleashed the first version of PHP way back in 1994.

- PHP is a MUST for students and working professionals to become a great Software Engineer especially when they are working in Web Development Domain.
- PHP is a recursive acronym for "PHP: Hypertext Pre-processor".
- PHP is a server side scripting language that is embedded in HTML. It is used to manage dynamic content, databases, session tracking, even build entire e-commerce sites.
- It is integrated with a number of popular databases, including MySQL, PostgreSQL, Oracle, Sybase, Informix, and Microsoft SQL Server.
- PHP is pleasingly zippy in its execution, especially when compiled as an Apache module on the UNIX side. The MySQL server, once started, executes even very complex queries with huge result sets in record-setting time.
- PHP supports a large number of major protocols such as POP3, IMAP, and LDAP. PHP4 added support for Java and distributed object architectures (COM and CORBA), making n- tier development a possibility for the first time.
- PHP is forgiving: PHP language tries to be as forgiving as possible.
- PHP Syntax is C-Like.

1.8 Common uses of PHP:

- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.

- You add, delete, and modify elements within your database through PHP.
- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.
- It can encrypt data.

1.9 Characteristics of PHP:

Five important characteristics make PHP's practical nature possible –

- Simplicity
- Efficiency
- Security
 - Flexibility Familiarity **Syntax**

<? php//PHPcodegoeshere?>

Applications of PHP:

- PHP is one of the most widely used languages over the web. There are some applications of php.
- PHP performs system functions, i.e. from files on a system it can create, open, read, write, and close them.
- PHP can handle forms, i.e. gather data from files, save data to a file, through email you can send data, return data to the user.
- You add, delete, and modify elements within your database through PHP.
- Access cookies variables and set cookies.
- Using PHP, you can restrict users to access some pages of your website.

INTRODUCTION TO SQL

INTRODUCTION TO SQL

SQL is a database computer language designed for the retrieval and management of data in a relational database. SQL stands for Structured Query Language. SQL is the language used to create, edit and manipulate a database. In other words, SQL is used to manage data held within a relational database management system (RDBMS). Because this is a database design series, we will not be working with SQL directly, but will design our database to work with SQL in future (once it is completely designed and ready to be programmed).

SQL is the general language used to communicate with relational database management systems. This means that we use SQL to communicate to MySQL, Oracle, SQL Server, etc. ... So learning about SQL will help you with a lot of different things! A RDBMS takes SQL and uses it to do something with the database.

Why to Learn SQL?

SQL is Structured Query Language, which is a computer language for storing, manipulating and retrieving data stored in a relational database. SQL is the standard language for Relational Database System. All the Relational Database Management Systems (RDMS) like MySQL, MS Access, Oracle, Sybase, Informix, Posture's and SQL Server use SQL as their standard database language.

Also, they are using different dialects, such as –

- MS SQL Server using T-SQL,
 - Oracle using PL/SQL,
- MS Access version of SQL is called JET SQL (native format) etc.

What Can SQL do?

- SQL can execute queries against a database.
- SQL can retrieve data from a database.
- SQL can insert records in a database.
- SQL can update records in a database.
- SQL can delete records from a database.
- SQL can create new databases.
- SQL can create new tables in a database.
- SQL can create stored procedures in a database.
- SQL can create views in a database.
- SQL can set permissions on tables, procedures, and views.

Applications of SQL

SQL is one of the most widely used query language over the databases. There is some applications of SQL.

- Allows users to access data in the relational database management systems.
- Allows users to describe the data.
- Allows users to define the data in a database and manipulate that data.
- Allows to embed within other languages using SQL modules, libraries & pre-compilers.
- Allows users to create and drop databases and tables.
- Allows users to create view, stored procedure, functions in a database.
- Allows users to set permissions on tables, procedures and views.

INTRODUCTION TO PROJECT

INTRODUCTION TO PROJECT

The second-hand book resale market has grown significantly as a sustainable and cost-effective way to access knowledge. With the rising costs of education and textbooks, students often struggle to afford new books. Simultaneously, many books go unused or are discarded after a single use, creating waste and missed opportunities for reuse.

This project focuses on creating a platform for second-hand book resale, providing a solution for students and readers to buy and sell books at affordable prices. The platform bridges the gap between sellers, who wish to declutter their shelves, and buyers, who need affordable access to educational or leisure reading materials.

The project aims to promote sustainability by encouraging the reuse of books, reducing paper waste, and fostering a community of book lovers. It leverages technology to simplify the process of listing, searching, and purchasing books, ensuring convenience for users.

By addressing the financial and environmental challenges associated with the traditional book industry, this project contributes to making education and reading more accessible while promoting eco-conscious behavior.

Book resale is the practice of selling used books, often in an effort to promote sustainability, save money, and offer previously loved titles to new readers. The book resale market has grown significantly in recent years, fueled by a combination of environmental awareness, economic factors, and the love for reading. Rather than letting books gather dust on shelves, many individuals and businesses now engage in the resale of books through various channels, including online marketplaces, second-hand bookstores, garage sales, and thrift shops.

The appeal of book resale is multifaceted. For sellers, it's an opportunity to declutter and earn some extra income, while for buyers, it offers access to books at a fraction of the cost of new ones. Additionally, buying second-hand books contributes to reducing the environmental impact of producing new books, making the resale market an eco-friendly option.

This trend is not just limited to physical books but has expanded to include e-books and audio books, with platforms like eBay, Amazon, and specialized bookstores providing a space for these digital formats to be resold. With growing interest in sustainable living and a passion for literature, the book resale industry is thriving and offers numerous benefits to both sellers and buyers, as well as to the planet.

OBJECTIVE

- **Affordable Books:** Second-hand books are typically sold at significantly lower prices than new ones, making education and leisure reading more affordable for students and readers.
- **Earning Opportunity:** Sellers can earn money by selling books they no longer need, creating a win-win situation for both buyers and sellers.
- **Reduced Financial Burden:** This is especially beneficial for students who need expensive textbooks for short-term use.
- **Broader Access:** Students from low-income backgrounds can access essential educational resources at a reduced cost.
- **Rare or Out-of-Print Books:** Second-hand markets can provide access to rare, out-of-print, or older editions that are not available in mainstream bookstores.
- The facility for reselling second-hand books involves a well-organized system that ensures convenience and satisfaction for both buyers and sellers. Below are the key facilities or features that can be incorporated into a second-hand book resale platform or service:
- **Online Marketplace:** A website or app where users can list books for sale and browse for books they want to purchase.
- **Physical Stores or Kiosks:** Dedicated locations where people can drop off and buy second-hand books.
- **Quick Registration:** Allow sellers to create accounts easily.
- **Book Details:** Enable users to upload book information such as title, author, edition, condition, and price, Photo.
- **Uploads:** Allow sellers to upload photos to attract buyers.
- **Pricing Assistance:** Provide pricing suggestions based on market

SCOPE OF THE PROJECT

SCOPE OF THE PROJECT

The purpose of this project is to establish a book resale business, either through physical locations (brick-and-mortar stores) or online platforms (e.g., an e-commerce website, online marketplace). The goal is to create a sustainable, scalable operation for purchasing used books, refurbishing or categorizing them, and reselling them to a variety of customers.

FEATURE

- **User Profiles:** Buyers and sellers can create accounts to manage their listings and transactions. Ratings and reviews for sellers/buyers to build trust.
- **Search and Filter:** Advanced search options (by title, author, genre, ISBN, etc.). Filters for price, condition, location, or language.
- **Book Listings:** Easy-to-use interface for sellers to upload book details (title, author, condition, price, photos). Condition grading system (e.g., New, Like New, Good, Acceptable).
- **Payment Options:** Secure online payments with multiple methods (credit card, digital wallets). Option for cash on delivery.
- **Delivery & Shipping:** Integration with delivery services for shipping. Pickup or drop-off options based on location.

SYSTEM ANALYSIS STUDY REPORT

SYSTEM ANALYSIS STUDY REPORT

1. Do you think you can preserve best facility in this center?

Ans: yes

2. The website is user friendly

Ans: No, it is free of cost for all.

3. How much Ram is needed for it?

Ans: 500Mb or more.

4. Which processer can support this processor?

Ans: Core2 duo and all Android phone processor.

5. Can this project edit by anyone?

Ans: No, only developers can edit it by using their username and password

6. This project had update regular?

Ans: Yes. It require updating at some time.

7. If any problem than any support will give you?

Ans.: We will provide all support need to you.

8. How much data speed required for this project?

Ans: Minimum 100 kbps speed required.

9. Are all internet browser support this site?

Ans: Yes all browser support.

HARDWARE & SOFTWARE REQUIREMENT

HARDWARE & SOFTWARE REQUIREMENT

The Software Requirements Specifications is produced at the culmination of the analysis task. The function and performance allocated to software as part of system engineering are refined by establishment a complete information description, a detailed functional and behavioral description, an indication of performance requirements and design constraints, appropriate validation criteria and other data pertinent to requirements.

1)Hardware Requirement:

Name of component	Specification
Processor	Ryzen 3 dual core
RAM	128 MB
Hard Disk	20 GB
Monitor	15'Color Monitor
Keyboard	122 keys

2)Software requirements:

Name of the component	Specification
Language	Html, CSS, JAVA script
Database	MYSQL server
Browser	Chrome
Software development kit	Java JDK 1.7 or Above
Operating system	Windows 10

DFD (DATA FLOW DIAGRAM)

DFD (DATA FLOW DIAGRAM)

It is easy to understand the flow of data through system with the right data flow diagram software. A data flow diagram (DFD) maps out the flow of information for any process or system. It uses defined symbols like rectangles, circles, and arrows, plus short text labels, to show data inputs, outputs, storage point and the routes between each destination. Data flowchart can range from simple, even hand-drawn process overviews, to in-depth, multi- level DFDs that dig progressively deeper into how the data is handled. They can be used to analyses an existing system or model a new one.



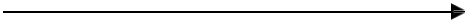
Symbols and Notations Used in DFDs

The symbols depict the four components of data flow diagram: -

External entity :- An outside system that sends or receives data, communication with the system being diagrammed. They are the sources and destinations of information entering or leaving the system. They might be an outside organization or person, a computer system or a business system. They might also known as terminators, sources and sinks or actors. They are typically drawn on edges of diagram.

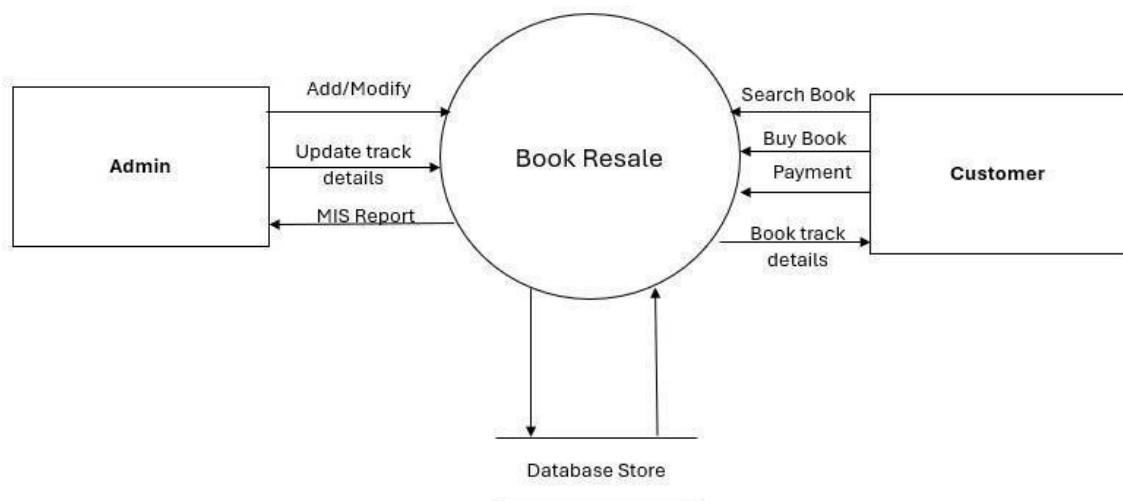
Data Store: Files or repositories that hold information for later use, such as a database table or only the register person can get appointment form.

Data flow: the route that data takes between the external entities, processes and data stores. It portrays the interface between the other components and is shown with arrows, typically Labelled with a short data name, like “Appointment details.”

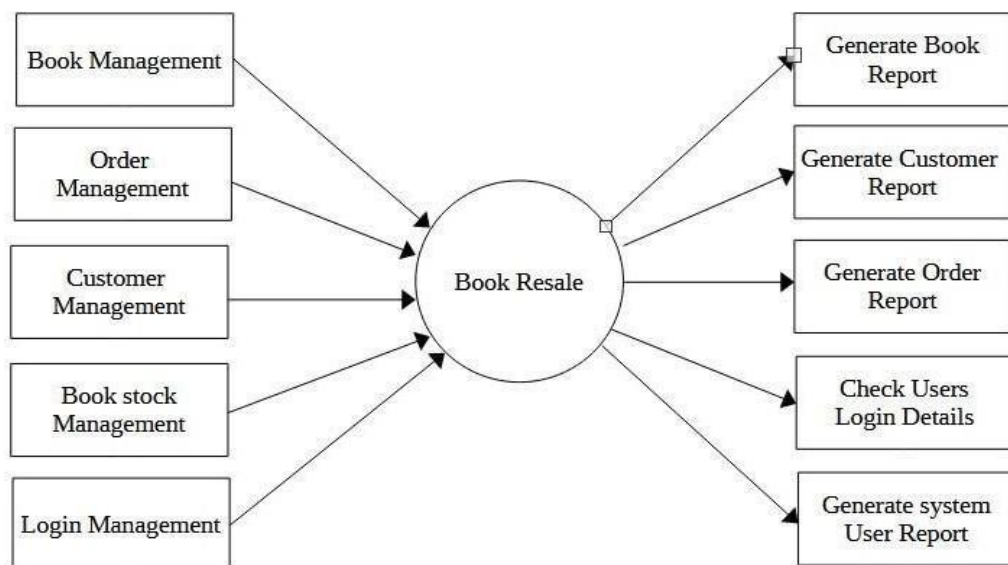
Notation	Symbol
External Entity	
Process	
Data Flow	

DATA FLOW DIAGRAM

Zero level diagram



1-Level Data Flow Diagram



ENTITY RELATIONSHIP DIAGRAM (E-R)

ENTITY RELATIONSHIP DIAGRAM (E-R)

The ER or (Entity Relational Model) is a high-level conceptual data model diagram. Entity-Relation model is based on the notion of real-world entities and their relationship between them. ER modelling helps you to analyse data requirements systematically to produce a well-designed database. So, it is considered a best practice to complete ER modelling before implementing your database.

Components of the ER Diagram

This model is based on three basic concepts:

- Entities
- Attributes
- Relationships

Entities:

A real-world thing either living or non-living that is easily recognizable and not recognizable. It is anything in the enterprise that is to be represented in our database. It may be a physical thing or simply a fact about the enterprise or an event that happens in the real world.

Attributes:

An attribute describes the property of an entity. An attribute is represented as Oval in an ER diagram. There are four types of attributes:

1. Key attribute.
2. Composite attribute.
3. Multivalued attribute.
4. Derived attribute.

Relationships:

Defines the numerical attributes of the relationship between two entities or entity sets.

Different types of cardinal relationships are:

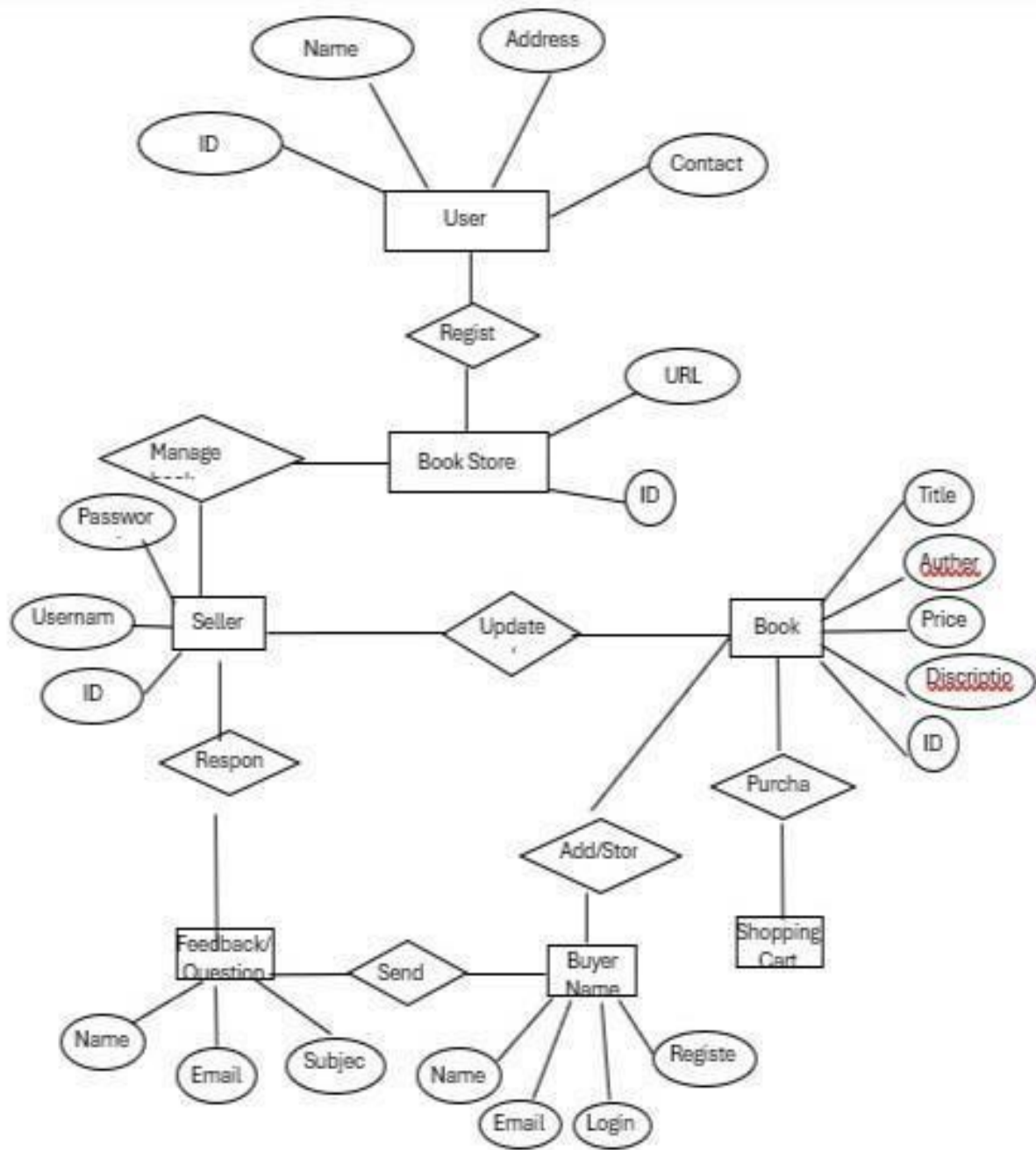
- One-to-One Relationships
- One-to-Many Relationships
- Many to One Relationships
- Many-to-Many Relationships

ER- Diagram Notations

ER- Diagram is a visual representation of data that describe how data is related to each other.

- Rectangles: This symbol represent entity types
- Ellipses: Symbol represent attributes
- Diamonds: This symbol represents relationship type
- Lines: It links attributes to entity types and entity types with other relationship types.
- Primary key: attributes are underlined
- Double Ellipses: Represent Multi-valued attribute
- Dashed Ellipses: Derived Attributes
- Double Rectangles: Weak Entity Sets
- Double Lines: Total participation of an entity in a relationship set

ER- Diagram



FEASIBILITY STUDY

FEASIBILITY STUDY

A feasibility study is carried out to select the best system that must performance requirement And it's working ability in an organization.

The feasibility of the system has been done in three Type:-

- Environmental Feasibility Study
- Economic Feasibility Study
- Technical Feasibility Study

Economic Feasibility:

In this we mainly calculate the cost of proposed system and the cost of customer and compare the cost to meet the users cost. The cost of hardware, facility cost, operating and supply costs are considered. The cost of feasibility is also considered on this such as wiring, lightning, A.C. etc.

Environmental Feasibility:

The environmental feasibility study considers both human and environmental health factors. The FS is a comparative process that looks at all potential solutions, then evaluates them against specific criteria to ultimately find the best choice.

Technical Feasibility:

Technically we have made feasibility in keeping mind about the hardware so that the designed software can work smoothly with maximum efficiency with the hardware.

TESTING

TESTING

Testing is the process of detecting errors. Testing performs a very critical role for quality assurance and for ensuring the reliability of software. The results of testing are used later on during maintenance also.

White Box Testing

White Box Testing is defined as the testing of a software solution's internal structure, design, and coding. In this type of testing, the code is visible to the tester. It focuses primarily on verifying the flow of inputs and outputs through the application, improving design and usability, strengthening security.

This is a unit testing method where a unit will be taken at a time and tested thoroughly at a statement level to find the maximum possible errors. I tested step wise every piece of code, taking care that every statement in the code is executed at least once. The white box testing is also called Glass Box Testing.

Black Box Testing

Black box testing is defined as a testing technique in which functionality of the Application under Test (AUT) is tested without looking at the internal code structure, implementation details and knowledge of internal paths of the software.

This testing method considers a module as a single unit and checks the unit at interface and communication with other modules rather getting into details at statement level. Here the module will be treated as a block box that will take some input and generate output. Output for a given set of input combinations are forwarded to other modules.

Cross browser-testing

Cross Browser Testing is a process to test web applications across multiple browsers. Cross browser testing involves checking compatibility of your application across multiple web browsers and ensures that your web application works correctly across different web browsers.

DATA DICTIONAR

DATA DICTIONAR

APPLICANT LOGIN:

COLUMN_NAME	DATA TYPE	CONSTRAINTS
EMAI	VARCHAR	UNIQUE KEY
PASSWORD	VARCHAR	UNIQUE KEY

CONTACT US:

COLUMN_NAME	DATA TYPE	CONSTRAINTS
NAME	VARCHAR	NOT NULL
EMAIL	VARCHAR	NOT NULL
MOB_NO	BIGINT	NOT NULL
MESSAGE	VARCHAR	NOT NULL

REGISTRATION :

COLUMN_NAME	DATA TYPE	CONSTRAINTS
NAME	VARCHAR	NOT NULL
EMAIL	VARCHAR	NOT NULL
CREATE_PASSWORD	VARCHAR	UNIQUE KEY
CONFORM_PASSWORD	VARCHAR	UNIQUE KEY

FEEDBACK :

COLUMN_NAME	DATA TYPE	CONSTRAINTS
NAME	VARCHAR	NOT NULL
EMAIL	VARCHAR	NOT NULL
MESSAGE	VARCHAR	NOT NULL

SOURCE CODE

SOURCE CODE

Style.css

```
@import
url('https://fonts.googleapis.com/css2?family=Quicksand:wght@300;400;500;600;700&display=sw
ap');

:root{
  --brown:#AC5743;
  --red:#c0392b;
  --orange:#f39c12;
  --black:#333;
  --white:#fff;
  --light-color:#666;
  --light-white:#ccc;
  --light-bg:#f5f5f5;
  --border:.2rem solid var(--black);
  --box-shadow:0 .5rem 1rem rgba(0,0,0,.1);
}

*{
  font-family: 'Quicksand', sans-serif;
  margin:0; padding:0;
  box-sizing: border-box;
  outline: none; border:none;
  text-decoration: none;
  transition:all .2s linear;
}

*::selection{
  background-color: var(--brown);
  color:var(--white);
}

*::-webkit-
scrollbar{ height: .
5rem; width: 1rem;
}
```

```

*::-webkit-scrollbar-
  track{ background-color:
    transparent;
}

*::-webkit-scrollbar-
  thumb{ background-color: var(--
    brown);
}

html{
  font-size: 62.5%;
  overflow-x: hidden;
  scroll-behavior: smooth;
  scroll-padding-top: 6.5rem;
}

body{
  background: var(--light-bg);
}

section{ padding:3rem 2rem;
}
.message{ position: sticky; top:0;
  margin:0 auto;
  max-width: 1200px;
  background-color: var(--light-bg);
  display: flex;
  align-items: center;
  padding:2rem;
  gap:1.5rem;
  z-index: 10000;
  justify-content: space-between;
}

.message span{

```

```

    color:var(--black);
    font-size: 2rem;
}

.message i{
    font-size: 2.5rem;
    color:var(--red);
    cursor: pointer;
}

.message
    i:hover{ transform:
    rotate(90deg);
}

.disabled{
    pointer-events:none;
    opacity: .5;
    user-select: none;
}

.btn,
.delete-btn,
.option-btn{
    display: inline-block;
    border-radius: .5rem;
    margin-top: 1rem;
    font-size: 1.8rem;
    color:var(--white);
    cursor: pointer;
    padding:1rem 3rem;
    text-transform: capitalize;
}
.btn{
    background-color: var(--brown);
}

.delete-btn{

```



```

    background-color: var(--red);
}

.option-btn{
    background-color: var(--orange);
}

.btn:hover,
.delete-btn:hover,
.option-btn:hover{
    background-color: var(--black);
}

.empty{ padding:1.5rem; font-size: 2rem;
color:var(--red);
border:var(--border);
box-shadow: var(--box-shadow);
text-align: center;
background-color: var(--white);
border-radius: .5rem;
text-transform: capitalize;
}

.more-btn{
    margin-top: 2rem;
    text-align: center;
}

.heading{ display: flex;
flex-flow: column;
align-items: center;
justify-content: center;
gap: 1rem;
background-color: var(--black);
text-align: center;

```

```
    min-height: 25vh;
}

.heading h3{
    font-size: 5.5rem;
    color:var(--white);
    text-transform: uppercase;
}
```

```
.heading p{
    font-size: 2.5rem;
    color:var(--light-white);
}
```

```
.heading p
  a{ color:var(--
    brown);
}
```

```
.heading p a:hover{
    text-decoration: underline;
}
```

```
.title{
    text-align: center;
    margin-bottom: 2rem;
    color:var(--black);
    text-transform: uppercase;
    font-size: 4rem;
}
```

```
@keyframes
  fadeIn{ 0%{
    transform: translateY(1rem);
  }
}
.header{ position:
  sticky;
```

```
top:0; left:0; right:0;
background-color: var(--white);
z-index: 1000;
box-shadow: var(--box-shadow);
}
```

```
.header .flex{ margin: 0 auto;
display: flex;
align-items: center;
justify-content: space-between;
padding:1.5rem 2rem;
max-width: 1200px;
position: relative;
}
```

```
.header .flex .logo{ font-size: 2.5rem;
color:var(--brown);
}
```

```
.header .flex .navbar
ul{ list-style: none;
}
```

```
.header .flex .navbar ul
li{ float: left;
position: relative;
}
```

```
.header .flex .navbar ul li
a{ font-size: 2rem;
padding:1rem 1.5rem;
display: block;
color:var(--light-color);
text-transform: capitalize;
}
```

```
.header .flex .navbar ul li
  a:hover{    background-color:
    var(--brown); color:var(--white);
  }
```

```
.header .flex .navbar ul li
  ul{ position: absolute;
    left:0;
    width: 15rem;
    background-color: var(--white);
    animation: fadeIn .2s linear;
    display: none;
  }
```

```
.header .flex .navbar ul li ul
  li{ width: 100%;
  }
```

```
.header .flex .navbar ul li:hover
  ul{ display: inline-block;
  }
```

```
.header-2 .flex .account-box
  { position: absolute;
    top: 60px;
    right: 10px;
    background: white;
    padding: 10px;
    box-shadow: 0px 0px 10px rgba(0,0,0,0.2);
    display: none;
  }
```

```
.header-2 .flex .account-box.active
  { display: block;
  }
```

```
.header-2 .flex .navbar.active {
```

```
display: block;
}
```

```
.header .flex .icons >
  *{ font-size: 2.5rem;
margin-left: 1rem;
color: var(--black);
cursor: pointer;
}
```

```
.header .flex .icons >
  *:hover{ color: rgb(88, 224, 88);
}
```

```
.header .flex .icons
  span{ font-size: 2rem;
}
```

```
#menu-
  btn{ display:
none;
}
```

```
.header .flex .account-
  box{ position: absolute;
top: 120%; right: 2rem;
background-color: var(--white);
border: var(--border);
text-align: center;
box-shadow: var(--box-shadow);
padding: 2rem;
border-radius: .5rem;
width: 33rem;
display: none;
animation: fadeIn .2s linear;
}
```

```
.header .flex .account-  
  box.active{ display: block;  
}
```

```
.header .flex .account-box  
  p{ padding-bottom: 1rem;  
    font-size: 2rem;  
    color:var(--light-color);  
    line-height: 1.5;  
}
```

```
.header .flex .account-box p  
  span{ color:var(--brown);  
}
```

```
.header .flex .account-box .delete-  
  btn{ margin-top: .5rem;  
}
```

```
.home{  
  min-height: 60vh;  
  background:url(uploaded_img/single3.jpg) no-repeat fixed;  
  background-size: cover;  
  background-position: center;  
  display: flex;  
  align-items: center;  
  justify-content: center;  
}
```

```
.home .content{ te  
  xt-align: center;  
}
```

```
.home .content h3{  
  text-transform: uppercase;  
  color:var(--black);  
  font-size: 4.5rem;  
}
```

```

.home .content
  p{ max-width:
    60rem;
    margin: 1rem auto;
    line-height: 2;
    font-size: 1.7rem;
    color: var(--light-color);
  }
.home .content
  a{ font-size:
    3rem;
  }
.products .box-
  container{ max-width:
    1200px; margin: 0 auto;
    display: grid;
    grid-template-columns: repeat(auto-fit, 33rem);
    gap: 1.5rem;
    align-items: flex-start;
    justify-content: center;
  }
.products .box-
  container .box{ background-
    color: var(--white); border: var(-
    -border);
    box-shadow: var(--box-shadow);
    border-radius: .5rem;
    padding: 2rem;
    text-align: center;
    position: relative;
  }
.products .box-
  container .box .image{ height:
    35rem;
    width: 100%;
    object-fit: cover;
  }

```

```
.products .box-container .box .fa-eye,
.products .box-
  container .box .price{ position:
    absolute;
    top: 1rem;
    border-radius: .5rem;
  }
```

```
.products .box-
  container .box .price{ font-size:
    2rem;
    padding: 1rem;
    background-color: var(--brown);
    color: var(--white);
    left: 1rem;
  }
```

```
.products .box-container .box .fa-
  eye{ right: 1rem;
    height: 4.5rem;
    width: 4.5rem;
    line-height: 4.2rem;
    font-size: 2rem;
    background-color: var(--white);
    color: var(--black);
    border: var(--border);
  }
```

```
.products .box-container .box .fa-
  eye: hover{ background-color: var(--
    black);
    color: var(--white);
  }
```

```
.products .box-
  container .box .name{ margin: 1rem;
    font-size: 2rem;
    color: var(--black);
  }
```

```
.products .box-container .box .qty{
```



```

width: 100%;
padding: 1.2rem 1.4rem;
font-size: 1.8rem;
color: var(--black);
border: var(--border);
border-radius: .5rem;
margin: .5rem 0;
}

.products .box-container .box .btn,
.products .box-container .box .option-
  btn{ width: 100%;
}

.home-contact{
  background-color: var(--black);
  text-align: center;
}
.home-
  contact .content{ max
    -width: 50rem;
    margin: 0 auto;
  }

.home-contact .content
  h3{ text-transform:
    uppercase; color: var(--
      white);
    font-size: 3rem;
  }
.home-contact .content
  p{ line-height: 2;
    font-size: 1.5rem;
    color: var(--light-white);
    padding: 1rem 0;
  }

.home-contact .content .btn:hover{

```

```
background-color: var(--white);
color: var(--black);
}
```

```
.about .flex{
  max-width: 1200px;
  margin: 0 auto;
  display: flex;
  align-items: center;
  flex-wrap: wrap;
  gap: 1.5rem;
}
```

```
.about .flex .image{
  flex: 1 1 40rem;
}
```

```
.about .flex .image
  img{ width: 100%;
}
```

```
.about .flex .content{
  flex: 1 1 40rem;
  text-align: center;
}
```

```
.about .flex .content
  h3{ font-size: 3rem;
      text-transform: capitalize;
      color: var(--black);
}
```

```
.about .flex .content
  p{ padding: 1rem 0;
      font-size: 1.5rem;
      color: var(--light-color);
      line-height: 2;
}
```

```
.about .flex:nth-child(2){ margin:3rem auto; flex-wrap: wrap-reverse; }
```

```
.reviews .box-container{ max-width: 1200px; margin:0 auto; display: grid; grid-template-columns: repeat(auto-fit, minmax(30rem, 1fr)); gap:1.5rem; align-items: flex-start; justify-content: center; }
```

```
.reviews .box-container .box{ padding:2rem; text-align: center; background-color: var(--white); box-shadow: var(--box-shadow); border:var(--border); border-radius: .5rem; }
```

```
.reviews .box-container .box img{ height: 10rem; width: 10rem; border-radius: 50%; }
```

```
.reviews .box-container .box p{ padding:1rem 0; font-size: 1.5rem; color:var(--light-color); line-height: 2; }
```

```
.reviews .box-  
  container .box .stars{ display:  
    inline-block; background: var(--  
      light-bg); border-radius: .5rem;  
    padding:1rem 1.5rem;  
    border:var(--border);  
  }
```

```
.reviews .box-container .box .stars  
  i{ color:var(--orange);  
    font-size: 1.7rem;  
    margin:0 .3rem;  
  }
```

```
.reviews .box-container .box  
  h3{ font-size: 2rem;  
    color:var(--black);  
    margin-top: 1rem;  
  }
```

```
.contact form{  
  max-width: 50rem;  
  background-color: var(--white);  
  border-radius: .5rem;  
  box-shadow: var(--box-shadow);  
  border:var(--border);  
  text-align: center;  
  padding:2rem;  
  margin:0 auto;  
}
```

```
.contact form  
  h3{ font-size:  
    2.5rem;  
    margin-bottom: 1rem;  
    color:var(--black);  
    text-transform: uppercase;  
  }
```

```

.contact
  form .box{ width:
    100%;
    border-radius: .5rem;
    border:var(--border);
    margin:1rem 0;
    padding:1.2rem 1.4rem;
    font-size: 1.8rem;
    color:var(--black);
    background-color: var(--light-bg);
  }

```

```

.contact form
  textarea{ height:
    15rem; resize: none;
  }

```

```

.search-form
  form{ max-width:
    1200px; margin:0
    auto; display: flex;
    align-items: center;
    gap:1rem;
  }

```

```

.search-form
  form .box{ width:
    100%; border:var(--
    border);
    background-color: var(--white);
    box-shadow: var(--box-shadow);
    padding:1.2rem 1.4rem;
    font-size: 1.8rem;
    border-radius: .5rem;
  }

```

```

.search-form
  form .btn{ margin-
    top: 0;
  }

```

```
/* */
```

```
.shopping-cart .box-  
  container{ max-width:  
    1200px; margin:0 auto;  
    display: grid;  
    grid-template-columns: repeat(auto-fit, 33rem);  
    gap:1.5rem;  
    align-items: flex-start;  
    justify-content: center;  
  }
```

```
.shopping-cart .box-  
  container .box{ background-color:  
    var(--white); border:var(--border);  
    box-shadow: var(--box-shadow);  
    border-radius: .5rem;  
    padding:2rem;  
    text-align: center;  
    position: relative;  
  }
```

```
.shopping-cart .box-  
  container .box .image{ height: 35rem;  
    width: 100%;  
    object-fit: cover;  
  }
```

```
.shopping-cart .box-container .box .fa-eye,  
.shopping-cart .box-container .box .fa-  
  times{ position: absolute;  
    top:1rem;  
    height: 4.5rem;  
    width: 4.5rem;  
    line-height: 4.3rem;  
    font-size: 2rem;  
    border-radius: .5rem;  
  }
```

```
.shopping-cart .box-container .box .fa-eye{ right:1rem;
border:var(--border);
background-color: var(--white);
color:var(--black);
}
```

```
.shopping-cart .box-container .box .fa-eye:hover{ background-color: var(--black);
color:var(--white);
}
```

```
.shopping-cart .box-container .box .fa-times{ background-color: var(--red);
color:var(--white);
}
```

```
.shopping-cart .box-container .box .fa-times:hover{ background-color: var(--black);
}
```

```
.shopping-cart .box-container .box .name{ font-size: 2rem;
color:var(--black);
margin:1rem 0;
}
```

```
.shopping-cart .box-container .box .price{ font-size: 2.5rem;
color:var(--red);
margin:.5rem 0;
}
```

```
.shopping-cart .box-container .box .qty{ width: 9rem;
padding:1.2rem 1.4rem;
border-radius: .5rem;
```

```

border:var(--border);
font-size: 1.8rem;
}

.shopping-cart .box-container .box .sub-
total{ margin-top: 1.5rem;
font-size: 2rem;
color:var(--light-color);
}

.shopping-cart .box-container .box .sub-total
span{ color:var(--red);
}

.shopping-cart .cart-
total{ max-width:
1200px; margin:0 auto;
padding:2rem;
text-align: center;
background-color: var(--white);
border:var(--border);
box-shadow: var(--box-shadow);
border-radius: .5rem;
margin-top: 2rem;
}

.shopping-cart .cart-total
p{ font-size: 2.5rem;
color:var(--light-color);
margin-bottom: 1rem;
}

.shopping-cart .cart-total p
span{ color:var(--red);
}

.display-order{
max-width: 1200px;

```



```

margin:0 auto;
text-align: center;
padding-bottom: 0;
}

.display-order
p{ display: inline-
block; border-
radius: .5rem; font-
size: 2rem; color:var(-
-black); padding:1rem
1.5rem; margin:.5rem;
background-color: var(--white);
box-shadow: var(--box-shadow);
border:var(--border);
}

.display-order p
span{ color:var(--
red);
}

.display-order .grand-
total{ margin-top: 2rem;
font-size: 2.5rem;
color:var(--light-color);
}
.display-order .grand-total
span{ color:var(--red);
}
.checkout form{
background-color: var(--white);
border:var(--border);
max-width: 1200px;
margin:0 auto;
border-radius: .5rem;
padding:2rem;

```

```

    box-shadow: var(--border);
  }
.checkout form
  h3{ padding:1.5rem;
    font-size:
    2.5rem;
    margin-bottom: 2rem;
    background-color: var(--black);
    color:var(--white);
    border-radius: .5rem;
    text-align: center;
    text-transform: capitalize;
  }

.checkout
  form .flex{ display: flex;
    flex-wrap: wrap;
    gap:1.5rem;
  }

.checkout
  form .flex .inputBox{ flex:1 1
    40rem;
  }
.checkout form .flex .inputBox input,
.checkout form .flex .inputBox
  select{ margin:1rem 0;
    width: 100%;
    background-color: var(--light-bg);
    padding:1.2rem 1.4rem;
    border:var(--border);
    font-size: 1.8rem;
    color:var(--black);
    border-radius: .5rem;
  }

.checkout form .flex .inputBox
  span{ font-size: 2rem;

```

```
    color:var(--light-color);  
}
```

```
.placed-orders .box-  
container{ max-width:  
1200px; margin:0 auto;  
display: flex;  
flex-wrap: wrap;  
gap:1.5rem;  
align-items: flex-start;  
}
```

```
.placed-orders .box-  
container .box{ background-  
color: var(--white);  
padding:1rem 2rem;  
border:var(--border);  
box-shadow: var(--box-shadow);  
flex:1 1 40rem;  
border-radius: .5rem;  
}
```

```
.placed-orders .box-container .box  
p{ margin: 1rem 0;  
font-size: 2rem;  
line-height: 1.5;  
color:var(--light-color);  
}
```

```
.placed-orders .box-container .box p  
span{ color:var(--brown);  
}
```

```
.footer{  
background-color: var(--white);  
}
```

```
.footer .box-
  container{ max-
    width: 1200px;
    margin:0 auto;
    display: grid;
    grid-template-columns: repeat(auto-fit, minmax(27rem, 1fr));
    gap:1.5rem;
    align-items: flex-start;
  }
```

```
.footer .box-container .box
  h3{ font-size: 2.2rem;
    color:var(--black);
    margin-bottom: 1.5rem;
    text-transform: uppercase;
  }
```

```
.footer .box-container a,
.footer .box-container
  p{ font-size: 1.7rem;
    color:var(--light-color);
    padding:1rem 0;
    display: block;
  }
```

```
.footer .box-container a i,
.footer .box-brownbrown p
  i{ color:var(--brown);
    padding-right: 1rem;
  }
.footer .box-container
  a:hover{ text-decoration:
    underline; color:var(--
      brown);
  }
.footer .credit{ text
  -align: center;
  padding-top: 2.5rem;
```

```

margin-top: 2.5rem;
border-top: var(--border);
font-size: 2rem;
color:var(--light-color);
}

```

```

.footer .credit
  span{ color:var(--
    brown);
}

```

```

.header .header-2 .flex .icons .user-box
  { position: absolute;
  top:120%; right:2rem;
  width: 30rem; border:
  var(--border);
  box-shadow: var(--box-shadow);
  border-radius: .5rem;
  padding:2rem;
  text-align: center;
  border-radius: .5rem;
  border:var(--border);
  background-color: var(--white);
  display: none;
  animation:fadeIn .2s linear;
}

```

```

.header .flex .icons .user-box
  p{ font-size: 2rem;
  color: var(--light-color);
  margin-bottom: 1.5rem;
}

```

```

/* Header styles */

```

```

.header { width:
  100%;

```

```
    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    background-color: #fff;
    position: relative;
    top: 0;
    z-index: 1000;
    position: sticky;
}
```

```
.header-1,
.header-2
{ display: flex;
  justify-content: space-between;
  align-items: center;
  padding: 10px 20px;
}
```

```
.header-1 {
  background-color: #f0f0f0;
}
```

```
.header-1 .flex
{ width: 100%;
  justify-content: space-between;
  align-items: center;
}
```

```
.header-1 .share a
{ margin: 0 5px;
  color: #555;
  font-size: 18px;
  text-decoration: none;
  transition: color 0.3s ease;
}
```

```
.header-1 .share a:hover
{ color: #007bff;
}
```

```

.header-1 p
{ margin: 0;
font-size: 14px;
}

.header-1 p a
{ color:
#007bff;
text-decoration: none;
font-weight: bold;
transition: color 0.3s ease;
}

.header-1 p a:hover {
text-decoration: underline;
}

.header-2
{ display: flex;
justify-content: space-between;
align-items: center;
}
/* Align logo to the left */
.header-2 .logo
{ font-size: 24px;
font-weight: bold;
text-transform: uppercase;
color: #333;
text-decoration: none;
/* Push the logo to the left */
margin-right: 25rem;
}
/* Align navbar and icons */
.navbar
{ display:
flex; gap:
20px;
}

```

```

.navbar a {
  text-decoration: none;
  color: #333;
  font-size: 16px;
  text-transform: capitalize;
  transition: color 0.3s ease;
}

.navbar a:hover
{ color:
  #007bff;
}

/* Align icons to the right */
.icons
{ display:
  flex;
  align-items: center;
  gap: 15px;
  margin-left: 25rem; /* Push icons to the right */
}

.icons i {
  font-size: 18px;
  cursor: pointer;
  color: #444d41;
  transition: color 0.3s ease;
}
.icons i:hover
{ color: #80dd80;
}
.icons a {
  position: relative;
  color: #333;
  text-decoration: none;
}

```



```

.icons a span
{ position: absolute;
top: -8px;
right: -12px;
background-color: #007bff;
color: #fff;
border-radius: 50%;
font-size: 12px;
width: 18px;
height: 18px;
display: flex;
align-items: center;
justify-content: center;
}

.header .header-2 .user-
box .active{ display: inline-block;
}

.header .header-2 .fles .icons >
*{ font-size: 2.5rem;
cursor: pointer;
}
#menu-
btn{ display:
none;
}
/* user profile show */
.header .header-2 .user-box
{ display: none;
position: absolute;
top: 60px;
right: 20px;
text-align: center;
background-color: #fff;
padding: 10px;
}

```

```

    box-shadow: 0 4px 6px rgba(0, 0, 0, 0.1);
    border-radius: 8px;
    width: 200px;
    animation: fadeIn .2s linear;
}

```

```

.user-box p
{ margin: 5px
0; font-size:
14px;

```

```

}
.user-box .delete-btn
{ display: inline-
block; margin-top:
10px; padding: 5px
10px;
background-color: #ff4d4d;
color: #fff;
text-decoration: none;
font-size: 14px;
border-radius: 4px;
transition: background-color 0.3s ease;
}

```

```

.user-box .delete-btn:hover
{ background-color:
#ff1a1a;
}

```

/* Responsive styles */

```

@media (max-width: 150px) {
    .navbar
    { display:
    none;
    }
    #menu-btn
    { display: block;
    }
}

```

```

/* media queries */

@media (max-

width:991px){ html{
  font-size: 55%;
}

}

@media (max-

width:768px){ #menu-btn{
  display: inline-block;
}

.header .flex{ padding:2rem;

}

.header .flex .navbar{ position: absolute; top:99%;
left:0; right:0;
background-color: var(--white);
border-top: var(--border);
border-bottom: var(--border);
clip-path: polygon(0 0, 100% 0, 100% 0, 0 0);
}

.header .flex .navbar.active{
  clip-path: polygon(0 0, 100% 0, 100% 100%, 0 100%);
}

.header .flex .navbar ul
li{ width: 100%;
padding:.5rem;

```

```

}

.header .flex .navbar ul li
  ul{ position: static;
    width: 100%;
    animation: none;
    background-color: var(--light-bg);
  }

.header .flex .navbar ul li ul
  li{ padding-left: 2rem;
  }

.home .content
  h3{ font-size:
    3.5rem;
  }
}

@media (max-
width:450px){ html{
  font-size: 50%;
}

.heading
  h3{ font-size:
    4rem;
  }

.title{
  font-size: 3rem;
  }
}

```

Login_style.css

```
body {  
    margin: 0;  
    padding: 0;  
    box-sizing: border-box;  
    background: url('hardhd.jpg') no-repeat center center fixed;  
    background-size: cover;  
    font-family: Arial, sans-serif;  
}
```

```
.container  
{  
    display: flex;  
    justify-content: center;  
    align-items: center;  
    height: 100vh;  
}
```

```
.form-container {  
    background: rgba(255, 255, 255, 0.1);  
    backdrop-filter: blur(10px);  
    padding: 30px;  
    border-radius: 10px;  
    width: 300px;  
    text-align: center;  
    box-shadow: 0 4px 30px rgba(0, 0, 0, 0.5);  
}
```

```
.form-container h1  
{  
    color: white;  
    margin-bottom: 20px;  
}
```

```
.form-container input  
{  
    width: 285px;  
    padding: 10px;  
    margin: 10px 0;  
    padding-left: 3px;  
    border: none;
```

```
border-radius: 5px;  
}
```

```
.form-container .password-wrapper  
{ position: relative;  
}
```

```
.form-container .password-wrapper .toggle-password  
{ position: absolute;  
left: 80%;  
top: 50%;  
width: 20%;  
transform: translateY(-50%);  
cursor: pointer;  
background: none;  
border: none;  
color: #007BFF;  
font-size: 14px;  
}
```

```
.form-container button  
{ width: 100%;  
padding: 10px;  
border: none;  
border-radius: 5px;  
background: #007BFF;  
color: white;  
font-size: 16px;  
cursor: pointer;  
transition: 0.3s;  
}
```

```
.form-container button:hover  
{ background: #0056b3;  
}
```

```
.form-container a
{ color: white;
text-decoration: none;
display: block;
margin-top: 10px;
}

.form-container a:hover
{ text-decoration:
underline;
}
.r_form{
text-decoration: none;
}
@media (max-width: 450px){
.from-
container{ hei
ght: 844px;
width: 50%;
}
}
```

Registration_form.php

```
<?php

@include 'dbconnection.php';

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

    // Sanitize and validate name: allow only alphabetic characters
    $filter_name = filter_var($_POST['name'], FILTER_SANITIZE_STRING);
    if (!preg_match('/^[a-zA-Z]+$/', $filter_name)) {
        echo "<script>alert('Name must only contain letters!');</script>";
        exit(); // Stop further execution if the name is invalid
    }
    $name = mysqli_real_escape_string($conn, $filter_name);

    // Sanitize and validate email
    $filter_email = filter_var($_POST['email'], FILTER_SANITIZE_EMAIL);
    $email = mysqli_real_escape_string($conn, $filter_email);

    // Sanitize passwords
    $filter_pass = filter_var($_POST['password'], FILTER_SANITIZE_STRING);
    $filter_cpass = filter_var($_POST['confirm_password'], FILTER_SANITIZE_STRING);

    // Email validation: check if email follows the pattern (@email.com)
    if (!preg_match('/^[a-zA-Z0-9._%+-]+@gmail\.com$/', $email)) {
        echo "<script>alert('Please enter a valid email with @email.com domain.');
```



```

        // Check if user already exists
        $select_users = mysqli_query($conn, "SELECT * FROM `users` WHERE email = '$email'")
        or die('Query failed');

        if (mysqli_num_rows($select_users) > 0) {
            echo "<script>alert('User already exists!');</script>";
        } elseif ($filter_pass != $filter_cpass) {
            echo "<script>alert('Confirm password does not match!');</script>";
        } else {
            // Insert the user with hashed password
            mysqli_query($conn, "INSERT INTO `users` (name, email, password) VALUES ('$name',
            '$email', '$pass')") or die('Query failed');
            echo "<script>
                alert('Registered successfully!');
                window.location.href = 'LOGIN.php';
            </script>";
            exit(); // Ensure script stops execution after redirection
        }
    }
}
?>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Register Page</title>
    <link rel="stylesheet" href="login_style.css">
    <style>
        .form-container .button
        { width: 100%;
        padding: 10px;
        border: none;
        border-radius: 5px;
        background: #007BFF;
        color: white;
        font-size: 16px;
    }
    }
?>

```

```

        cursor: pointer;
        transition: 0.3s;
    }
    .form-container .button:hover
    { background: #0056b3;
    }
</style>

</head>
<body>
    <?php
    if (isset($message)) {
        foreach ($message as $msg)
        { echo "
        <div class='message'>
            <span>$msg</span>
            <i class='fas fa-times' onclick='this.parentElement.remove();'></i>
        </div>
        ";
        }
    }
    ?>
    <div class="container">
        <form action="#" method="POST">
            <div class="form-container">
                <h1>Register</h1>
                <?php
                if (isset($message)) {
                    foreach ($message as $msg) {
                        echo "<div class='message'>$msg</div>";
                    }
                }
                ?>
                <input type="text" name="name" placeholder="Name" required>
                <input type="email" name="email" placeholder="Email" required>
                <div class="password-wrapper">
                    <input type="password" name="password" placeholder="Password" id="password"
required>

```

```

        <button type="button" class="toggle-password"
onclick="togglePasswordVisibility()">Show</button>
    </div>
    <div class="password-wrapper">
        <input type="password" name="confirm_password" placeholder="Confirm Password"
id="confirm_password" required>
        <button type="button" class="toggle-password"
onclick="toggleConfirmPasswordVisibility()">Show</button>
    </div>

    <input type="submit" class="button" name="submit" value="Register">
    <a href="LOGIN.php">Already have an account? Login</a>
</div>
</form>
</div>

<script>
function togglePasswordVisibility() {
    const passwordField = document.getElementById('password');
    if (passwordField.type === 'password') {
        passwordField.type = 'text';
    } else {
        passwordField.type = 'password';
    }
}

function toggleConfirmPasswordVisibility() {
    const confirmPasswordField = document.getElementById('confirm_password');
    if (confirmPasswordField.type === 'password') {
        confirmPasswordField.type = 'text';
    } else {
        confirmPasswordField.type = 'password';
    }
}
</script>
<?php

?>
</body>
</html>

```

LOGIN.php

```
<?php

@include 'dbconnection.php';

session_start();

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

    // Sanitize email and password inputs
    $filter_email = filter_var($_POST['email'], FILTER_SANITIZE_EMAIL);
    $email = mysqli_real_escape_string($conn, $filter_email);
    $filter_pass = filter_var($_POST['password'], FILTER_SANITIZE_STRING);

    // Fetch the user data based on the provided email
    $select_users = mysqli_query($conn, "SELECT * FROM `users` WHERE email = '$email'") or
    die('Query failed');

    if (mysqli_num_rows($select_users) > 0) {

        // Get the user data from the database
        $row = mysqli_fetch_assoc($select_users);

        // Verify the entered password with the stored hashed password
        if (password_verify($filter_pass, $row['password'])) {

            // If the user is an admin
            if($row['user_type'] === 'admin') {
                $_SESSION['admin_name'] = $row['name'];
                $_SESSION['admin_email'] = $row['email'];
                $_SESSION['admin_id'] = $row['id'];

                echo "<script>
                    alert('Login successful!');
                    window.location.href = 'admin_page.php';
                </script>";
                exit();
            }
        }
    }
}
```

```

// If the user is a normal user
} elseif ($row['user_type'] == 'user') {
    $_SESSION['user_name'] = $row['name'];
    $_SESSION['user_email'] = $row['email'];
    $_SESSION['user_id'] = $row['id'];

    echo "<script>
        alert('Login successful!');
        window.location.href = 'Home.php';
    </script>";
    exit();

} else {
    $message[] = 'No user found!';
}

} else {
    $message[] = 'Incorrect email or password!';
}

} else {
    $message[] = 'No user found with that email!';
}

}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Page</title>
    <link rel="stylesheet" href="login_style.css">
    <style>
        body
        { margin:
0;

```

```

padding: 0;
box-sizing: border-box;
background: url('hardhd.jpg') no-repeat center center fixed;
background-size: cover;
font-family: Arial, sans-serif;
}
</style>
</head>
<body>
<div class="container">
<div class="form-container">
<h1>LOGIN</h1>
<?php
if (isset($message)) {
    foreach ($message as $msg) {
        echo "<div class='message'>$msg</div>";
    }
}
?>
<form name="f1" onsubmit="return validation()" action="#" method="post">
<input type="email" name="email" placeholder="Email" required>
<div class="password-wrapper">
    <input type="password" name="password" placeholder="Password" id="password"
required>
    <button type="button" class="toggle-password"
onclick="togglePasswordVisibility()">Show</button>
</div>
<button name="submit" type="submit">Login</button>
</form>
<div class="Rform">
<a href="registration_form.php">Register</a>
<a href="forget_password.php">Forgot Password?</a>
</div>
</div>
</div>

<script>

```

```
function togglePasswordVisibility() {  
  const passwordField = document.getElementById('password');  
  const toggleButton = document.querySelector('.toggle-password');  
  if (passwordField.type === 'password') {  
    passwordField.type = 'text';  
    toggleButton.textContent = 'Hide';  
  } else {  
    passwordField.type = 'password';  
    toggleButton.textContent = 'Show';  
  }  
}  
</script>  
</body>  
</html>
```

Forget_password.php

```
<?php
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    include 'dbconnection.php'; // Include DB connection file

    // Get the email input from the form
    $email = $_POST['email'];

    // Email format validation
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $message[] = "Please enter a valid email!";
    } else {
        // Check if email exists in the database
        $sql = "SELECT * FROM users WHERE email = ?";
        $stmt = $conn->prepare($sql);
        $stmt->bind_param("s", $email);
        $stmt->execute();
        $result = $stmt->get_result();

        if ($result->num_rows > 0) {
            // Email exists, redirect to the new password page
            $_SESSION['email'] = $email;
            header('Location: new_password.php');
            exit(); // Make sure the script stops after redirection
        } else {
            $message[] = 'No user found!';
        }
    }
}

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
```



```

<title>Reset Password</title>
<link rel="stylesheet" href="login_style.css">
<style>
    body
    { margin:
    0;
    padding: 0;
    box-sizing: border-box;
    background: url('hardhd.jpg') no-repeat center center fixed;
    background-size: cover;
    font-family: Arial, sans-serif;
    }
</style>
</head>
<body>
    <div class="container">
        <div class="form-container">
            <h1>Reset-password</h1>
            <?php
            if (isset($message)) {
                foreach ($message as $msg) {
                    echo "<div class='message'>$msg</div>";
                }
            }
            ?>
            <form name="f1" onsubmit="return validation()" action="#" method="post">
                <input type="email" name="email" placeholder="Email" required>
                <button name="submit" type="submit">Submit</button>
                <a href="LOGIN.php">Login Page</a>

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

</body>
</html>

```

new_password.php

```
<?php
session_start();
if (!isset($_SESSION['email'])) {
    header('Location: LOGIN.php');
    exit();
}

include 'dbconnection.php'; // Include DB connection file

if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $new_password = $_POST['new_password'];
    $retype_password = $_POST['retype_password'];

    // Password validation to ensure it contains at least one special character and one number
    if (!preg_match('/^(?=.*[!@#$%^&*])(?=.*\d){8,}$/', $new_password)) {
        $message[] = "Password must be at least 8 characters long, contain at least a special character,
and a number.";
    } else {
        // Check if new password and retype password match
        if ($new_password == $retype_password) {
            // Update the password in the database
            $email = $_SESSION['email'];
            $hashed_password = password_hash($new_password, PASSWORD_DEFAULT); // Secure
password hash

            $sql = "UPDATE users SET password = ? WHERE email = ?";
            $stmt = $conn->prepare($sql);
            $stmt->bind_param("ss", $hashed_password, $email);

            if ($stmt->execute())
            { echo "<script>
                alert('Password reset successfully!');
                window.location.href = 'LOGIN.php';
                </script>";
                exit(); // Make sure the script stops after redirection
            } else {
                $message[] = "Error updating password!";
            }
        }
    }
}
```

```

        }
    } else {
        $message[]="Passwords does not match!";
    }
}
}
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Login Page</title>
    <link rel="stylesheet" href="login_style.css">
    <style>
        body
        { margin:
0;
padding: 0;
box-sizing: border-box;
background: url('hardhd.jpg') no-repeat center center fixed;
background-size: cover;
font-family: Arial, sans-serif;
}
    </style>
</head>
<body>
    <div class="container">
        <div class="form-container">
            <h1>Reset-Password</h1>
            <?php
            if (isset($message)) {
                foreach ($message as $msg) {
                    echo "<div class='message'>$msg</div>";
                }
            }
            ?>
            <form name="f1" action="#" method="post">

```

```
        <input type="new_password" name="new_password" placeholder="New-password"
required>
        <input type="retype_password" name="retype_password" placeholder="Retype-
Password" required>

        <button name="submit" type="submit">Reset</button>
    </form>
</div>
</div>

</body>
</html>
```

dbconnection.php

```
<?php
```

```
$servername = "localhost";
```

```
$username = "root";
```

```
$password = "";
```

```
$dbname = "responsiveform";
```

```
$conn = mysqli_connect($servername,$username,$password,$dbname);
```

```
if($conn){
```

```
    // echo"connection seccussful";
```

```
} else{
```

```
    echo"conection failed";
```

```
}
```

```
?>
```

logout.php

```
<?php
```

```
include 'dbconnection.php';
```

```
session_start();
```

```
session_unset();
```

```
session_destroy();
```

```
header('location:LOGIN.php');
```

```
?>
```

Home.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
}

if(isset($_POST['add_to_cart'])){

    $product_name = $_POST['product_name'];
    $product_price = $_POST['product_price'];
    $product_image = $_POST['product_image'];
    $product_quantity = $_POST['product_quantity'];

    $check_cart_numbers = mysqli_query($conn, "SELECT * FROM `cart` WHERE name =
'$product_name' AND user_id = '$user_id'") or die('query failed');

    if(mysqli_num_rows($check_cart_numbers) > 0){
        $message[] = 'already added to cart!';
    }else{
        mysqli_query($conn, "INSERT INTO `cart`(user_id, name, price, quantity, image)
VALUES('$user_id', '$product_name', '$product_price', '$product_quantity', '$product_image')") or
die('query failed');
        $message[] = 'product added to cart!';
    }

}

?>

<!DOCTYPE html>
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>home</title>

  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

  <!-- custom css file link -->
  <link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<section class="home">

  <div class="content">
    <h3>Buy & Sell Used Books at Affordable Prices</h3>
    <p>Join our community to discover a variety of books at great prices. Find, sell, and exchange
books online.</p>
    <a href="about.php" class="white-btn">discover more</a>
  </div>

</section>

<section class="products">

  <h1 class="title">latest products</h1>

  <div class="box-container">

    <?php
      $select_products = mysqli_query($conn, "SELECT * FROM `products` LIMIT 6") or
die('query failed');
      if(mysqli_num_rows($select_products) > 0){

```



```

        while($fetch_products = mysqli_fetch_assoc($select_products)){
            ?>
            <form action="" method="post" class="box">
                
                <div class="name"><?php echo $fetch_products['name']; ?></div>
                <div class="price">₹<?php echo $fetch_products['price']; ?></div>
                <input type="number" min="1" name="product_quantity" value="1" class="qty">
                <input type="hidden" name="product_name" value="<?php echo $fetch_products['name'];
            ?>">
                <input type="hidden" name="product_price" value="<?php echo $fetch_products['price']; ?>">
                <input type="hidden" name="product_image" value="<?php echo $fetch_products['image'];
            ?>">
                <input type="submit" value="add to cart" name="add_to_cart" class="btn">
            </form>
            <?php
                }
            }else{
                echo '<p class="empty">no products added yet!</p>';
            }
            ?>
        </div>

        <div class="load-more" style="margin-top: 2rem; text-align:center">
            <a href="shop.php" class="option-btn">load more</a>
        </div>

    </section>

    <section class="about">

        <div class="flex">

            <div class="image">
                
            </div>

            <div class="content">
                <h3>about us</h3>
            </div>
        </div>
    </section>

```

<p>Lorem ipsum dolor, sit amet consectetur adipisicing elit. Impedit quos enim minima ipsa dicta officia corporis ratione saepe sed adipisci?</p>

read more

</div>

</div>

</section>

<section class="home-contact">

<div class="content">

<h3>have any questions?</h3>

<p>Lorem ipsum, dolor sit amet consectetur adipisicing elit. Atque cumque exercitationem repellendus, amet ullam voluptatibus?</p>

Contact Us

</div>

</section>

<?php include 'footer.php'; ?>

<!-- custom js file link -->

<script src="script.js"></script>

</body>

</html>

header.php

```
<?php
include("dbconnection.php");
if(isset($message)){
    foreach($message as
        $message){ echo '
        <div class="message">
            <span>'.$message.'</span>
            <i class="fas fa-times" onclick="this.parentElement.remove();"></i>
        </div>
        ';
    }
}

// Fetch cart items count
$user_id = isset($_SESSION['user_id']) ? $_SESSION['user_id'] : 0;
$cart_rows_number = 0;

if ($user_id) {
    $select_cart_number = mysqli_query($conn, "SELECT * FROM `cart` WHERE user_id = '$user_id'" ) or die('Query failed');
    $cart_rows_number = mysqli_num_rows($select_cart_number);
}
?>

<header class="header">
    <div class="header-1">
        <div class="flex">
            <div class="share">
                <a href="#" class="fab fa-facebook-f"></a>
                <a href="#" class="fab fa-twitter"></a>
                <a href="#" class="fab fa-instagram"></a>
                <a href="#" class="fab fa-linkedin"></a>
            </div>
            <p>
                <a href="LOGIN.php">Login</a> | <a href="registration_form.php">Register</a>
            </p>
        </div>
    </div>
</header>
```

```

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

<div class="header-2">
  <div class="flex">
    <a href="home.php" class="logo">Book Resale</a>

    <nav class="navbar">
      <a href="home.php">HOME</a>
      <a href="about.php">ABOUT</a>
      <a href="shop.php">SHOP</a>
      <a href="contact.php">CONTACT</a>
      <a href="orders.php">ORDERS</a>
    </nav>

    <div class="icons">
      <div id="menu-btn" class="fas fa-bars"></div>
      <a href="search_page.php" class="fas fa-search"></a>
      <div id="user-btn" class="fas fa-user"></div>
      <a href="cart.php">
        <i class="fas fa-shopping-cart"></i>
        <span><?php echo $cart_rows_number; ?></span>
      </a>
    </div>

    <div class="account-box">
      <?php if (isset($_SESSION['user_name'])): ?>
        <p>Username: <span><?php echo $_SESSION['user_name']; ?></span></p>
        <p>Email: <span><?php echo $_SESSION['user_email']; ?></span></p>
        <a href="logout.php" class="delete-btn">Logout</a>
      <?php else: ?>
        <p>You are not logged in.</p>
      <?php endif; ?>
    </div>

  </div>
</div>

```

```
</header>
```

```
<!-- JavaScript for icons interaction -->
```

```
<script>
```

```
  let userBox = document.querySelector('.account-box');
```

```
  let navbar = document.querySelector('.navbar');
```

```
  document.querySelector('#user-btn').onclick = () =>
```

```
    { userBox.classList.toggle('active');
```

```
      navbar.classList.remove('active');
```

```
    }
```

```
  document.querySelector('#menu-btn').onclick = () =>
```

```
    { navbar.classList.toggle('active');
```

```
      userBox.classList.remove('active');
```

```
    }
```

```
  window.onscroll = () =>
```

```
    { userBox.classList.remove('active');
```

```
      navbar.classList.remove('active');
```

```
    }
```

```
</script>
```

```
<style>
```

```
</style>
```

footer.php

```
<section class="footer">
  <div class="box-container">
    <div class="box">
      <h3>quick links</h3>
      <a href="home.php">HOME</a>
      <a href="about.php">ABOUT</a>
      <a href="shop.php">SHOP</a>
      <a href="contact.php">CONTACT</a>
    </div>
    <div class="box">
      <h3>extra links</h3>
      <a href="LOGIN.php">LOGIN</a>
      <a href="registration_form.php">REGISTER</a>
      <a href="cart.php">CART</a>
      <a href="orders.php">ORDERS</a>
    </div>

    <div class="box">
      <h3>contact info</h3>
      <p> <i class="fas fa-phone"></i> +123-456-7890 </p>
      <p> <i class="fas fa-phone"></i> +111-222-3333 </p>
      <p> <i class="fas fa-envelope"></i> ssachin12059@gmail.com </p>
      <p> <i class="fas fa-map-marker-alt"></i> Chh. Sambhaji nagar, india - 431133 </p>
    </div>
    <div class="box">
      <h3>follow us</h3>
      <a href="#"> <i class="fab fa-facebook-f"></i> facebook </a>
      <a href="#"> <i class="fab fa-twitter"></i> twitter </a>
      <a href="#"> <i class="fab fa-instagram"></i> instagram </a>
      <a href="#"> <i class="fab fa-linkedin"></i> linkedin </a>
    </div>
  </div>

  <p class="credit"> &copy; copyright @ <?php echo date('Y'); ?> by <span>Mr. Sachin</span>
</p>

</section>
```

search_page.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
};

if(isset($_POST['add_to_cart'])){

    $product_name = $_POST['product_name'];
    $product_price = $_POST['product_price'];
    $product_image = $_POST['product_image'];
    $product_quantity = $_POST['product_quantity'];

    $check_cart_numbers = mysqli_query($conn, "SELECT * FROM `cart` WHERE name =
'$product_name' AND user_id = '$user_id'") or die('query failed');

    if(mysqli_num_rows($check_cart_numbers) > 0){
        $message[] = 'already added to cart!';
    }else{
        mysqli_query($conn, "INSERT INTO `cart`(user_id, name, price, quantity, image)
VALUES('$user_id', '$product_name', '$product_price', '$product_quantity', '$product_image')") or
die('query failed');
        $message[] = 'product added to cart!';
    }

};

?>

<!DOCTYPE html>
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>search page</title>

  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css">

  <!-- custom css file link -->
  <link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
  <h3>search page</h3>
  <p> <a href="home.php">home</a> / search </p>
</div>

<section class="search-form">
  <form action="" method="post">
    <input type="text" name="search" placeholder="search products..." class="box">
    <input type="submit" name="submit" value="search" class="btn">
  </form>
</section>

<section class="products" style="padding-top: 0;">

  <div class="box-container">
    <?php
      if(isset($_POST['submit'])){
        $search_item = $_POST['search'];
        $select_products = mysqli_query($conn, "SELECT* FROM `products` WHERE name LIKE '%{$search_item}%'") or die('query failed');

```



```

        if(mysqli_num_rows($select_products) >
        0){ while($fetch_product =
        mysqli_fetch_assoc($select_products)){
?>
<form action="" method="post" class="box">
    
    <div class="name"><?php echo $fetch_product['name']; ?></div>
    <div class="price">₹<?php echo $fetch_product['price']; ?></div>
    <input type="number" class="qty" name="product_quantity" min="1" value="1">
    <input type="hidden" name="product_name" value="<?php echo $fetch_product['name']; ?>">
    <input type="hidden" name="product_price" value="<?php echo $fetch_product['price']; ?>">
    <input type="hidden" name="product_image" value="<?php echo $fetch_product['image'];
?>">
    <input type="submit" class="btn" value="add to cart" name="add_to_cart">
</form>
<?php
    }
    }else{
        echo '<p class="empty">no result found!</p>';
    }
    }else{
        echo '<p class="empty">search something!</p>';
    }
?>
</div>

</section>

<?php include 'footer.php'; ?>

<!-- custom js file link -->
<script src="script\script.js"></script>

</body>
</html>

```

about.php

```
<?php

@include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
}

?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>about</title>

    <!-- font awesome cdn link -->
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

    <!-- custom admin css file link -->
    <link rel="stylesheet" href="style.css">

</head>
<body>

<?php @include 'header.php'; ?>

<section class="heading">
    <h3>about us</h3>
    <p> <a href="home.php">home</a> / about </p>
```

```
</section>
```

```
<section class="about">
```

```
<div class="flex">
```

```
<div class="image">
```

```

```

```
</div>
```

```
<div class="content">
```

```
<h3>why choose us?</h3>
```

```
<p>we make book resale easy, profitable, and eco-friendly. Whether you're decluttering  
your shelves, upgrading your collection, or simply looking to earn extra cash, we offer a seamless  
and rewarding experience..</p>
```

```
<a href="shop.php" class="btn">shop now</a>
```

```
</div>
```

```
</div>
```

```
<div class="flex">
```

```
<div class="content">
```

```
<h3>what we provide?</h3>
```

```
<p>The wide scope of books offered by us incorporates fantasies, moral stories, showed  
storybooks, reference books, general learning books, sentence structure books, shading books,  
action books, sticker books, and more.</p>
```

```
<a href="contact.php" class="btn">contact us</a>
```

```
</div>
```

```
<div class="image">
```

```

```

```
</div>
```

```
</div>
```

```
<div class="flex">
```

```
<div class="image">
```

```

</div>
```

```
<div class="content">
  <h3>who we are?</h3>
  <p>This BookStore is a meta-search engine for comparing books prices and know
availability across all popular Indian book stores.</p>
  <a href="#reviews" class="btn">clients reviews</a>
</div>
```

```
</div>
```

```
</section>
```

```
<section class="reviews" id="reviews">
```

```
<h1 class="title">client's reviews</h1>
```

```
<div class="box-container">
```

```
<div class="box">
  
  <p>Nice book with all major indian mythology stories together.</p>
  <div class="stars">
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star-half-alt"></i>
  </div>
  <h3>USER 1</h3>
</div>
```

```
<div class="box">
  
  <p>A good moral story for youthful mind. A must read story for everyone. Recommended
for understanding life in a better way.</p>
  <div class="stars">
```

```
<i class="fas fa-star"></i>
<i class="fas fa-star"></i>
<i class="fas fa-star"></i>
<i class="fas fa-star"></i>
<i class="fas fa-star-half-alt"></i>
</div>
<h3>USER 2</h3>
</div>
```

```
<div class="box">
  
  <p>The book covers the entire Ramayana in a really descriptive yet concise way. Great book
for kids to read and learn about the e Ramayana.</p>
  <div class="stars">
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star-half-alt"></i>
  </div>
  <h3>USER 3</h3>
</div>
```

```
<div class="box">
  
  <p>Very nicely deted story of Mahabharat with key events highlighted. Recommended for
children and parents to read at leisure. Ciao !</p>
  <div class="stars">
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star"></i>
    <i class="fas fa-star-half-alt"></i>
  </div>
  <h3>USER 4</h3>
</div>
```

```
<div class="box">
```

```

    
    <p>Superb novel so many things learn to short time, and if you are interested for
mahabharata this book help to learn new way of learning</p>
    <div class="stars">
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star-half-alt"></i>
    </div>
    <h3>USER 5</h3>
</div>

<div class="box">
    
    <p>There is no better way than this to introduce Indian Es to children. Written in
conversational mode with pixtures.</p>
    <div class="stars">
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star"></i>
        <i class="fas fa-star-half-alt"></i>
    </div>
    <h3>USER 6</h3>
</div>

</div>

</section>

<?php @include 'footer.php'; ?>

<script src="script.js"></script>

</body>
</html>

```

shop.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
}

if(isset($_POST['add_to_cart'])){

    $product_name = $_POST['product_name'];
    $product_price = $_POST['product_price'];
    $product_image = $_POST['product_image'];
    $product_quantity = $_POST['product_quantity'];

    $check_cart_numbers = mysqli_query($conn, "SELECT * FROM `cart` WHERE name =
'product_name' AND user_id = '$user_id'") or die('query failed');

    if(mysqli_num_rows($check_cart_numbers) > 0){
        $message[] = 'already added to cart!';
    }else{
        mysqli_query($conn, "INSERT INTO `cart`(user_id, name, price, quantity, image)
VALUES('$user_id', '$product_name', '$product_price', '$product_quantity', '$product_image')") or
die('query failed');
        $message[] = 'product added to cart!';
    }

}

?>

<!DOCTYPE html>
<html lang="en">
```

```

<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>shop</title>

  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

  <!-- custom css file link -->
  <link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
  <h3>our shop</h3>
  <p> <a href="Home.php">home</a> / shop </p>
</div>

<section class="products">

  <h1 class="title">latest products</h1>

  <div class="box-container">
    <?php
      $select_products = mysqli_query($conn, "SELECT * FROM `products`") or die('query
failed');
      if(mysqli_num_rows($select_products) >
        0){ while($fetch_products =
        mysqli_fetch_assoc($select_products)){
        ?>
        <form action="" method="post" class="box">
        
        <div class="name"><?php echo $fetch_products['name']; ?></div>

```



```

<div class="price">₹<?php echo $fetch_products['price']; ?>/></div>
<input type="number" min="1" name="product_quantity" value="1" class="qty">
<input type="hidden" name="product_name" value="<?php echo $fetch_products['name'];
?>">
<input type="hidden" name="product_price" value="<?php echo $fetch_products['price']; ?>">
<input type="hidden" name="product_image" value="<?php echo $fetch_products['image'];
?>">
<input type="submit" value="add to cart" name="add_to_cart" class="btn">
</form>
<?php
    }
} else {
    echo '<p class="empty">no products added yet!</p>';
}
?>
</div>

```

```

</section>

```

```

<?php include 'footer.php'; ?>

```

```

<!-- custom js file link -->
<script src="script.js"></script>

```

```

</body>

```

```

</html>

```

contact.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
}

if(isset($_POST['send'])){

    $name = mysqli_real_escape_string($conn, $_POST['name']);
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $number = $_POST['number'];
    $msg = mysqli_real_escape_string($conn, $_POST['message']);

    $select_message = mysqli_query($conn, "SELECT* FROM `message` WHERE name = '$name'
    AND email = '$email' AND number = '$number' AND message = '$msg'") or die('query failed');

    if(mysqli_num_rows($select_message) > 0){
        $message[] = 'message sent already!';
    }else{
        mysqli_query($conn, "INSERT INTO `message`(user_id, name, email, number, message)
        VALUES('$user_id', '$name', '$email', '$number', '$msg')") or die('query failed');
        $message[] = 'message sent successfully!';
    }

}

?>

<!DOCTYPE html>
<html lang="en">
<head>
```

```

<meta charset="UTF-8">
<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>contact</title>

<!-- font awesome cdn link -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

<!-- custom css file link -->
<link rel="stylesheet" href="style.css">
</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
    <h3>contact us</h3>
    <p> <a href="home.php">home</a> / contact </p>
</div>

<section class="contact">
    <form action="" method="post">
        <h3>say something!</h3>
        <input type="text" name="name" required placeholder="enter your name" class="box">
        <input type="email" name="email" required placeholder="enter your email" class="box">
        <input type="number" name="number" required placeholder="enter your number"
class="box">
        <textarea name="message" class="box" placeholder="enter your message" id="" cols="30"
rows="10"></textarea>
        <input type="submit" value="send message" name="send" class="btn">
    </form>
</section>
<?php include 'footer.php'; ?>

<!-- custom js file link -->
<script src="script.js"></script>
</body>
</html>

```

order.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:login.php');
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>orders</title>

    <!-- font awesome cdn link -->
    <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-awesome/6.0.0/css/all.min.css">

    <!-- custom css file link -->
    <link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
    <h3>your orders</h3>
    <p> <a href="Home.php">home</a> / orders </p>
</div>
```

```

<section class="placed-orders">

    <h1 class="title">placed orders</h1>

    <div class="box-container">

        <?php
            $order_query = mysqli_query($conn, "SELECT * FROM `orders` WHERE user_id =
            '$user_id'" ) or die('query failed');
            if(mysqli_num_rows($order_query) >
                0){ while($fetch_orders =
                mysqli_fetch_assoc($order_query)){
            ?>
            <div class="box">
                <p> placed on : <span><?php echo $fetch_orders['placed_on']; ?></span> </p>
                <p> name : <span><?php echo $fetch_orders['name']; ?></span> </p>
                <p> number : <span><?php echo $fetch_orders['number']; ?></span> </p>
                <p> email : <span><?php echo $fetch_orders['email']; ?></span> </p>
                <p> address : <span><?php echo $fetch_orders['address']; ?></span> </p>
                <p> payment method : <span><?php echo $fetch_orders['method']; ?></span> </p>
                <p> your orders : <span><?php echo $fetch_orders['total_products']; ?></span> </p>
                <p> total price : <span><?php echo $fetch_orders['total_price']; ?></span> </p>
                <p> payment status : <span style="color:<?php if($fetch_orders['payment_status'] ==
                'pending'){ echo 'red'; }else{ echo 'green'; } ?>,"><?php echo $fetch_orders['payment_status'];
                ?></span> </p>
            </div>
        <?php
        }
        }else{
            echo '<p class="empty">no orders placed yet!</p>';
        }
        ?>
    </div>
</section>

<?php include 'footer.php'; ?>
<!-- custom js file link -->
<script src="script.js"></script>
</body>
</html>

```

Cart.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if(!isset($user_id)){ header('location:
    LOGIN.php');
}

if(isset($_POST['update_cart'])){
    $cart_id = $_POST['cart_id'];
    $cart_quantity = $_POST['cart_quantity'];
    mysqli_query($conn, "UPDATE `cart` SET quantity= '$cart_quantity' WHERE id = '$cart_id'")
    or die('query failed');
    $message[] = 'cart quantity updated!';
}

if(isset($_GET['delete'])){
    $delete_id = $_GET['delete'];
    mysqli_query($conn, "DELETE FROM `cart` WHERE id = '$delete_id'") or die('query failed');
    header('location:cart.php');
}

if(isset($_GET['delete_all'])){
    mysqli_query($conn, "DELETE FROM `cart` WHERE user_id = '$user_id'") or die('query
    failed');
    header('location:cart.php');
}

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
```

```

<meta http-equiv="X-UA-Compatible" content="IE=edge">
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>cart</title>

<!-- font awesome cdn link -->
<link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

<!-- custom css file link -->
<link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
    <h3>shopping cart</h3>
    <p> <a href="Home.php">home</a> / cart </p>
</div>
<section class="shopping-cart">

    <h1 class="title">products added</h1>

    <div class="box-container">
        <?php
            $grand_total = 0;
            $select_cart = mysqli_query($conn, "SELECT * FROM `cart` WHERE user_id = '$user_id'")
or die('query failed');
            if(mysqli_num_rows($select_cart) >
                0){ while($fetch_cart =
                mysqli_fetch_assoc($select_cart)){
        ?>
        <div class="box">
            <a href="cart.php?delete=<?php echo $fetch_cart['id']; ?>" class="fas fa-times"
onclick="return confirm('delete this from cart?');"></a>
            
            <div class="name"><?php echo $fetch_cart['name']; ?></div>
            <div class="price">₹<?php echo $fetch_cart['price']; ?>/</div>

```

```

        <form action="" method="post" >
            <input type="hidden" name="cart_id" value="<?php echo $fetch_cart['id']; ?>">
            <input type="number" min="1" max="10" name="cart_quantity" value="<?php echo
$fetch_cart['quantity']; ?>" style="box-sizing: border-box; border-radius: 3px; background:rgb(223,
223, 223); font-size: 28px; width: 120px;">
            <input type="submit" name="update_cart" value="update" class="option-btn">
        </form>

        <div class="sub-total"> sub total : <span>₹<?php echo $sub_total = ($fetch_cart['quantity'] *
$fetch_cart['price']); ?>/></span> </div>
    </div>
    <?php
    $grand_total += $sub_total;
    }
    }else{
        echo '<p class="empty">your cart is empty</p>';
    }
    ?>
</div>

<div style="margin-top: 2rem; text-align:center;">
    <a href="cart.php?delete_all" class="delete-btn <?php echo ($grand_total > 1)?":'disabled'; ?>"
onclick="return confirm('delete all from cart?');">delete all</a>
</div>

<div class="cart-total">
    <p>grand total : <span>₹<?php echo $grand_total; ?>/></span></p>
    <div class="flex">
        <a href="shop.php" class="option-btn">continue shopping</a>
        <a href="checkout.php" class="btn <?php echo ($grand_total > 1)?":'disabled'; ?>">proceed to
checkout</a>
    </div>
</div>

</section>
<?php include 'footer.php'; ?>

<!-- custom js file link -->
<script src="script.js"></script>

</body>
</html>

```


checkout.php

```
<?php

include 'dbconnection.php';

session_start();

$user_id = $_SESSION['user_id'];

if (!isset($user_id))
{
    header('location:LOGIN.php');
}

if (isset($_POST['order_btn'])) {

    // Get input data
    $name = mysqli_real_escape_string($conn, $_POST['name']);
    $number = $_POST['number'];
    $email = mysqli_real_escape_string($conn, $_POST['email']);
    $method = mysqli_real_escape_string($conn, $_POST['method']);
    $address = mysqli_real_escape_string($conn, 'flat no.' . $_POST['flat'] . ', ' . $_POST['street'] . ', ' .
    $_POST['city'] . ', ' . $_POST['country'] . ' - ' . $_POST['pin_code']);
    $placed_on = date('d-M-Y');

    // Validate phone number (must be numeric and 10 digits)
    if (!preg_match('/^\d{10}$/', $number)) {
        $message[] = 'Please enter a valid 10-digit phone number.';
    }

    // Validate email (must be a valid Gmail address)
    if (!filter_var($email, FILTER_VALIDATE_EMAIL)) {
        $message[] = 'Please enter a valid email address.';
    } elseif (!preg_match('/@gmail\.com$/', $email)) {
        $message[] = 'Please enter a valid Gmail address.';
    }

    // Initialize cart total and cart products
    $cart_total = 0;
```

```

$cart_products[] = ";

// Fetch cart products for the user
$cart_query = mysqli_query($conn, "SELECT * FROM `cart` WHERE user_id = '$user_id'" ) or
die('query failed');
if (mysqli_num_rows($cart_query) > 0) {
    while ($cart_item = mysqli_fetch_assoc($cart_query)) {
        $cart_products[] = $cart_item['name'] . ' ( ' . $cart_item['quantity'] . ' ) ';
        $sub_total = ($cart_item['price'] * $cart_item['quantity']);
        $cart_total += $sub_total;
    }
}

// Prepare total products list
$total_products = implode(' , ', $cart_products);

// Check if the order already exists in the database
$order_query = mysqli_query($conn, "SELECT * FROM `orders` WHERE name = '$name'
AND number = '$number' AND email = '$email' AND method = '$method' AND address =
'$address' AND total_products = '$total_products' AND total_price = '$cart_total'" ) or die('query
failed');

// Validate if the cart is empty
if ($cart_total == 0) {
    $message[] = 'Your cart is empty.';
} else {
    // If order already exists, show message
    if (mysqli_num_rows($order_query) > 0) {
        $message[] = 'Order already placed!';
    } else {
        // Insert new order into the database
        mysqli_query($conn, "INSERT INTO `orders` (user_id, name, number, email, method,
address, total_products, total_price, placed_on) VALUES('$user_id', '$name', '$number', '$email',
'$method', '$address', '$total_products', '$cart_total', '$placed_on')" ) or die('query failed');
        $message[] = 'Order placed successfully!';
        // Clear cart after placing order
        mysqli_query($conn, "DELETE FROM `cart` WHERE user_id = '$user_id'" ) or die('query
failed');
    }
}

```

```

}
?>

<!DOCTYPE html>
<html lang="en">
<head>
  <meta charset="UTF-8">
  <meta http-equiv="X-UA-Compatible" content="IE=edge">
  <meta name="viewport" content="width=device-width, initial-scale=1.0">
  <title>checkout</title>

  <!-- font awesome cdn link -->
  <link rel="stylesheet" href="https://cdnjs.cloudflare.com/ajax/libs/font-
awesome/6.0.0/css/all.min.css">

  <!-- custom css file link -->
  <link rel="stylesheet" href="style.css">

</head>
<body>

<?php include 'header.php'; ?>

<div class="heading">
  <h3>checkout</h3>
  <p> <a href="Home.php">home</a> / checkout </p>
</div>
<section class="display-order">

  <?php
    $grand_total = 0;
    $select_cart = mysqli_query($conn, "SELECT * FROM `cart` WHERE user_id = '$user_id'")
or die('query failed');
    if(mysqli_num_rows($select_cart) >
    0){ while($fetch_cart =
    mysqli_fetch_assoc($select_cart)){
      $total_price = ($fetch_cart['price'] * $fetch_cart['quantity']);
      $grand_total += $total_price;

```

```

?>
<p><?php echo $fetch_cart['name']; ?> <span>(<?php echo '$'.$fetch_cart['price'].'-'.' x '.
$fetch_cart['quantity']; ?>)</span> </p>
<?php
}
}else{
    echo '<p class="empty">your cart is empty</p>';
}
?>
<div class="grand-total"> grand total : <span>₹<?php echo $grand_total; ?>/</span> </div>

</section>

```

```

<section class="checkout">

```

```

<form action="" method="post">
    <h3>place your order</h3>
    <div class="flex">
        <div class="inputBox">
            <span>Your name :</span>
            <input type="text" name="name" required placeholder="enter your name">
        </div>
        <div class="inputBox">
            <span>Phone number :</span>
            <input type="number" name="number" required placeholder="enter your number">
        </div>
        <div class="inputBox">
            <span>Add email :</span>
            <input type="email" name="email" required placeholder="enter your email">
        </div>
        <div class="inputBox">
            <span>Payment method :</span>
            <select name="method">
                <option value="cash on delivery">Cash on Delivery</option>
                <option value="credit card">Credit Card</option>
                <option value="paypal">Paypal</option>
                <option value="paytm">Paytm</option>
            </select>

```

```

</div>
<div class="inputBox">
  <span>Address line 01 :</span>
  <input type="number" min="0" name="flat" required placeholder="e.g. flat no.">
</div>
<div class="inputBox">
  <span>Address line 01 :</span>
  <input type="text" name="street" required placeholder="e.g. street name">
</div>
<div class="inputBox">
  <span>city :</span>
  <input type="text" name="city" required placeholder="e.g. mumbai">
</div>
<div class="inputBox">
  <span>state :</span>
  <input type="text" name="state" required placeholder="e.g. maharashtra">
</div>
<div class="inputBox">
  <span>country :</span>
  <input type="text" name="country" required placeholder="e.g. india">
</div>
<div class="inputBox">
  <span>pin code :</span>
  <input type="number" min="0" name="pin_code" required placeholder="e.g. 123456">
</div>
</div>
<input type="submit" value="Order now" class="btn" name="order_btn">
</form>

</section>

<?php include 'footer.php'; ?>

<!-- custom js file link -->
<script src="script.js"></script>

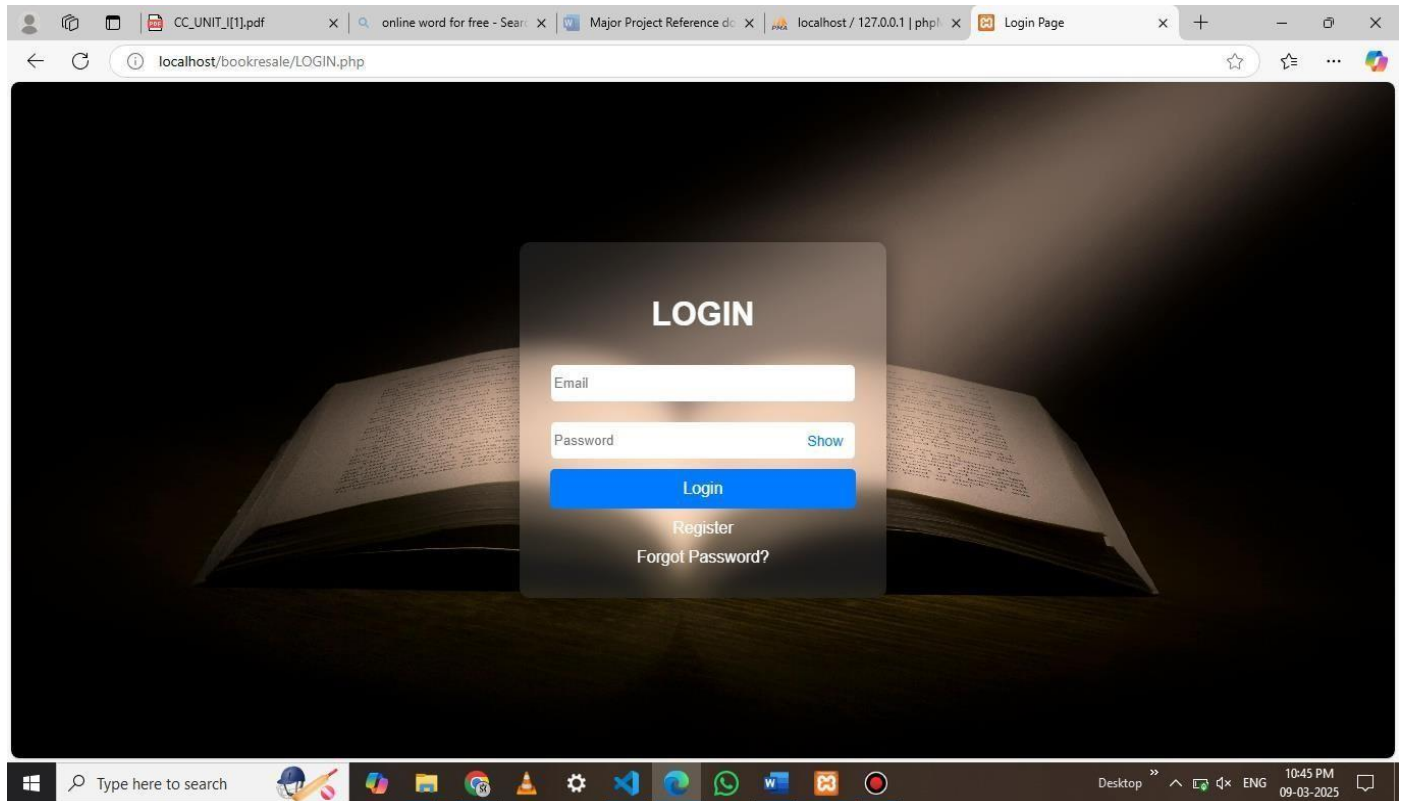
</body>
</html>

```

OUTPUT

OUTPUT

LOGIN PAGE

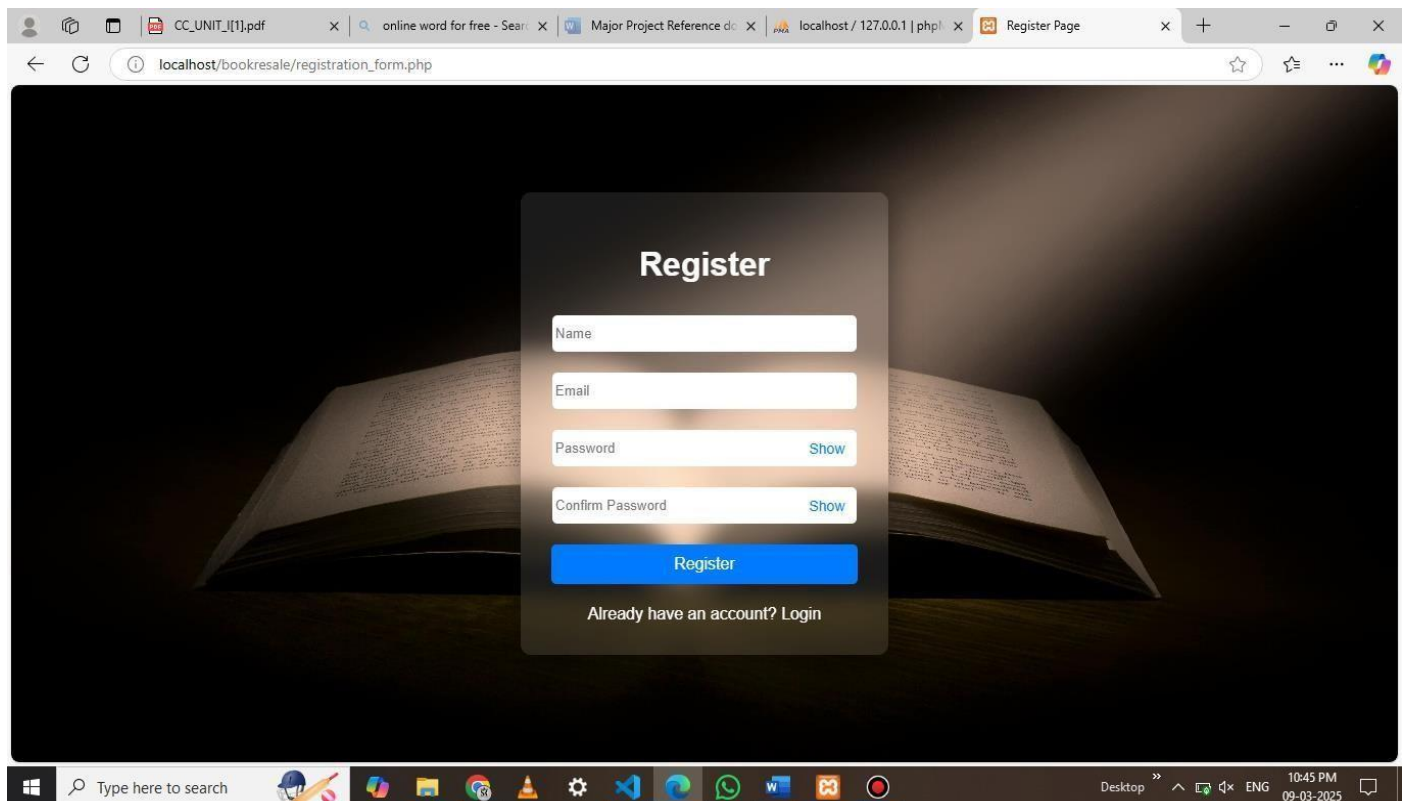


The screenshot shows a web browser window with the address bar displaying 'localhost/bookresale/LOGIN.php'. The page features a dark background with a glowing open book. A semi-transparent login form is centered on the screen. The form has a title 'LOGIN' and contains the following elements:

- An 'Email' input field.
- A 'Password' input field with a 'Show' toggle link to its right.
- A blue 'Login' button.
- A 'Register' link.
- A 'Forgot Password?' link.

The Windows taskbar at the bottom shows the search bar, task view button, and several application icons. The system clock indicates 10:45 PM on 09-03-2025.

REGISTRATION PAGE

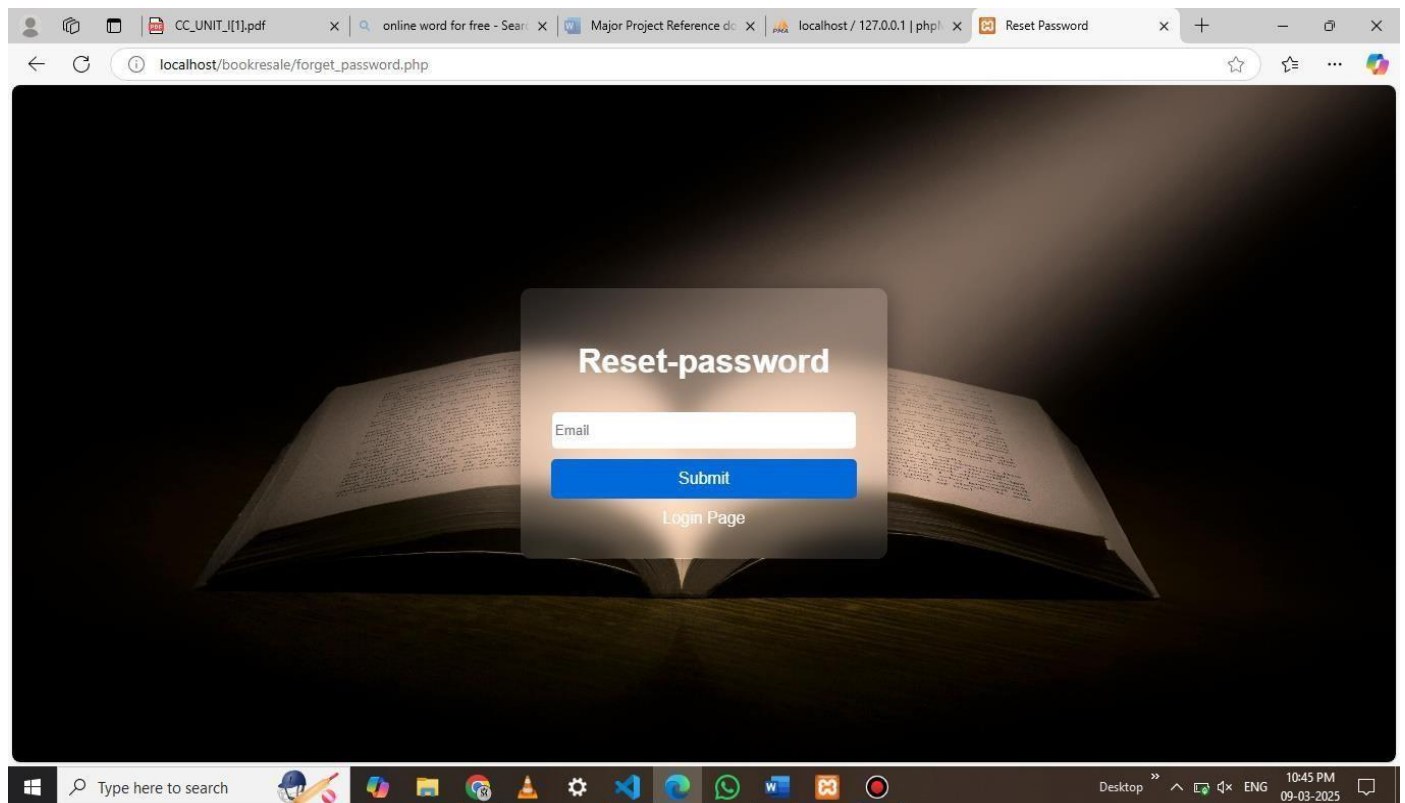


The screenshot shows a web browser window with the address bar displaying 'localhost/bookresale/registration_form.php'. The page features a dark background with a glowing open book. A semi-transparent registration form is centered on the screen. The form has a title 'Register' and contains the following elements:

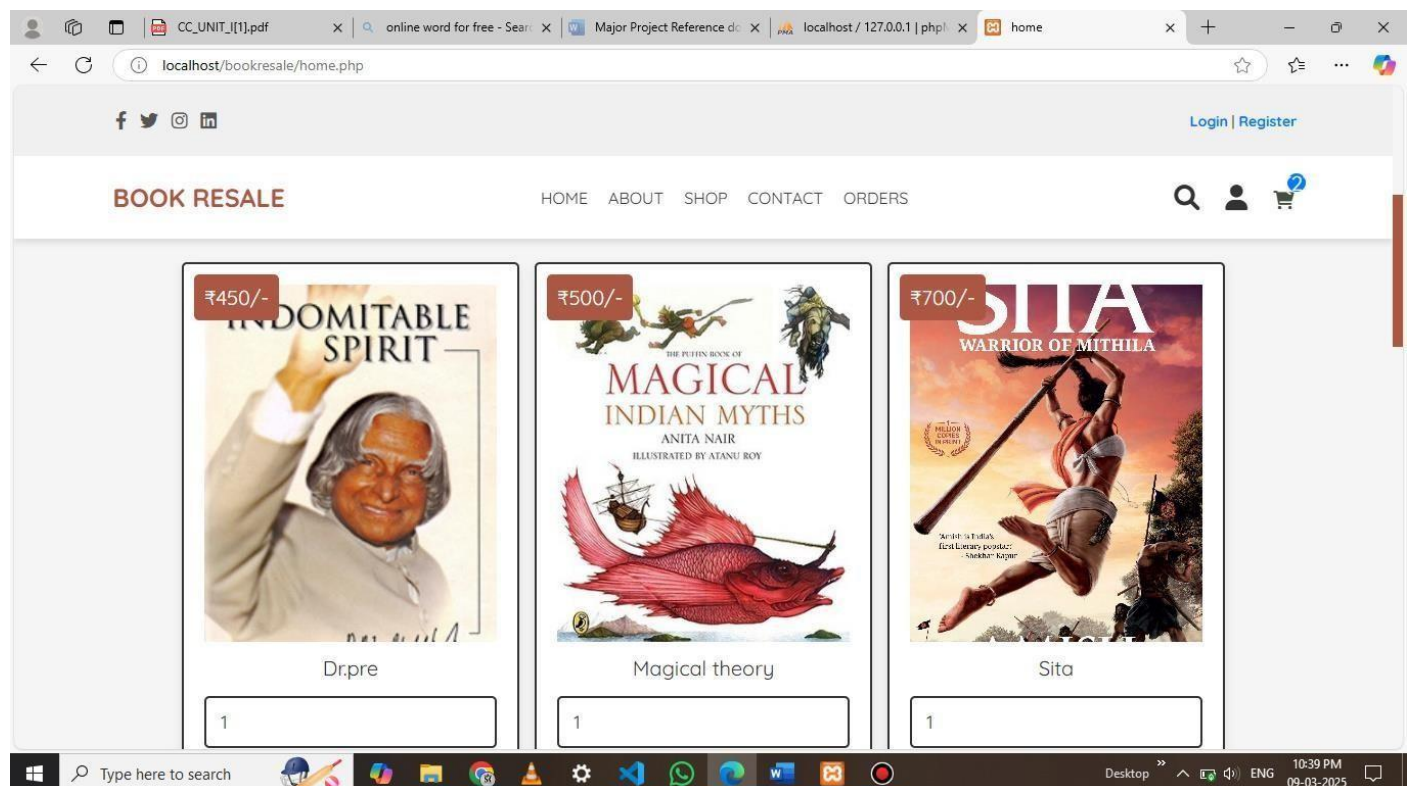
- A 'Name' input field.
- An 'Email' input field.
- A 'Password' input field with a 'Show' toggle link to its right.
- A 'Confirm Password' input field with a 'Show' toggle link to its right.
- A blue 'Register' button.
- A link that says 'Already have an account? Login'.

The Windows taskbar at the bottom shows the search bar, task view button, and several application icons. The system clock indicates 10:45 PM on 09-03-2025.

RESET PASSWORD PAGE



HOME PAGE



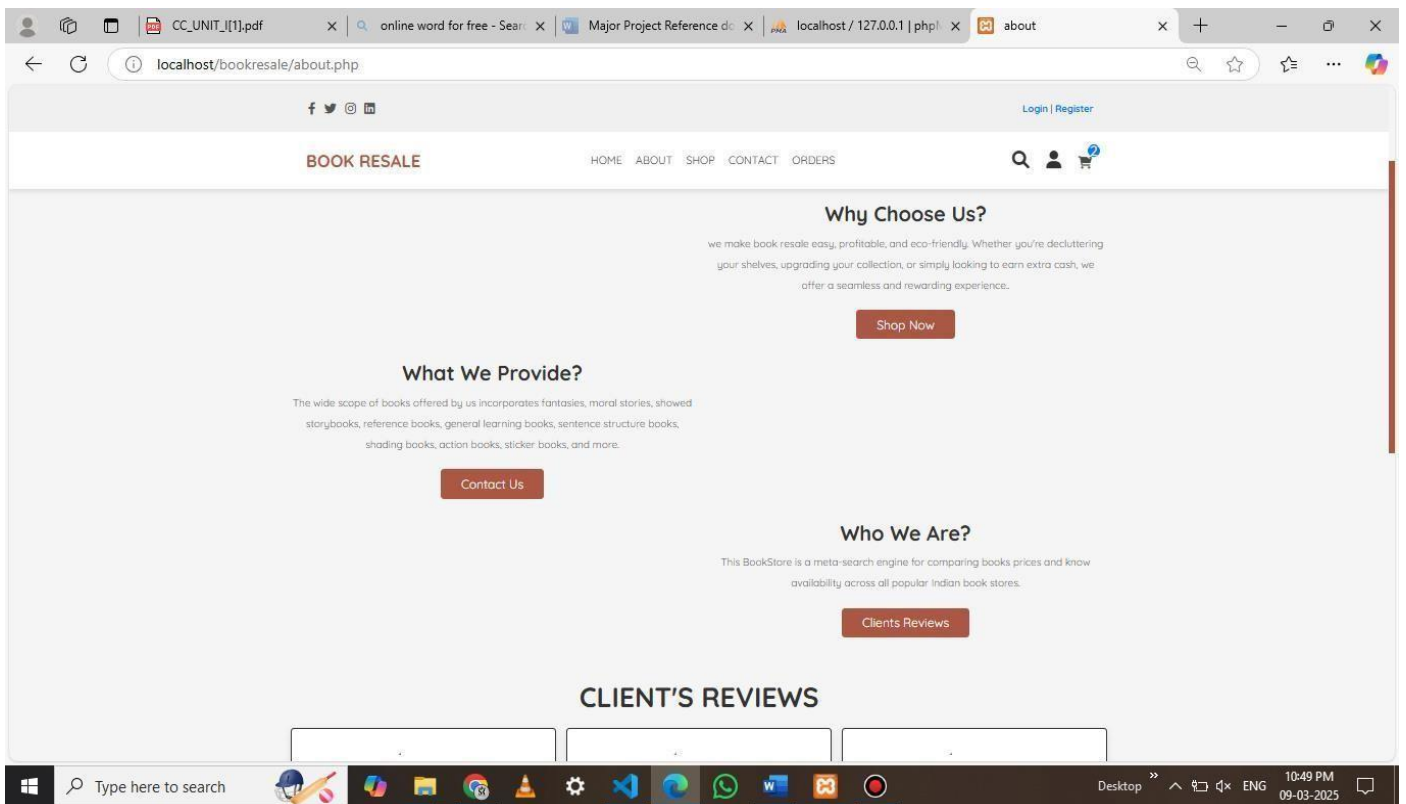
SEARCH PAGE

The screenshot shows a web browser window with the URL `localhost/bookresale/search_page.php`. The page has a header with social media icons, a `Login | Register` link, and a navigation menu with `HOME`, `ABOUT`, `SHOP`, `CONTACT`, and `ORDERS`. The main heading is `BOOK RESALE`. Below the navigation, there is a large `SEARCH PAGE` heading with a breadcrumb `home / search`. A search bar contains the placeholder text `search products...` and a `Search` button. Below the search bar, there is a button labeled `Search Something!`. The footer contains four sections: `QUICK LINKS` with `HOME` and `ABOUT`; `EXTRA LINKS` with `LOGIN` and `REGISTER`; `CONTACT INFO` with phone numbers `+123-456-7890` and `+111-222-3333`; and `FOLLOW US` with `facebook` and `twitter`. The Windows taskbar at the bottom shows the search bar, taskbar icons, and system tray with the date `09-03-2025` and time `10:42 PM`.

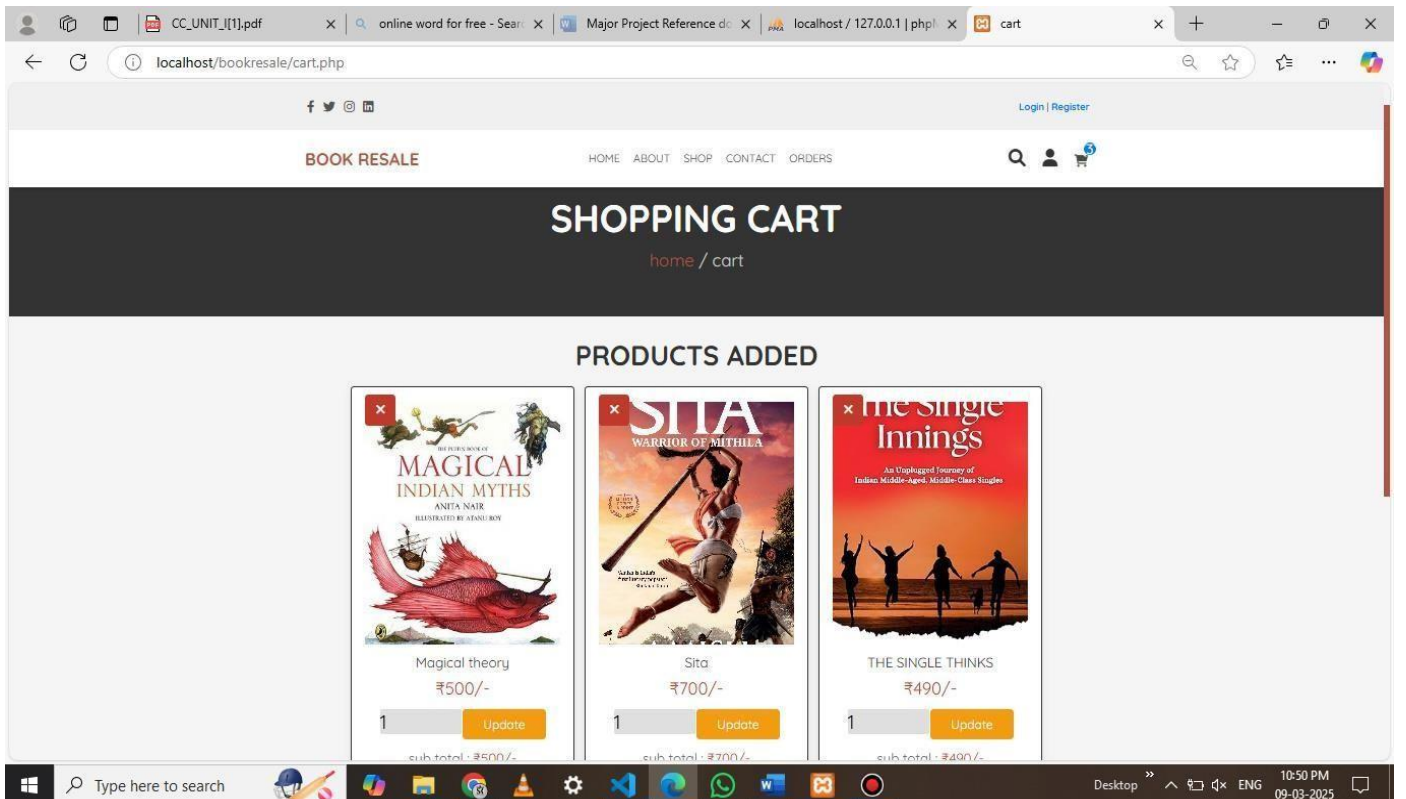
CONTACT PAGE

The screenshot shows a web browser window with the URL `localhost/bookresale/contact.php`. The page has a header with social media icons, a `Login | Register` link, and a navigation menu with `HOME`, `ABOUT`, `SHOP`, `CONTACT`, and `ORDERS`. The main heading is `BOOK RESALE`. Below the navigation, there is a large `CONTACT US` heading with a breadcrumb `home / contact`. A form titled `SAY SOMETHING!` contains four input fields: `enter your name`, `enter your email`, `enter your number`, and `enter your message`. Below the input fields is a `Send Message` button. The Windows taskbar at the bottom shows the search bar, taskbar icons, and system tray with the date `09-03-2025` and time `10:48 PM`.

ABOUT PAGE



SHOPPING CART



USRE PROFILE

The screenshot shows the home page of a website titled "BOOK RESALE". The navigation bar includes links for HOME, ABOUT, SHOP, CONTACT, and ORDERS. A search icon, a user profile icon, and a shopping cart icon with a "3" badge are on the right. The main banner features a background image of an open book with the text "BUY & SELL USED BOOKS AT AFFORDABLE PRICES". Below this, it says "Join our community to discover a variety of books at great prices. Find, buy and exchange books online." and a "discover more" link. A user profile overlay is visible on the right, showing "Username: sachin", "Email: ssachin12059@gmail.com", and a "Logout" button. The footer shows "LATEST PRODUCTS". The browser's address bar is "localhost/bookresale/home.php".

CHECKOUT PAGE

The screenshot shows the checkout page of the "BOOK RESALE" website. The navigation bar is the same as the home page. The main heading is "grand total : ₹1690/-". Below this is a "Place Your Order" form. The form contains the following fields: "Your name :", "Phone number :", "Add email :", "Payment method :", "Address line 01 :", "Address line 01 :", "city :", "state :", "country :", and "pin code :". Each field has a placeholder text. The "Payment method" dropdown is set to "Cash on Delivery". There is an "Order Now" button at the bottom of the form. The browser's address bar is "localhost/bookresale/checkout.php".

CONCLUSION

CONCLUSION

Conclusion: The Book Resale project successfully created a platform that enables users to buy and sell second-hand books with ease. Through the use of web technologies like HTML, CSS, JavaScript, PHP, and MySQL, the project provided a secure and user-friendly environment for listing books, browsing options, and managing transactions. The project not only enhanced my web development skills but also provided valuable insights into the practical aspects of e-commerce and database management.

Furthermore, this project provided valuable hands-on experience in e-commerce functionality, such as secure transaction processing and real-time updates for book listings. It not only strengthened my technical abilities but also improved my problem-solving skills as I navigated challenges related to user experience, performance optimization, and security. Overall, the Book Resale project has equipped me with a solid foundation for future web development projects and enhanced my understanding of the practical applications of web technologies in building functional online platforms.

BIBLIOGRAPHY

BIBLIOGRAPHY

- **HTML and CSS: Design and Build Websites**-By Jon Docket
- **The Essential Guide to CSS and HTML Web Design (Essentials)**-By Craig Grannel
- **Smashing CSS: Professional Techniques for Modern Layout**-By Eric Meyer
- **The Joy of PHP Programming: A Beginner's Guide**-By Alan Forbes
- **The Complete Beginner's Guide to Learn PHP Programming**-By Bruce Berke
- **New Features And Good Practices**-By Josh Lockhart
- **PHP Advanced and Object-Oriented Programming: Visual QuickPro Guide**-By Larry
Ulman
- **SQL Queries for Mere Mortals**-By John L. Viescas and Michael J. Hernandez
- **SQL Performance Tuning**-y peter gultzan
- **The Art of Sql**-By Stephane Faroult
- **Learn JavaScript VISUALLY**-By Ivelin Demirov

WEB REFERENCES

WEB REFERENCES

- <https://www.w3schools.com>
- <https://www.geeksforgeeks.org>
- <https://www.tutorialspoint.com>
- <https://www.youtube.com>
- <https://www.html-css-js.com>
- <https://www.js-fiddle.net>
- www.wikipedia.org
- www.youtube.com/webdevelopment