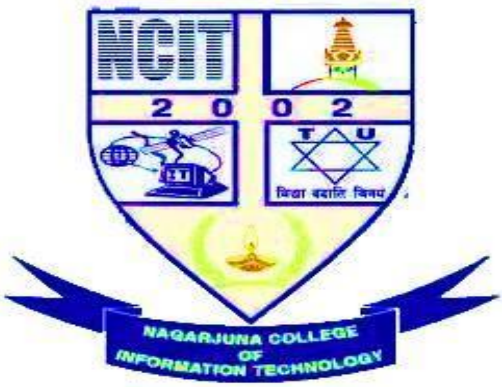
Bachelor of Science in Computer Science and Information Technology

Lab Report

For 6th SEM

E-Commerce (CSC 381)



NAGARJUNA COLLEGE OF IT

(Affiliated to Tribhuvan University)

Bagalamukhi-sankhamul.Rd, Lalitpur

Submitted by:

Nischal Gautam

Roll no:17

Submitted to:

Er. Uttam Karki

Date of Submission: 2082/03/22

Signature:

Lab No.	Titles	Date of Submission	Signature
1	Case Study On E-Commerce		
2	i) Design a Web Page of your CV with headings as Objective, educational qualification, achievements, strengths, hobbies and personal details ii) Design a table from given data iii) Design a Frame from given data		
3	Implementation of Form Validation Technique using PHP		
4	Implementation of Catalog Design		
5	Implementation of Access Control Mechanism		
6	Implementation of Session Management		
7	E-Commerce Lab Manual Using SocialGo		

LAB 1: CASE STUDY ON E-COMMERCE

Introduction

Electronic commerce, commonly referred to as *e-commerce*, signifies the buying and selling of goods and services through electronic networks—most notably, the internet. It has become a fundamental aspect of global trade, enabling individuals and businesses to conduct transactions without the limitations of geography or time. E-commerce platforms facilitate seamless commercial interactions, offering consumers access to a vast marketplace and enabling sellers to reach global audiences with relative ease.

The proliferation of smartphones, secure payment systems, and advancements in artificial intelligence have significantly boosted the adoption of e-commerce. From localized platforms such as Daraz in South Asia to global giants like Amazon and Alibaba, the digital marketplace has evolved to serve varied consumer needs while integrating sophisticated technological enhancements to optimize user experience, security, and operational efficiency.

This case study explores five major e-commerce platforms—**Daraz**, **Alibaba**, **Uber**, **Amazon**, and **SocialDeal**—by analyzing their operational frameworks, revenue generation strategies, technological features, and their role in modern commerce.

Case Studies of Selected E-Commerce Platforms

1. DARAZ

Introduction & History:

Daraz, established in 2012 in Pakistan, initially emerged as a fashion-focused online retail platform. Over time, it expanded its presence to other South Asian countries including Nepal, Bangladesh, Sri Lanka, and Myanmar. Its exponential growth attracted the attention of Alibaba Group, leading to its acquisition in 2018. This strategic move allowed Daraz to benefit from Alibaba's extensive global infrastructure, advanced technological capabilities, and deep e-commerce expertise. The acquisition also helped streamline operations, expand its marketplace offerings, and improve logistical efficiency across the region. As a result, Daraz has grown into a comprehensive e-commerce ecosystem, facilitating millions of daily transactions and serving as a digital bridge between sellers and buyers across South Asia.

Business Model:

Daraz operates on a hybrid marketplace model, supporting both Business-to-Consumer (B2C) and Business-to-Business (B2B) transactions. The platform enables third-party vendors to sell directly to customers while also maintaining its own product listings and inventory. Moreover,

Daraz has developed its proprietary logistics solution, Daraz Express (DEX), which ensures timely deliveries and helps maintain control over the supply chain. Its model effectively combines the scalability of a marketplace with the reliability of integrated logistics, catering to the region's growing e-commerce demands.

Revenue Model:

The platform generates revenue through multiple streams. These include seller commissions on product sales, delivery and service fees, advertising revenue through sponsored listings, and premium services like DarazMall, which offers verified products from top brands. Seasonal promotions, exclusive partnerships, and campaigns such as "11.11" and "Dashain Dhamaka" also contribute to revenue through increased traffic and sales volumes.

Payment Gateway:

Daraz supports a wide variety of payment options, offering flexibility and convenience to its users. These include cash on delivery (COD), bank card payments, mobile wallets such as eSewa and Khalti, and direct bank transfers. All payment methods are secured with modern encryption technologies and comply with industry standards for secure online transactions.

Security Features:

To ensure the safety and trust of its users, Daraz employs multiple layers of security including SSL encryption, two-factor authentication (2FA), identity verification for sellers, and fraud detection systems. It also enforces a strict seller review process and adheres to regional data protection laws, helping to establish a secure shopping environment.

Feature Enhancements:

Daraz continuously invests in enhancing the user experience through AI-driven recommendation engines, smart search functionality, live streaming sales, gamification elements like spin-to-win, and shipment tracking. The mobile app offers a streamlined and interactive interface that keeps users engaged while facilitating fast, efficient purchases.

2. ALIBABA

Introduction & History:

Alibaba Group was founded in 1999 by Jack Ma with the initial goal of connecting Chinese manufacturers with international buyers. It started as a B2B platform but has since grown into one of the world's most comprehensive digital commerce ecosystems. Headquartered in Hangzhou, China, Alibaba now encompasses a vast array of subsidiaries and platforms across retail, finance, logistics, and cloud computing. Its digital presence is divided among Alibaba.com for B2B, Tmall for B2C, and Taobao for C2C, each catering to distinct customer bases and

offering unique features. The group's remarkable scalability, robust technological infrastructure, and wide user base have positioned it as a global e-commerce powerhouse.

Business Model:

Alibaba utilizes a multi-platform business model, enabling it to serve multiple segments of the digital economy. Its platforms collectively cover B2B (Alibaba.com), B2C (Tmall), and C2C (Taobao) transactions. The group does not hold inventory itself but operates as a facilitator, allowing sellers to list products while providing the necessary digital infrastructure, marketing tools, and logistics support. It also owns Cainiao, a logistics data company that integrates third-party shipping solutions, and Alibaba Cloud, which offers a full suite of cloud computing services.

Revenue Model:

Alibaba's revenue is derived from various channels, including service fees charged to merchants, advertising, membership programs, transaction commissions, digital wallet service fees via Alipay, logistics services, and cloud computing services through Alibaba Cloud. This diversification ensures financial resilience and continuous growth across multiple industries.

Payment Gateway:

Alipay, Alibaba's flagship digital wallet and payment platform, supports QR code scanning, peer-to-peer transfers, and global payments. It is renowned for its security, ease of use, and wide acceptance across platforms. Alipay has become an essential tool for both domestic and international transactions, ensuring that users can make secure payments across all Alibaba-owned services.

Security Features:

Alibaba prioritizes user and merchant security with a range of protective features. These include real-time fraud detection algorithms, end-to-end data encryption, biometric user authentication, and multi-layered access control systems. The group also continuously monitors suspicious activity to ensure platform integrity and user safety.

Feature Enhancements:

Alibaba platforms are equipped with cutting-edge technologies such as augmented reality (AR) for virtual shopping experiences, AI-powered recommendation systems, voice-assisted navigation, image-based product search, real-time data analytics dashboards, and live commerce tools. These innovations improve user engagement, personalization, and transaction efficiency.

3. UBER

Introduction & History:

Uber Technologies Inc., founded in 2009 in San Francisco by Garrett Camp and Travis Kalanick, began as a revolutionary idea to transform urban transportation. Initially conceived as a simple app to hail luxury black cars, Uber quickly evolved into a global ride-hailing platform that redefined the taxi industry. Over the years, it expanded beyond ride-sharing to encompass food delivery through Uber Eats, logistics through Uber Freight, and even autonomous driving research. The core appeal of Uber lies in its on-demand service model, which provides users with real-time access to transportation and delivery services via a mobile application. Its global presence and ability to adapt to regional needs have made it a staple in urban mobility worldwide.

Business Model:

Uber's business model operates as a two-sided digital marketplace, connecting service providers (drivers, restaurants, couriers) with consumers in real-time. It does not own the vehicles or restaurants but instead provides the platform, technology, and logistics coordination necessary to facilitate seamless transactions. Uber earns a commission on each completed ride or delivery and dynamically prices services based on demand, availability, and distance. The company's model emphasizes scalability and network effects, whereby more users and drivers enhance service efficiency and reduce wait times.

Revenue Model:

Uber's revenue streams include ride fares, service fees, dynamic pricing or "surge" pricing, subscription models such as Uber One, in-app advertising, and partnerships with commercial businesses. Uber Eats contributes a significant portion of revenue through restaurant commissions and delivery charges, while Uber Freight earns from B2B logistics services.

Payment Gateway:

Uber integrates multiple payment methods, including credit and debit cards, mobile wallets (e.g., PayPal, Apple Pay, Google Pay), and cash in selected regions. All transactions are processed through secure, encrypted gateways. Payment flexibility increases accessibility for a diverse global user base.

Security Features:

Uber has invested heavily in user safety and platform integrity. The app features GPS tracking, real-time route sharing, driver background checks, trip monitoring, and an emergency contact system. Additional safeguards include two-factor authentication, user identity verification, and community-based rating systems for both riders and drivers. These features foster transparency and trust between users and service providers.

Feature Enhancements:
Uber's platform continuously evolves with innovations such as fare estimates before booking, route optimization, multi-stop trip planning, ride scheduling, and integration with local transit systems. For drivers, the app includes performance analytics, incentive programs, and navigation assistance. Voice command capabilities and wheelchair-accessible ride options further enhance inclusivity and user convenience.

4. AMAZON

Introduction & History:
Amazon.com was founded by Jeff Bezos in 1994 as an online bookstore operating out of a garage in Seattle. Today, Amazon stands as the world's largest online retailer and a global technology leader with services extending into e-commerce, cloud computing, artificial intelligence, logistics, digital streaming, and more. What began as a niche platform has grown into a customer-centric enterprise with operations in over 100 countries and a market capitalization that rivals entire national economies. Amazon's guiding philosophy—customer obsession—has driven its relentless innovation, rapid delivery systems, and expansive product offerings, making it a household name in the digital economy.

Business Model:
Amazon employs a dual business model consisting of direct retail operations and a third-party seller marketplace. While the company sells millions of products directly, it also provides a platform for independent sellers to list and sell their items. Complementary services like Amazon Prime, Fulfillment by Amazon (FBA), and Amazon Web Services (AWS) further enhance its business infrastructure. Through vertical integration, Amazon controls key stages of the value chain including storage, logistics, and customer service, which ensures high efficiency and customer satisfaction.

Revenue Model:
Amazon's revenue is highly diversified. It earns through online retail sales, subscription fees (e.g., Prime), seller commissions, advertising revenue, digital content purchases, and cloud computing services via AWS. AWS, in particular, has emerged as a major profit center, powering websites and applications for thousands of companies globally. Its advertising segment also contributes significantly by allowing sellers and brands to promote their products within Amazon's ecosystem.

Payment Gateway:
Amazon supports a wide array of secure payment methods including credit and debit cards, Amazon Pay, net banking, gift cards, and installment plans (EMI). In emerging markets, Amazon

has also adapted to include localized payment methods. Its payment infrastructure is known for its speed, reliability, and security, which enhances customer trust.

Security Features:
To maintain a safe shopping environment, Amazon implements robust security protocols including multi-factor authentication, fraud detection algorithms, data encryption, secure checkout systems, and strict compliance with global data protection standards such as GDPR. Seller verification and account protection systems are in place to ensure that buyers receive genuine products from reliable vendors.

Feature Enhancements:
Amazon continues to innovate with features such as 1-click ordering, voice shopping via Alexa, personalized product suggestions using machine learning, augmented reality (AR) previews, and seamless return policies. With initiatives like same-day delivery, drone-based shipping trials, and cashier-less stores (Amazon Go), the company consistently pushes the boundaries of retail convenience.

5. SOCIALDEAL

Introduction & History:
SocialDeal is a European-based digital platform that specializes in offering localized deals and discounts, primarily in the domains of dining, leisure, wellness, and entertainment. Unlike large e-commerce platforms that focus on global or national markets, SocialDeal is designed to support small and medium-sized enterprises (SMEs) by promoting their services to local audiences through limited-time offers. The platform has gained popularity in Western Europe by providing users with affordable experiences and businesses with effective digital marketing solutions.

Business Model:
SocialDeal operates on a discount-driven model that creates value for both vendors and consumers. Vendors list promotional offers for a limited period, which helps attract new customers and increase foot traffic to physical locations. In return, SocialDeal earns a commission for every deal purchased through its platform. This model is highly effective in competitive urban markets where price-sensitive consumers are actively looking for value-based experiences.

Revenue Model:
The company generates revenue primarily through commission-based sales, where it takes a percentage of each transaction. Additionally, SocialDeal offers premium placement and

advertising opportunities for businesses that want to enhance their visibility. This dual-income approach allows for sustainable operations while maintaining affordable prices for users.

Payment Gateway:
SocialDeal offers a secure and user-friendly payment experience through various digital options such as credit/debit cards, online banking, and mobile wallets. The payment process is streamlined and includes real-time confirmation of transactions, contributing to user satisfaction and retention.

Security Features:
The platform places a strong emphasis on data privacy and transaction security. All payments are encrypted using industry-standard protocols, and user information is handled in compliance with European data protection regulations. Additionally, SocialDeal's customer service policies ensure dispute resolution and refund processing in the event of service discrepancies.

Feature Enhancements:
SocialDeal enhances the shopping experience with mobile app access, personalized location-based deal suggestions, verified customer reviews, QR-code based voucher redemption, and seamless digital interfaces. Its real-time inventory management and easy refund mechanisms add layers of convenience for both customers and vendors.

Conclusion

The evolution of e-commerce platforms such as **Daraz**, **Alibaba**, **Uber**, **Amazon**, and **SocialDeal** represents a pivotal shift in how consumers interact with goods and services in the digital age. Each platform, while distinct in its operational scope and market focus, shares common attributes of scalability, technological integration, and customer-centric innovation. Amazon and Alibaba, for instance, dominate the global market through their expansive product offerings, advanced logistics, and strategic diversification into cloud computing and digital payments. In contrast, Daraz exemplifies the power of localized platforms backed by global expertise, offering tailored solutions to meet the unique needs of South Asian consumers.

Uber has transformed traditional mobility services by leveraging the sharing economy and digital logistics to create an ecosystem of convenience, particularly in urban environments. Meanwhile, SocialDeal illustrates the value of niche platforms in supporting regional economies and smaller vendors through hyper-local digital marketing and deal-based engagement. Collectively, these platforms reflect the growing significance of data-driven strategies, AI-enhanced personalization, secure financial infrastructure, and adaptive business models in the success of e-commerce.

In conclusion, the future of e-commerce is not limited to transactional convenience but is increasingly shaped by user experience, innovation, and trust. As digital transformation accelerates across industries and borders, these platforms will continue to play a crucial role in defining the commercial landscape of tomorrow.

NAGARJUNA COLLEGE

of Information Technology

(Affiliated to Tribhuvan University)

Shankhamul- 9 , Lalitpur, Nepal

Lab report-2 of E-commerce

Subject Code: CSC381

Submitted by:
Name: Nischal Gautam
Roll no: 17
Subject: E-Commerce

Submitted to:
Name: Er. Uttam Karki
Signature:

EXPERIMENT – 1

Q. Design a Web Page of your CV with headings as Objective, educational qualification, achievements, strengths, hobbies and personal details. Apply following specifications: Insert a horizontal line after every above-mentioned heading (Eg. Insert horizontal line once objectives are complete) Set any light color as page background. Bold and underline every heading Insert your image on left side of web page Use heading tag to specify the headings After every heading is over put a horizontal line Use pre tag for Educational Qualification Use Base font tag for all the text

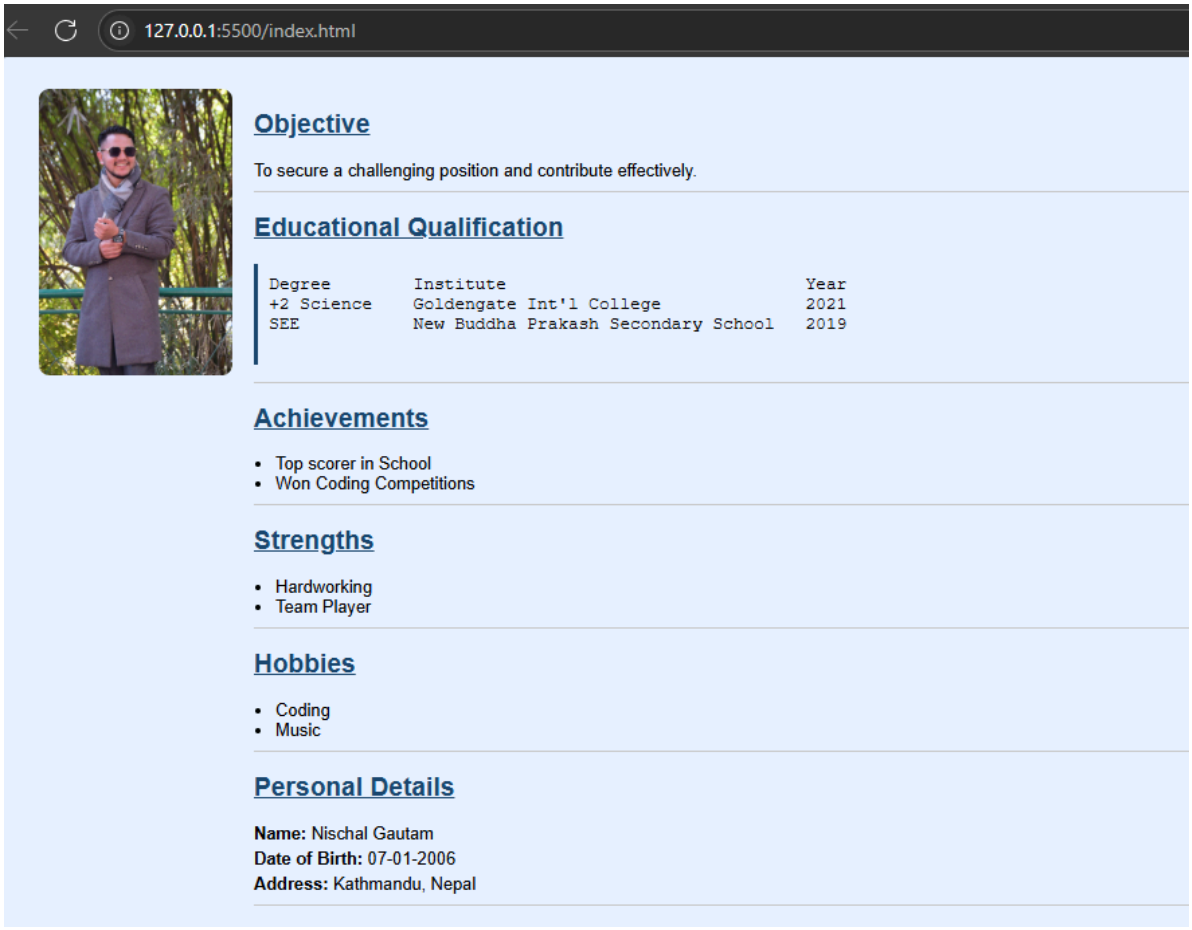
Code:

```
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Nischal Gautam - CV</title>
    <style>
        body {
            background-color: #e6f2ff; /* Light background */
            font-family: Arial, sans-serif;
            margin: 40px;
        }
        .container {
            display: flex;
            gap: 20px;
        }
        .profile-pic {
            width: 180px;
            height: auto;
            border-radius: 10px;
        }
        .content {
            flex: 1;
        }
        h2 {
            font-weight: bold;
            text-decoration: underline;
            color: #1a4a72;
```

```
        }
        hr {
            border: none;
            border-top: 2px solid #ccc;
            margin: 10px 0 20px 0;
        }
        pre {
            background-color: #e6f2ff; /* Light background */
            padding: 10px;
            border-left: 4px solid #1a4a72;
            font-family: Courier, monospace;
        }
        ul {
            margin: 0;
            padding-left: 20px;
        }
        p {
            margin: 5px 0;
        }
    </style>
</head>
<body>
    <basefont face="Arial" size="3" color="black">

    <div class="container">
        <div>
            <img src="WhatsApp Image 2025-06-30 at 13.53.57_1ae9d271.jpg" alt="My
Image" class="profile-pic">
        </div>

        <div class="content">
            <h2>Objective</h2>
            <p>To secure a challenging position and contribute effectively.</p>
            <hr>

            <h2>Educational Qualification</h2>
            <pre>
Degree        Institute                          Year
+2 Science    Goldengate Int'l College           2021
SEE           New Buddha Prakash Secondary School   2019
```

```
        </pre>
        <hr>

        <h2>Achievements</h2>
        <ul>
            <li>Top scorer in School</li>
            <li>Won Coding Competitions</li>
        </ul>
        <hr>

        <h2>Strengths</h2>
        <ul>
            <li>Hardworking</li>
            <li>Team Player</li>
        </ul>
        <hr>

        <h2>Hobbies</h2>
        <ul>
            <li>Coding</li>
            <li>Music</li>
        </ul>
        <hr>

        <h2>Personal Details</h2>
        <p><strong>Name:</strong> Nischal Gautam</p>
        <p><strong>Date of Birth:</strong> 07-01-2006</p>
        <p><strong>Address:</strong> :Kathmandu, Nepal</p>
        <hr>
      </div>
    </div>

</body>
</html>
```

Output:


127.0.0.1:5500/index.html
Objective
To secure a challenging position and contribute effectively.
Educational Qualification
Degree Institute Year
+2 Science Goldengate Int'l College 2021
SEE New Buddha Prakash Secondary School 2019
Achievements
Top scorer in School
Won Coding Competitions
Strengths
Hardworking
Team Player
Hobbies
Coding
Music
Personal Details
Name: Nischal Gautam
Date of Birth: 07-01-2006
Address: Kathmandu, Nepal

EXPERIMENT – 2

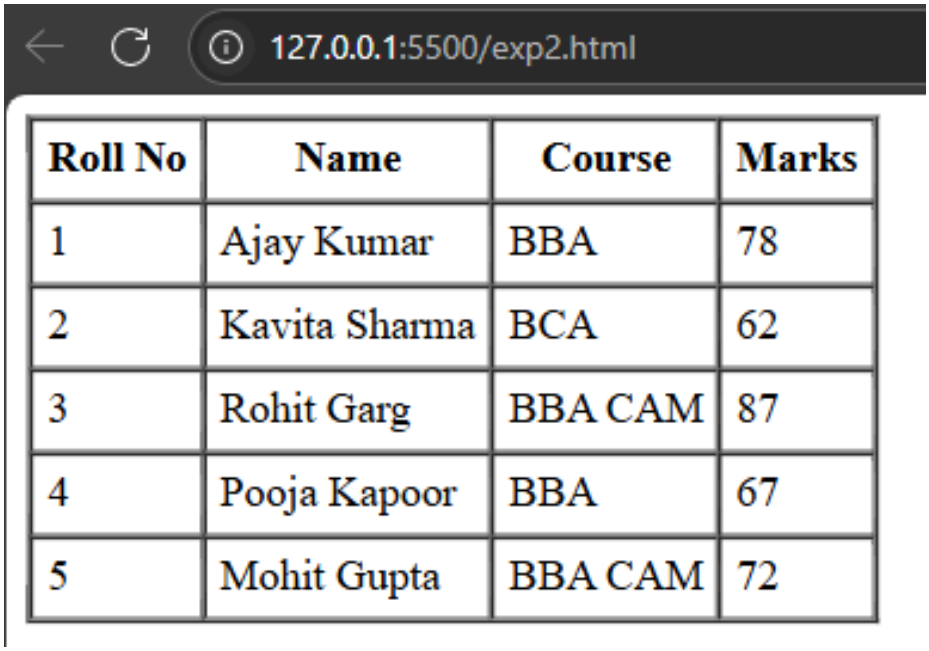
Q. Design tables as follows:

Roll no	Name	Course	Marks
1	Ajay Kumar	BBA	78
2	Kavita Sharma	BCA	62
3	Rohit Garg	BBA CAM	87
4	Pooja Kapoor	BBA	67
5	Mohit Gupta	BBA CAM	72

Code:

```
<!DOCTYPE html>
<html>
<head><title>Student Table</title></head>
<body>
   <table border="1" cellpadding="5" cellspacing="0">
      <tr>
         <th>Roll No</th>
         <th>Name</th>
         <th>Course</th>
         <th>Marks</th>
      </tr>
      <tr><td>1</td><td>Ajay Kumar</td><td>BBA</td><td>78</td></tr>
      <tr><td>2</td><td>Kavita Sharma</td><td>BCA</td><td>62</td></tr>
      <tr><td>3</td><td>Rohit Garg</td><td>BBA CAM</td><td>87</td></tr>
      <tr><td>4</td><td>Pooja Kapoor</td><td>BBA</td><td>67</td></tr>
      <tr><td>5</td><td>Mohit Gupta</td><td>BBA CAM</td><td>72</td></tr>
   </table>
</body>
</html>
```

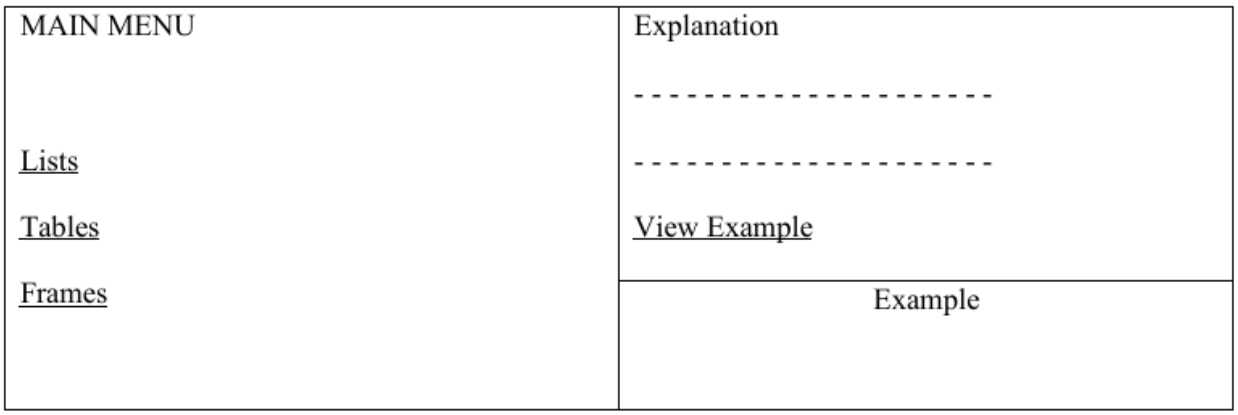
Output:



Roll No	Name	Course	Marks
1	Ajay Kumar	BBA	78
2	Kavita Sharma	BCA	62
3	Rohit Garg	BBA CAM	87
4	Pooja Kapoor	BBA	67
5	Mohit Gupta	BBA CAM	72

EXPERIMENT-3

Q. Design following frame:



Code:

Main.html:

```
<!DOCTYPE html>
<html>
<head>
   <title>Lab 3 - Frames</title>
</head>
<frameset cols="25%, 75%">
   <!-- Left Menu -->
   <frame src="menu.html" name="menu">

   <!-- Right Side: Explanation (top) and Example (bottom) -->
   <frameset rows="60%, 40%">
      <frame src="explanation.html" name="explanation">
      <frame src="example.html" name="example">
   </frameset>
</frameset>
</html>
```

Menu.html:

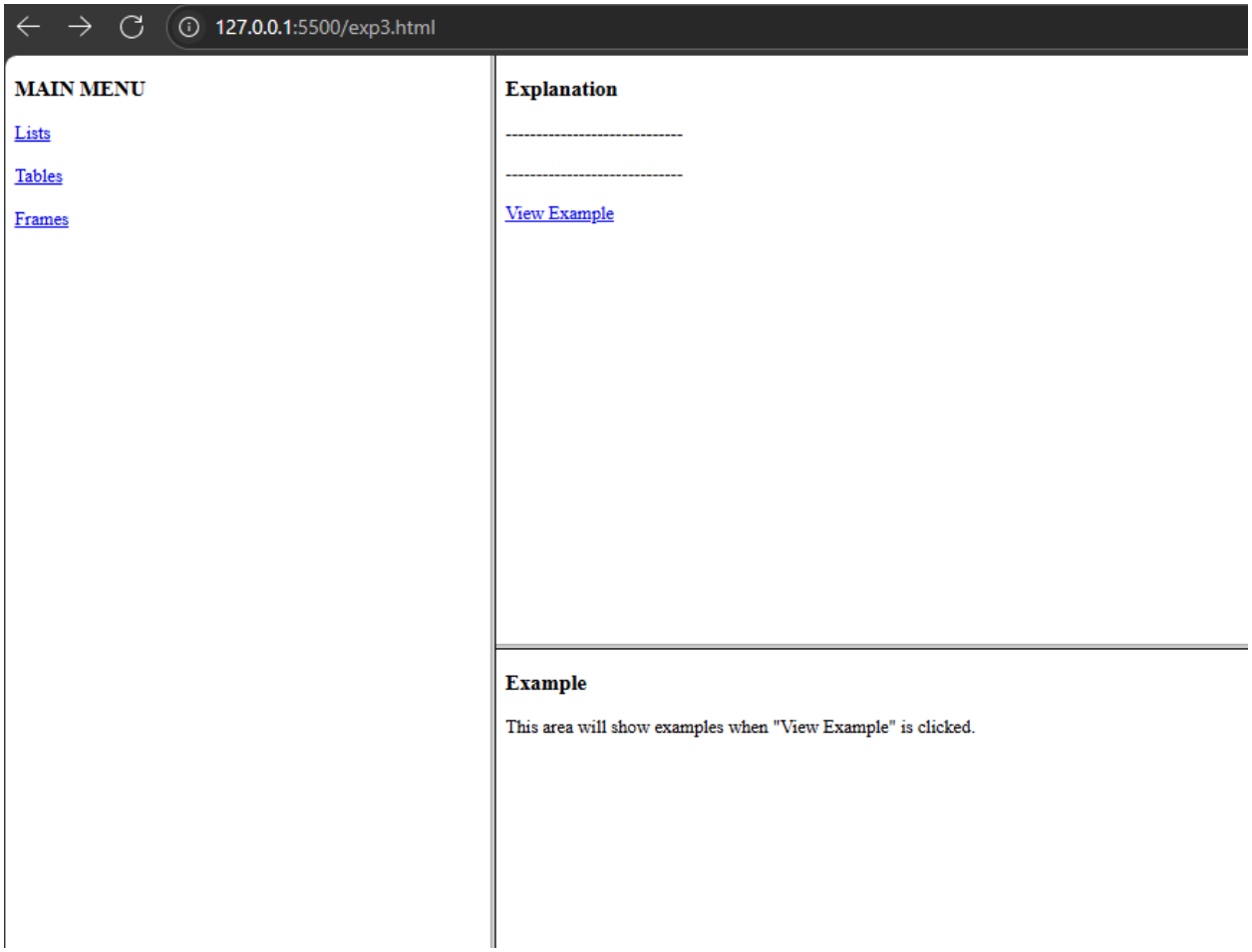
```
<!DOCTYPE html>
<html>
<head>
   <title>Main Menu</title>
</head>
<body>
   <h3>MAIN MENU</h3>
   <a href="lists.html" target="explanation">Lists</a><br><br>
   <a href="tables.html" target="explanation">Tables</a><br><br>
   <a href="frames.html" target="explanation">Frames</a>
</body>
</html>
```

Explanation.html:

```
<!DOCTYPE html>
<html>
<head><title>Explanation</title></head>
<body>
   <h3>Explanation</h3>
   <p>----------</p>
   <p>----------</p>
   <a href="example-detail.html" target="example"><u>View Example</u></a>
</body>
</html>
```

Example.html:

```
<!DOCTYPE html>
<html>
<head><title>Example</title></head>
<body>
   <h3>Example</h3>
   <p>This area will show examples when "View Example" is clicked.</p>
</body>
</html>
```


127.0.0.1:5500/exp3.html
MAIN MENU
Lists
Tables
Frames
Explanation

View Example
Example
This area will show examples when "View Example" is clicked.

EXPERIMENT – 4

Q. Design following HTML form:

Code:

```
<!DOCTYPE html>
<html>
<head>
   <title>Employee Form</title>
</head>
<body>
   <h2><b><u>Employee form</u></b></h2>

   <form>
      <!-- Name -->
      <label>name</label><br>
      <input type="text" name="name"><br><br>

      <!-- Designation -->
      <label>designation</label>
      <select name="designation">
         <option>manager</option>
         <option>developer</option>
         <option>designer</option>
         <option>tester</option>
      </select><br><br>

      <!-- Salary and Password -->
```

```
        <label>Salary</label>
        <input type="text" name="salary">
        <label>pwd</label>
        <input type="password" name="pwd"><br><br>

        <!-- Specialization -->
        <label>Specialization</label>
        <input type="radio" name="specialization" value="IT"> IT
        <input type="radio" name="specialization" value="management"> management
        <br><br>

        <!-- Hobbies -->
        <label>hobbies</label><br>
        <input type="checkbox" name="hobbies" value="travelling"> travelling<br>
        <input type="checkbox" name="hobbies" value="sports"> sports<br>
        <input type="checkbox" name="hobbies" value="reading"> reading<br>
        <input type="checkbox" name="hobbies" value="netsurfing"> net surfing<br><br>

        <!-- Address -->
        <label>address</label><br>
        <textarea name="address" rows="4" cols="40" placeholder="please enter the
address"></textarea><br><br>

        <!-- Buttons -->
        <input type="submit" value="submit">
        <input type="reset" value="reset">
    </form>
</body>
</html>
```

Output:

Employee form

name
Nischal

designation manager

Salary 50,000 pwd ••••

Specialization IT management

hobbies
- travelling
- sports
- reading
- net surfing

address
kapan,kathmandu

submit reset

EXPERIMENT NO. 3

Aim: To implement form Validation technique using PHP.

Theory: It is very essential to have the input to your form validated before taking the form submission data for further processing. When there are many fields in the form, the PHP validation script becomes too complex. Moreover, since you are doing the same or similar validation for most of the forms that you make, just too much of duplicate effort is spent on form validations.

Using the PHP form validation script

1. Include formvalidator.php in your form processing script require_once "formvalidator.php”

Index.html

```
<!DOCTYPE html>
<html>
<head>
   <title>PHP Form Validation</title>
</head>
<body>
   <h2>Registration Form</h2>
   <form action="process.php" method="post">
      Name: <input type="text" name="Name"><br><br>
      Email: <input type="text" name="Email"><br><br>
      Comments: <textarea name="Comments"></textarea><br><br>
      <input type="submit" value="Submit">
   </form>
</body>
</html>
```

Formvalidator.php

```
<?php
class FormValidator {
   private $validations = array();
```

```
private $errors = array();

public function addValidation($field, $type, $message) {
   $this->validations[] = array(
      "field" => $field,
      "type" => $type,
      "message" => $message
   );
}

public function ValidateForm() {
   $is_valid = true;
   foreach ($this->validations as $val) {
      $field = $val["field"];
      $type = $val["type"];
      $message = $val["message"];

      if ($type == "req" && empty($_POST[$field])) {
         $this->errors[$field] = $message;
         $is_valid = false;
      }
      if ($type == "email" && !filter_var($_POST[$field], FILTER_VALIDATE_EMAIL)) {
         $this->errors[$field] = $message;
         $is_valid = false;
      }
   }

   foreach ($this->custom_validators as $custom) {
      if (!$custom->DoValidate($_POST, $this->errors)) {
```

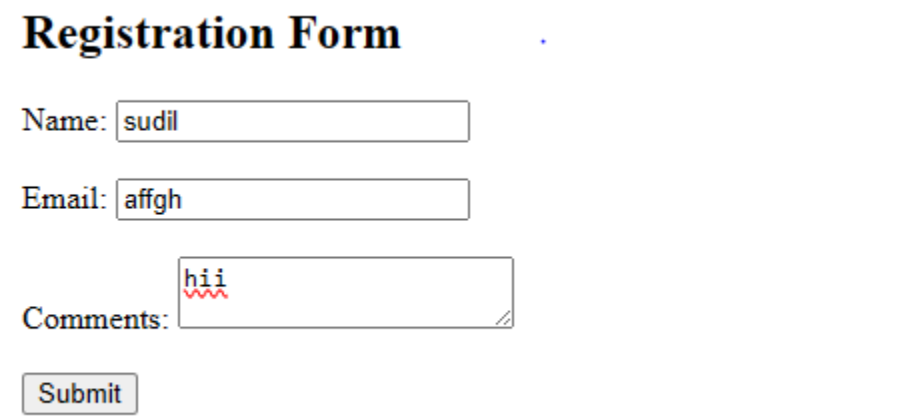
```
            $is_valid = false;
          }
        }

        return $is_valid;
    }

    public function GetErrors() {
        return $this->errors;
    }

    private $custom_validators = array();
    public function AddCustomValidator($obj) {
        $this->custom_validators[] = $obj;
    }
}

abstract class CustomValidator {
    abstract function DoValidate(&$formars, &$error_hash);
}
?>
```

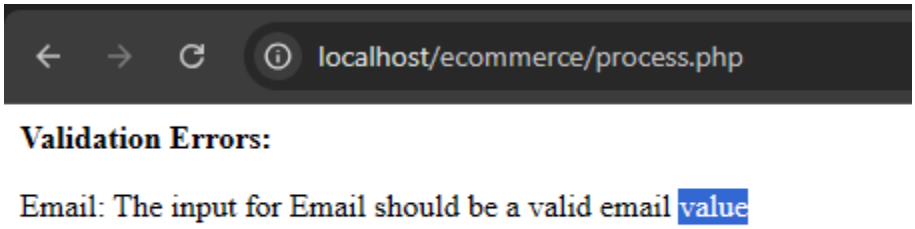
Process.php

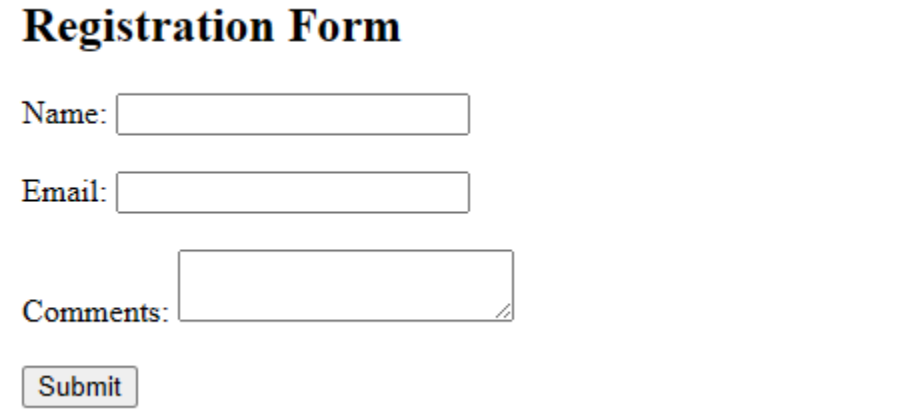
```
<?php
require_once "formvalidator.php";

$validator = new FormValidator();
$validator->addValidation("Name", "req", "Please fill in Name");
$validator->addValidation("Email", "email", "The input for Email should be a valid email value");
```

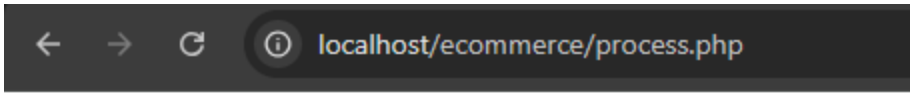
```
$validator->addValidation("Email", "req", "Please fill in Email");

// Optional custom validation: disallow URLs in comments
class MyValidator extends CustomValidator {
   function DoValidate(&$formars, &$error_hash) {
      if (stristr($formars['Comments'], 'http://')) {
         $error_hash['Comments'] = "No URLs allowed in comments";
         return false;
      }
      return true;
   }
}
$custom_validator = new MyValidator();
$validator->AddCustomValidator($custom_validator);

// Run validation
if (!$validator->ValidateForm()) {
   echo "<b>Validation Errors:</b><br>";
   $error_hash = $validator->GetErrors();
   foreach ($error_hash as $field => $error) {
      echo "<p>$field: $error</p>";
   }
} else {
   echo "<h3>Form Submitted Successfully</h3>";
   echo "<p>Name: " . htmlspecialchars($_POST['Name']) . "</p>";
   echo "<p>Email: " . htmlspecialchars($_POST['Email']) . "</p>";
   echo "<p>Comments: " . htmlspecialchars($_POST['Comments']) . "</p>";
}
?>
```

Output:


Registration Form
Name: sudil
Email: affgh
Comments: hii
Submit


localhost/ecommerce/process.php
Validation Errors:
Email: The input for Email should be a valid email value


Registration Form
Name:
Email:
Comments:
Submit



Validation Errors:

Name: Please fill in Name

Email: Please fill in Email

Question 2: Create a FormValidator object and add the form validation descriptors.

$validator = new FormValidator();

$validator->addValidation("Name","req","Please fill in Name");

$validator->addValidation("Email","email", "The input for Email should be a valid email value");
$validator->addValidation("Email","req","Please fill in Email");

The first argument is the name of the input field in the form. The second argument is the validation descriptor that tells the type of the validation required. The third argument is the error message to be displayed if the validation fails.

Process.php(object)

```
<?php

require_once "formvalidator.php";


$validator = new FormValidator();

$validator->addValidation("Name", "req", "Please fill in Name");

$validator->addValidation("Email", "email", "The input for Email should be a valid email value");

$validator->addValidation("Email", "req", "Please fill in Email");


// Optional custom validation for comments

class MyValidator extends CustomValidator {

   function DoValidate(&$formars, &$error_hash) {
```

```
        if (stristr($formars['Comments'], 'http://')) {
            $error_hash['Comments'] = "No URLs allowed in comments";
            return false;
        }
        return true;
    }
}
$custom_validator = new MyValidator();
$validator->AddCustomValidator($custom_validator);

// Run validation
if (!$validator->ValidateForm()) {
    echo "<b>Validation Errors:</b><br>";
    $errors = $validator->GetErrors();
    foreach ($errors as $field => $message) {
        echo "<p>$field: $message</p>";
    }
} else {
    echo "<h2>Form Submitted Successfully!</h2>";
    echo "<p><strong>Name:</strong> " . htmlspecialchars($_POST['Name']) . "</p>";
    echo "<p><strong>Email:</strong> " . htmlspecialchars($_POST['Email']) . "</p>";
    echo "<p><strong>Comments:</strong> " . htmlspecialchars($_POST['Comments']) . "</p>";
}
?>
```

Output:

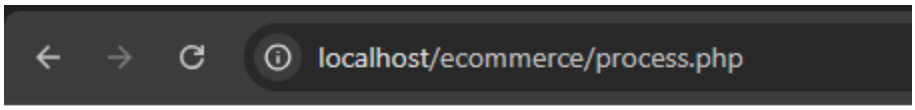
Registration Form

Name: sudil

Email: affgh

Comments: hii

Submit



Validation Errors:

Email: The input for Email should be a valid email value

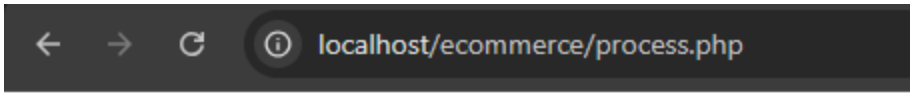
Registration Form

Name:

Email:

Comments:

Submit



Validation Errors:

Name: Please fill in Name

Email: Please fill in Email

Question 3: Validate the form by calling ValidateForm() function

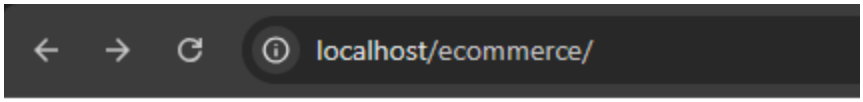
- **Add Custom Validation:**
  1. Create a class for the custom validation and override the DoValidate() function
  2. Add the custom validation object

Process.php

```
<?php
require_once "formvalidator.php";

// Step 1: Create validator object
$validator = new FormValidator();

// Step 2: Add built-in validations
$validator->addValidation("Name", "req", "Please fill in Name");
$validator->addValidation("Email", "email", "The input for Email should be a valid email value");
$validator->addValidation("Email", "req", "Please fill in Email");

// Step 3: Create custom validation class
class MyValidator extends CustomValidator {
   function DoValidate(&$formars, &$error_hash) {
      if (stristr($formars['Comments'], 'http://')) {
         $error_hash['Comments'] = "No URLs allowed in comments";
```

```
            return false;
        }
        return true;
    }
}

// Step 4: Add custom validator
$custom_validator = new MyValidator();
$validator->AddCustomValidator($custom_validator);

// Step 5: Validate the form
if (!$validator->ValidateForm()) {
    echo "<b>Validation Errors:</b>";
    $error_hash = $validator->GetErrors();
    foreach ($error_hash as $inpname => $inp_err) {
        echo "<p>$inpname : $inp_err</p>\n";
    }
} else {
    echo "<h2>Form Submitted Successfully!</h2>";
    echo "<p><strong>Name:</strong> " . htmlspecialchars($_POST['Name']) . "</p>";
    echo "<p><strong>Email:</strong> " . htmlspecialchars($_POST['Email']) . "</p>";
    echo "<p><strong>Comments:</strong> " . htmlspecialchars($_POST['Comments']) . "</p>";
}
?>
```

Output:



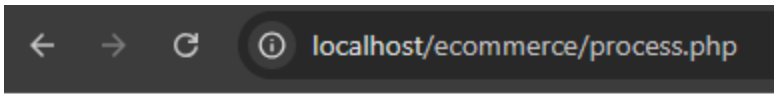
Registration Form

Name: Sp

Email: sud

Comments: http://

Submit



Validation Errors:

Email : The input for Email should be a valid email value

Comments : No URLs allowed in comments

Experiment No. 4
Product Catalog Design

Aim: To implement Catalog design

Theory:

Nothing works better than a product catalog when you are wanting to expand your customer base. A catalog has the potential to reach those customers who will never set foot in your store. The only drawback is that designing a product catalog can be overwhelming, but a checklist of tasks to complete will make the design process much easier. Here are 5 steps that will help you transform your catalog from an idea to a tangible advertising tool.

Code

Catlog.html

```
<!DOCTYPE html>
<html>
<head>
  <title>Mobile Phones Catalog</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background-color: white;
      margin: 0;
      padding: 0;
    }
    .container {
      display: flex;
      padding: 20px;
    }
    .sidebar {
      width: 20%;
      padding: 10px;
```

```
    border-right: 1px solid #ccc;
}
.sidebar h3 {
    color: #c60;
}
.sidebar ul {
    list-style-type: none;
    padding-left: 0;
}
.sidebar ul li {
    margin: 3px 0;
    font-size: 14px;
}

.catalog {
    width: 60%;
    padding: 10px 30px;
}

.catalog h2 {
    color: #c60;
    border-bottom: 1px solid #ccc;
    padding-bottom: 5px;
}

.brand-section {
    margin-top: 15px;
}
```

```
.brand-title {
    color: #c60;
    font-size: 16px;
    margin-bottom: 5px;
}

.phone-entry {
    display: flex;
    margin-bottom: 15px;
}

.phone-entry img {
    height: 100px;
    margin-right: 10px;
}

.phone-info {
    font-size: 14px;
}

.right-panel {
    width: 20%;
    padding: 10px;
    border-left: 1px solid #ccc;
}

.right-panel h4 {
    color: #c60;
    font-size: 16px;
```

```
            border-bottom: 1px solid #ccc;
            margin-bottom: 10px;
        }

        .right-panel ul {
            list-style: none;
            padding-left: 0;
        }

        .right-panel ul li {
            margin-bottom: 10px;
        }

        .right-panel ul li a {
            text-decoration: none;
            color: #000;
            font-size: 13px;
        }

        .right-panel ul li a:hover {
            text-decoration: underline;
        }
    </style>
</head>
<body>

<div class="container">
    <!-- Sidebar -->
    <div class="sidebar">
```

```
    <h3>Mobile phones</h3>
    <ul>
      <li>Acer (51)</li>
      <li>Aeg (1)</li>
      <li>Airfins (4)</li>
      <li>Alcatel (213)</li>
      <li>Apple (72)</li>
      <li>Asus (60)</li>
      <li>BenQ (73)</li>
      <li>BBK (22)</li>
    </ul>
  </div>

  <!-- Catalog -->
  <div class="catalog">
    <h2>MOBILE PHONES CATALOG</h2>

    <!-- Acer -->
    <div class="brand-section">
      <div class="brand-title">Acer</div>

      <div class="phone-entry">
        <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTDB5Q4DpA7zHjSRS9FqXMClG8LTkDfiTyNCg&s" alt="Acer Iconia">
        <div class="phone-info">
          <strong>Acer Iconia Smart</strong><br>
          Year: 2011<br>
          Network: GSM 1800/GSM 1900/GSM 850/GSM 900/HSDPA<br>
```

```
            Dimensions: 141.7x64.5x13.6 mm<br>
            Weight: 185 g
        </div>
    </div>

    <div class="phone-entry">
        <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcT5HUbxskdVPcxfMTTlGEB-aui3yvZ660GLxA&s" alt="Acer Liquidmini">
        <div class="phone-info">
            <strong>Acer Liquidmini E310</strong><br>
            Year: 2011<br>
            Network: GSM 1800/GSM 1900/GSM 850/GSM 900/HSDPA<br>
            Dimensions: 110.4x57.5x13 mm<br>
            Weight: 109 g
        </div>
    </div>

    <div class="phone-entry">
        <img
src="https://m-cdn.phonearena.com/images/reviews/86880-image/Acer-beTouch-E210-Review-Comparison.webp" alt="Acer beTouch E210">
        <div class="phone-info">
            <strong>Acer beTouch E210</strong><br>
            Year: 2011<br>
            Network: GSM 850/900/1800/1900/HSDPA<br>
            Dimensions: 116x63x11.5 mm<br>
            Weight: 109 g
        </div>
    </div>
```

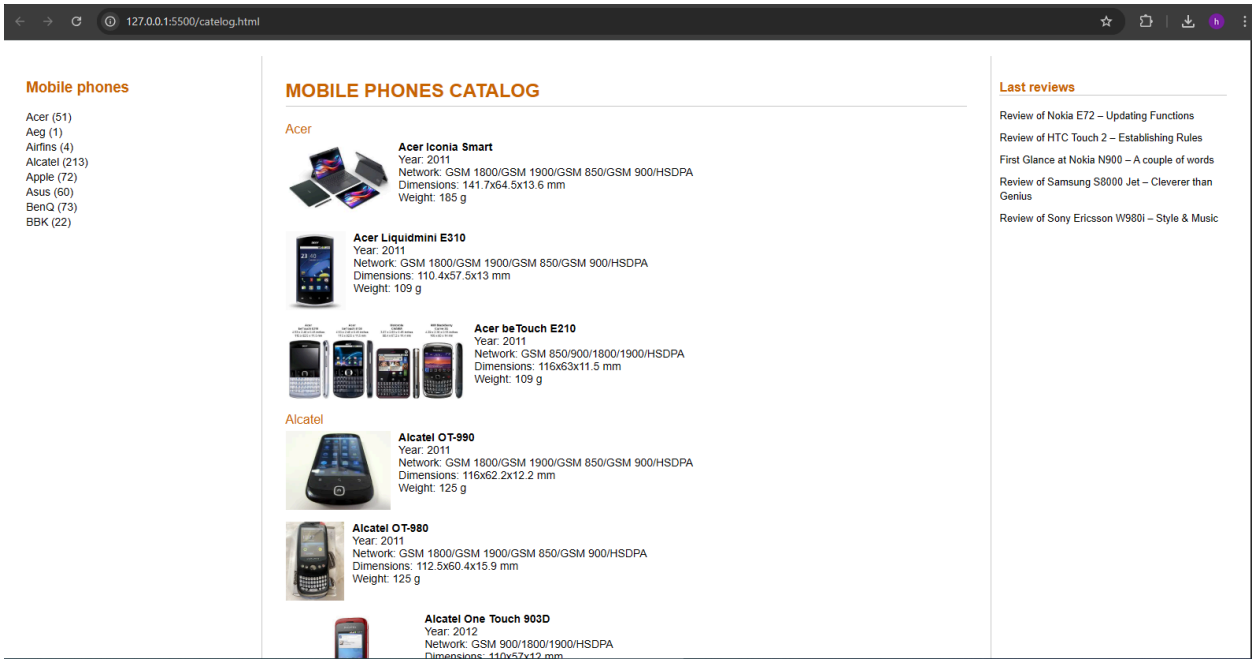
```
    </div>


    <!-- Alcatel -->
    <div class="brand-section">
      <div class="brand-title">Alcatel</div>


      <div class="phone-entry">
        <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcTuSxxhqWvPBjppclci5cum0T23rR1RpwwJnA&s" alt="Alcatel OT-990">
        <div class="phone-info">
          <strong>Alcatel OT-990</strong><br>
          Year: 2011<br>
          Network: GSM 1800/GSM 1900/GSM 850/GSM 900/HSDPA<br>
          Dimensions: 116x62.2x12.2 mm<br>
          Weight: 125 g
        </div>
      </div>


      <div class="phone-entry">
        <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQucRWIF7rFYehERvNT-L16IyerpfwNQ4mHYg&s" alt="Alcatel OT-980">
        <div class="phone-info">
          <strong>Alcatel OT-980</strong><br>
          Year: 2011<br>
          Network: GSM 1800/GSM 1900/GSM 850/GSM 900/HSDPA<br>
          Dimensions: 112.5x60.4x15.9 mm<br>
          Weight: 125 g
        </div>
```

```
        </div>


        <div class="phone-entry">
            <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQUdGMb4V-3ZiSsyRGroE_TN9aV2x5GRSuV9A&s" alt="Alcatel One Touch 903D">
            <div class="phone-info">
                <strong>Alcatel One Touch 903D</strong><br>
                Year: 2012<br>
                Network: GSM 900/1800/1900/HSDPA<br>
                Dimensions: 110x57x12 mm<br>
                Weight: 118 g
            </div>
        </div>
    </div>


    <!-- Apple -->
    <div class="brand-section">
        <div class="brand-title">Apple</div>


        <div class="phone-entry">
            <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQk9ZWOa_e0ILYXTfQcuRT7XL7Y3d7YqgEZNw&s" alt="iPhone 4">
            <div class="phone-info">
                <strong>Apple iPhone 4 CDMA</strong><br>
                Year: 2011<br>
                Network: CDMA 1900/CDMA 2000 1X/CDMA 800/EV-DO<br>
                Dimensions: 115.2x58.6x9.3 mm<br>
                Weight: 137 g
```

```
        </div>
      </div>

      <div class="phone-entry">
        <img
src="https://encrypted-tbn0.gstatic.com/images?q=tbn:ANd9GcQkLscpmW1i6GAfif7OLXfav58KcFlvkRVMbA&s" alt="iPhone 4S">
        <div class="phone-info">
          <strong>Apple iPhone 4S</strong><br>
          Year: 2011<br>
          Network: GSM/CDMA/HSDPA/EVDO<br>
          Dimensions: 115.2x58.6x9.3 mm<br>
          Weight: 140 g
        </div>
      </div>

      <div class="phone-entry">
        <img
src="https://eoe.works/cdn/shop/files/image_cce7613a-5439-4ea9-87ab-844e6de03206_1200x1200.png?v=1685608426" alt="iPhone 5">
        <div class="phone-info">
          <strong>Apple iPhone 5</strong><br>
          Year: 2012<br>
          Network: GSM/CDMA/HSDPA/LTE<br>
          Dimensions: 123.8x58.6x7.6 mm<br>
          Weight: 112 g
        </div>
      </div>
    </div>
  </div>
```

```
    <!-- Right Panel -->
    <div class="right-panel">
      <h4>Last reviews</h4>
      <ul>
        <li><a href="#">Review of Nokia E72 – Updating Functions</a></li>
        <li><a href="#">Review of HTC Touch 2 – Establishing Rules</a></li>
        <li><a href="#">First Glance at Nokia N900 – A couple of words</a></li>
        <li><a href="#">Review of Samsung S8000 Jet – Cleverer than Genius</a></li>
        <li><a href="#">Review of Sony Ericsson W980i – Style & Music</a></li>
      </ul>
    </div>
</div>

</body>

</html>
```

Output:



LAB-5

Aim: Implementation of Access Control Mechanism

Index.html:

```
<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<meta name="viewport" content="width=device-width,
initial-scale=1" />
<title>Login</title>
<style>
  body {
    font-family: Arial, sans-serif;
    background: #f5f5f5;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
  }
  .login-container {
    background: white;
    padding: 20px;
    border-radius: 6px;
    box-shadow: 0 2px 6px rgba(0,0,0,0.1);
    width: 300px;
  }
  h2 {
    text-align: center;
    margin-bottom: 15px;
  }
  input[type="text"],
  input[type="password"],
  input[type="submit"] {
```

```
    width: 100%;
    padding: 10px;
    margin-bottom: 12px;
    border: 1px solid #ccc;
    border-radius: 4px;
  }
  input[type="submit"] {
    background: #4a90e2;
    color: white;
    border: none;
    cursor: pointer;
  }
  input[type="submit"]:hover {
    background: #357ABD;
  }
</style>
</head>
<body>

<div class="login-container">
  <h2>Login</h2>
  <form action="loginhandler.php" method="POST"
autocomplete="off">
    <input type="text" name="username" placeholder="Username"
required />
    <input type="password" name="password"
placeholder="Password" required />
    <input type="submit" value="Login" />
  </form>
</div>

</body>
</html>
```

Accesscontrol.php:

```
<?php
session_start();

if (!isset($_SESSION['username'])) {
    header("Location: index.html");
    exit();
}

if ($_SESSION['role'] !== 'admin') {
    echo "<p style='color:red; text-align:center;'>Access
denied. You do not have permission.<br><a
href='logout.php'>Logout</a></p>";
    exit();
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
<meta charset="UTF-8" />
<title>Admin Panel</title>
<style>
  body {
    font-family: Arial, sans-serif;
    background: #eef2f7;
    display: flex;
    justify-content: center;
    align-items: center;
    height: 100vh;
    margin: 0;
  }
  .container {
    background: white;
    padding: 20px;
```

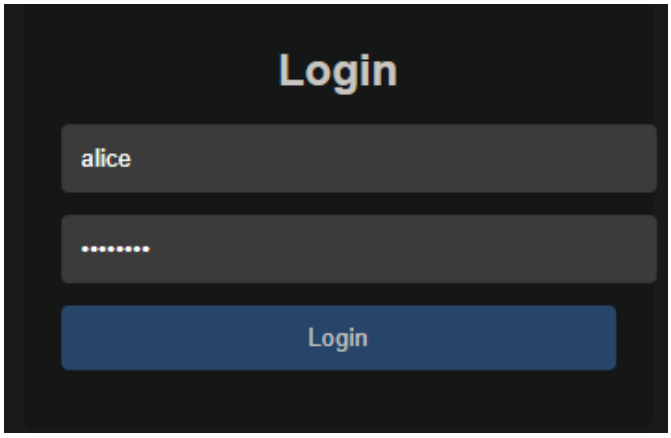
```
    border-radius: 6px;
    text-align: center;
    box-shadow: 0 2px 8px rgba(0,0,0,0.1);
    width: 300px;
  }
  a {
    display: inline-block;
    margin-top: 15px;
    padding: 10px 15px;
    background: #4a90e2;
    color: white;
    text-decoration: none;
    border-radius: 4px;
  }
  a:hover {
    background: #357ABD;
  }
</style>
</head>
<body>

<div class="container">
  <h1>Admin Panel</h1>
  <p>Welcome, <?php echo
htmlspecialchars($_SESSION['username']); ?>!</p>
  <a href="logout.php">Logout</a>
</div>

</body>
</html>
```

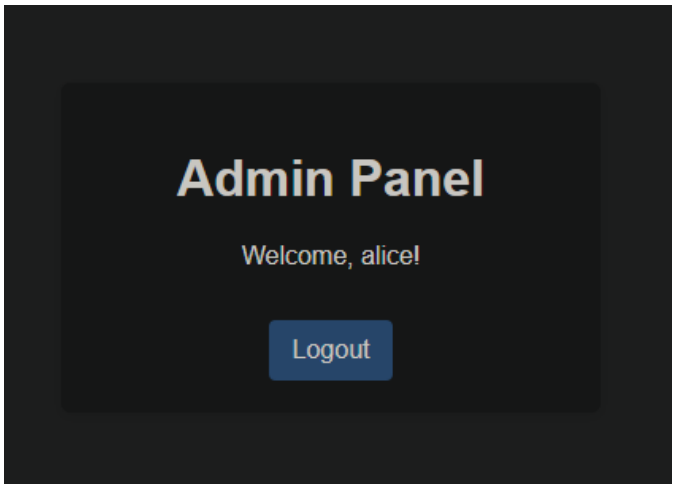
Loginhandler.php:

```
<?php
session_start();

$users = [
    'alice' => ['password' => password_hash('alice123',
PASSWORD_DEFAULT), 'role' => 'admin'],
    'bob' => ['password' => password_hash('bob123',
PASSWORD_DEFAULT), 'role' => 'user'],
];

$username = $_POST['username'] ?? '';
$password = $_POST['password'] ?? '';

function showMessage($message) {
    echo "<div style='
        max-width: 400px; margin: 100px auto;
        padding: 20px; border: 1px solid #ddd;
        border-radius: 8px; font-family: Arial, sans-serif;
        color: #333; text-align: center;
        box-shadow: 0 4px 10px rgba(0,0,0,0.1);
        '>$message<br><br>
        <a href='index.html' style='
          text-decoration: none;
          color: #4a90e2;
          font-weight: bold;'>Go back to Login</a>
    </div>";
}

if (isset($users[$username])) {
    if (password_verify($password,
$users[$username]['password'])) {
        $_SESSION['username'] = $username;
        $_SESSION['role'] = $users[$username]['role'];
```

```
        header("Location: accesscontrol.php");
        exit();
    } else {
        showMessage(" Invalid password.");
    }
} else {
    showMessage(" User not found.");
}
?>
```

Output:





Lab No. 6

Aim: To implement Session management.

Theory:

PHP Session Management:

PHP Session Management is a mechanism used to store information (variables) to be used across multiple pages of a website. Unlike cookies, which store data on the client-side, sessions store data on the server-side. This makes them more secure and ideal for storing sensitive data like user login information.

Overview:

An overview of PHP session management is shown in Figure 6-1. When a user first enters the session-based application by making a request to a page that starts a session, PHP generates a session ID and creates a file that stores the session-related variables. PHP sets a cookie to hold the session ID in the response the script generates. The browser then records the cookie and includes it in subsequent requests. In the example shown in Figure 8-1, the script welcome.php records session variables in the session store, and a request to next.php then has access to those variables because of the session ID.

The out-of-the-box configuration of PHP session management uses disk-based files to store session variables. Using files as the session store is adequate for most applications in which the numbers of concurrent sessions are limited.

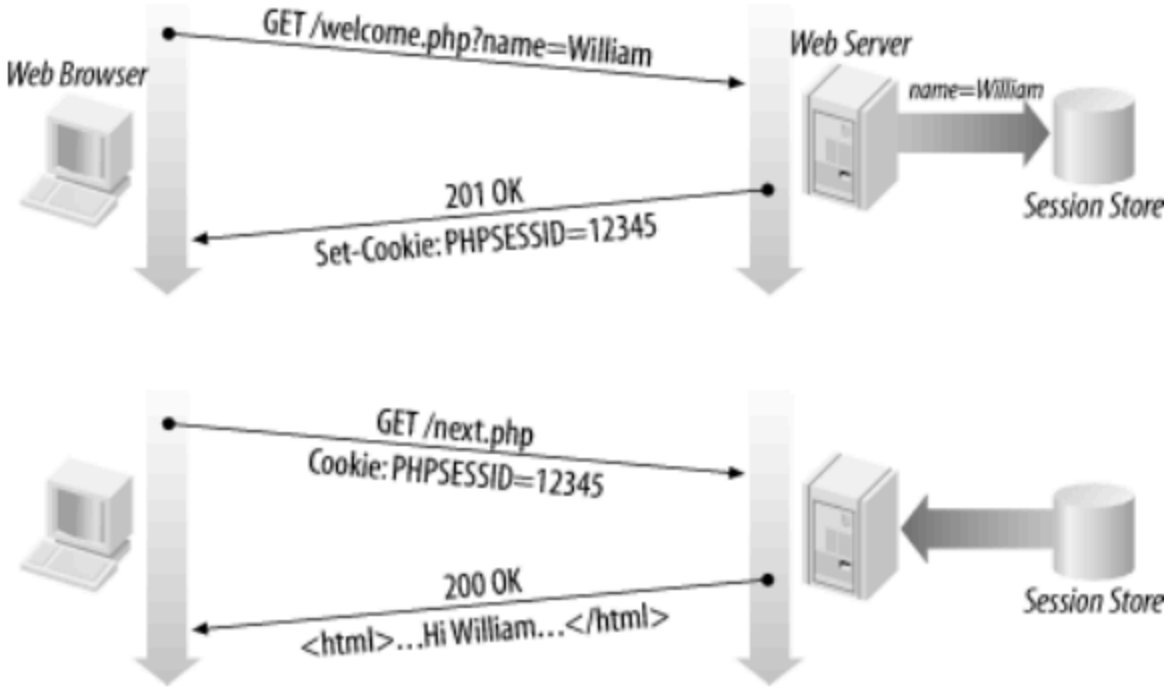


Fig-. The interaction between the browser and the server when initial requests are made to a session-based application

Starting a Session :

PHP provides a session_start( ) function that creates a new session and subsequently identifies and establishes an existing one. Either way, a call to the session_start( ) function initializes a session.

The first time a PHP script calls session_start( ), a session identifier is generated, and, by default, a Set-Cookie header field is included in the response. The response sets up a session cookie in the browser with the name PHPSESSID and the value of the session identifier. The PHP session management automatically includes the cookie without the need to call to the setcookie( ) or header( ) functions.

Using Session Variables in PHP:

Session variables allow data to be stored on the server and accessed across multiple pages during a user's visit.

- To start a session, use the session_start() function at the beginning of the script.
- Session variables are stored in the $_SESSION superglobal array.
- Example:

  session_start();

  $_SESSION["username"] = "JohnDoe";

Ending a Session

When a user logs out, the session should be ended to clear stored data and prevent unauthorized access.

- session_unset() removes all session variables.
- session_destroy() deletes the session file on the server.
- Example:

  session_start();

  session_unset();

  session_destroy();

After this, the user will need to start a new session to log in again.

Code:

Login.php

```
<?php
session_start();

if ($_SERVER["REQUEST_METHOD"] == "POST") {
   $username = trim($_POST["username"]);
   $password = trim($_POST["password"]);

   // Simple static login validation (you can replace with DB check)
```

```
    if ($username === "JohnDoe" && $password === "1234") {
        $_SESSION["username"] = $username;
        header("Location: welcome.php");
        exit();
    } else {
        $error = "Invalid username or password!";
    }
}
?>
<!DOCTYPE html>
<html>
<head>
    <title>Login</title>
    <style>
        body {
            font-family: Arial, sans-serif;
            background: linear-gradient(to right, #4facfe, #00f2fe);
            display: flex;
            justify-content: center;
            align-items: center;
            height: 100vh;
        }
        .login-container {
            background: white;
            padding: 30px;
            border-radius: 15px;
            box-shadow: 0px 4px 10px rgba(0,0,0,0.2);
```

```
    width: 320px;
}
h2 {
    text-align: center;
    margin-bottom: 20px;
    color: #333;
}
input {
    width: 100%;
    padding: 10px;
    margin: 8px 0;
    border-radius: 8px;
    border: 1px solid #ccc;
}
button {
    width: 100%;
    padding: 10px;
    background: #4facfe;
    border: none;
    border-radius: 8px;
    color: white;
    font-size: 16px;
    cursor: pointer;
}
button:hover {
    background: #00c6ff;
}
```

```
        .error {
            color: red;
            text-align: center;
        }
    </style>
</head>
<body>
    <div class="login-container">
        <h2>Login</h2>
        <?php if (!empty($error)) echo "<p class='error'>$error</p>"; ?>
        <form method="post" action="">
            <input type="text" name="username" placeholder="Enter Username" required>
            <input type="password" name="password" placeholder="Enter Password" required>
            <button type="submit">Login</button>
        </form>
    </div>
</body>
</html>
```

Welcome.php

```
<?php
session_start();
if (!isset($_SESSION["username"])) {
    header("Location: login.php");
    exit();
}
```

```
?>
<!DOCTYPE html>
<html>
<head>
  <title>Welcome</title>
  <style>
    body {
      font-family: Arial, sans-serif;
      background: linear-gradient(to right, #43cea2, #185a9d);
      display: flex;
      justify-content: center;
      align-items: center;
      height: 100vh;
    }
    .welcome-box {
      background: white;
      padding: 40px;
      border-radius: 15px;
      text-align: center;
      box-shadow: 0px 4px 10px rgba(0,0,0,0.2);
      width: 350px;
    }
    h2 {
      margin-bottom: 20px;
      color: #333;
    }
    .logout-btn {
      padding: 10px 20px;
      background: #ff4b5c;
      border: none;
```

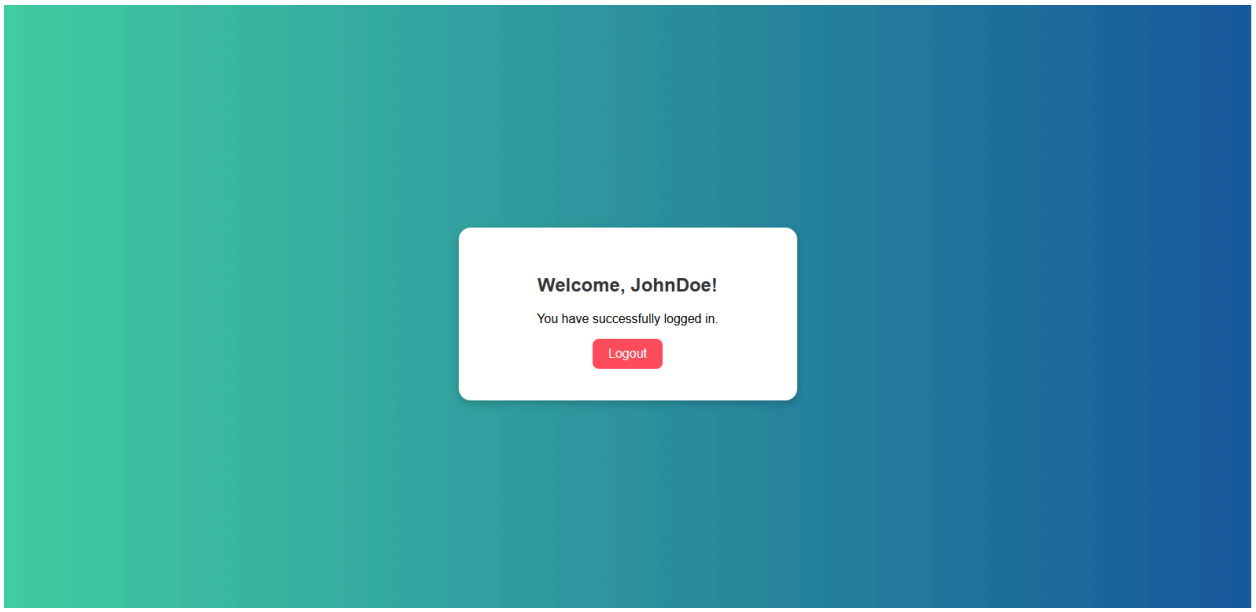
```
            border-radius: 8px;
            color: white;
            font-size: 16px;
            cursor: pointer;
        }
        .logout-btn:hover {
            background: #ff1e3c;
        }
    </style>
</head>
<body>
    <div class="welcome-box">
        <h2>Welcome, <?php echo htmlspecialchars($_SESSION["username"]); ?>!</h2>
        <p>You have successfully logged in.</p>
        <form action="logout.php" method="post">
            <button type="submit" class="logout-btn">Logout</button>
        </form>
    </div>
</body>
</html>
```

Logout.php

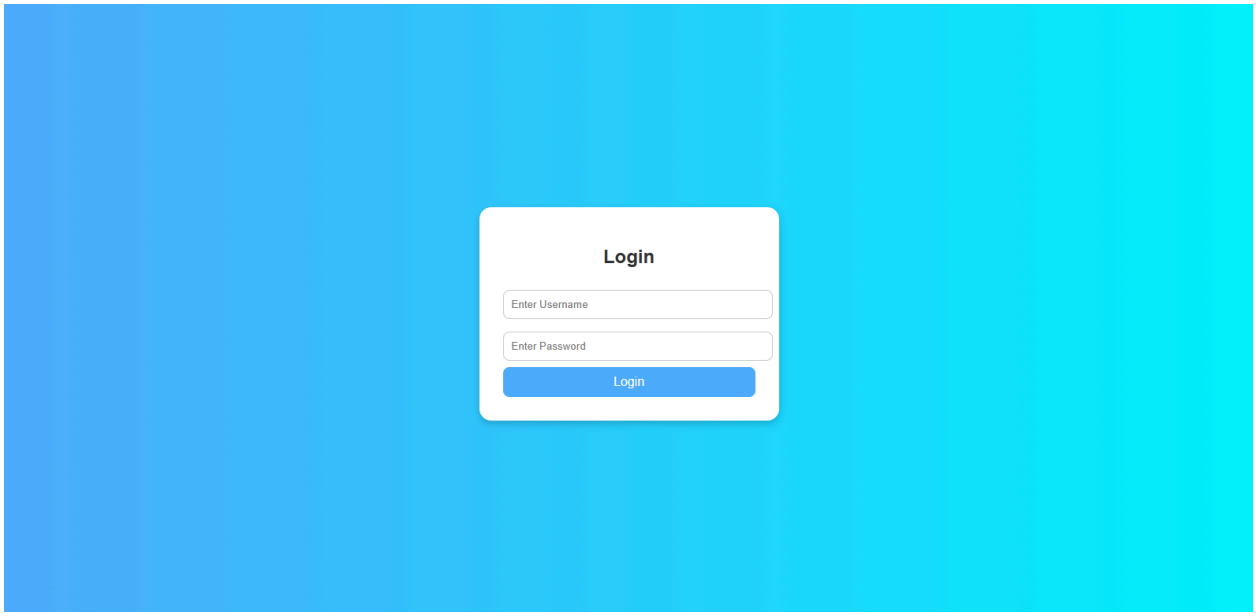
```
<?php
session_start();
session_unset();
session_destroy();
header("Location: login.php");
exit();
?>
```

Output:

After login:



After logout:


Login
Enter Username
Enter Password
Login