Set B

Develop Online Shopping System that allows customer to register themselves in a system. System has three users such as: 1) Customer 2) Admin. A customer can buy a product online by adding it into the cart, providing a address to deliver and proceed to checkout. Admin can add the product which are shown to the customers. A customer can not buy any product without registering himself in the system.

Applicable after Practical No. 5.
Consider the following relations to store the data.

Product:
ProductID (Primary Key)
Name
Description
Price
StockQuantity

Category:
CategoryID (Primary Key)
CategoryName

Customer:
CustomerID (Primary Key)
FirstName
LastName
Email
Password
Address

Order:
OrderID (Primary Key)
CustomerID (Foreign Key referencing Customer)
OrderDate
TotalAmount

OrderItem:
OrderItemID (Primary Key)
OrderID (Foreign Key referencing Order)
ProductID (Foreign Key referencing Product)
Quantity
Subtotal

Payment:
PaymentID (Primary Key)
OrderID (Foreign Key referencing Order)
Amount
PaymentDate

<table>
<tr><td>Practical No.: 06</td><td colspan="3">Enrolment No.:</td></tr>
<tr><td>Objective:</td><td colspan="3">To acquaintance the concept of File management in php.</td></tr>
<tr><td>Practical Exercise:</td><td colspan="3">How are the customer details are organized and stored in the database using PHP? Specifically, consider the information provided in the customer.csv file, which includes data such as the Customer ID, Name, Products customer ordered, Total amount per customer, and location. Discuss the process of reading this CSV file using PHP and storing the extracted information in a database, ensuring that the details for each customer, including the contact, mail ID etc. are accurately captured and maintained in the database records.

Consider the following customer.csv file for input the class details of student.

Customer Data
customer_id,first_name,last_name,email,phone_number,address,city,state,zip_code
1,John,Doe,johndoe@example.com,555-123-4567,123 Main St,Anytown,CA,12345
2,Jane,Smith,janesmith@example.com,555-987-6543,456 Elm St,Smalltown,NY,54321
3,Alice,Johnson,alicejohnson@example.com,555-456-7890,789 Oak St,Bigcity,TX,67890

Product Data
product_id,product_name,description,price
101,Laptop,"15"" Laptop with 8GB RAM and 512GB SSD",899.99
102,Smartphone,"6.5"" Smartphone with 128GB storage",699.99
103,Headphones,"Noise-cancelling headphones with Bluetooth connectivity",149.99
104,Tablet,"10"" Tablet with 64GB storage",349.99</td></tr>
<tr><td>Time required to implement and debug: 03 Hours</td><td colspan="2">CO Mapped: CO1 and CO2</td><td>PO Mapped: PO1, PO2 and PO4</td></tr>
<tr><td>References to solve the problem:</td><td colspan="3">Steven Holzner, The Complete Reference, Tata McGraw Hill, Page No. 55-74 and 92-104</td></tr>
<tr><td></td><td colspan="2">Solution</td><td>Viva</td></tr>
<tr><td>Out of Marks:</td><td colspan="2">10</td><td>05</td></tr>
<tr><td>Secured by Student:</td><td colspan="2"></td><td></td></tr>
<tr><td>Teacher's Signature:</td><td colspan="2"></td><td></td></tr>
<tr><td>Date:</td><td colspan="2"></td><td></td></tr>
</table>

Practical No.: 07	**Enrolment No.:**	
Objective:	To understand the concept of validation and database connection.	
Practical Exercise:	Create a Registration form for visitors to become a Customer. Take appropriate control to design the registration form and apply client-side and server-side validation on fields based on its nature. Registration form of customers must have CustomerID (Primary Key), Name, email, password, dateOfBirth, gender, address, mobileNo, and two buttons signup and reset. The following must be applied. 1. Name should not contain any number or special symbol. 2. Password length must be al-least 8 character, and it must have at-least one number and one special symbol. 3. Email should be valid email address. 4. Mobile number contains only number. 5. Password and retype password must be matched. 6. Apply appropriate validations for dateOfBirth field. 8. Whenever user click on reset button data should be clear from all fields. After successful registration data should be displayed in another page in table format.	
Time required to implement and debug: 03 Hours	**CO Mapped:** CO1 and CO2	**PO Mapped:** PO1, PO2 and PO4
References to solve the problem:	Steven Holzner, The Complete Reference, Tata McGraw Hill, Page No. 55-74 and 92-104	
	Solution	**Viva**
Out of Marks:	**10**	**05**
Secured by Student:		
Teacher's Signature:		
Date:		

Practical No.: 08	**Enrolment No.:**
Objective:	To understand the concept of cookies, session and authentication.
Practical Exercise:	Consider the Customer, Product, and Order tables given above. **Customer Registration:** • Design a secure registration form that captures essential details such as first name, last name, email, password, address, city, state, and zip code. • Ensure that the customer_id is generated automatically as a primary key. • Implement server-side validation to check for the uniqueness of the email address and enforce password strength requirements. • Validate the format of email addresses and zip codes, and sanitize input data to prevent SQL injection attacks. • Upon successful registration, store the customer's information in the database and use sessions to maintain login status. **Product Management:** • Create forms to add, update, and delete products. • Validate input data to prevent SQL injection and ensure data integrity. • Implement server-side validation to ensure that product names are unique and prices are numeric. • Store product information in the database tables. **Order Processing:** • Design an order form that allows customers to select products and specify quantities. • Validate input data to prevent manipulation and ensure that quantities are numeric and positive. • Upon submission, create a new order record in the database, linking it to the corresponding customer and product records. **Login Mechanism:** • Implement a secure login form that captures the customer's email and password. • Validate the credentials against the database to authenticate the user. • Set a session upon successful authentication to maintain the login status. • Additionally, utilize cookies for a persistent login feature, allowing customers to stay logged in across sessions.
Time required to implement and debug: 03 Hours	**CO Mapped:** CO1 and CO2 / **PO Mapped:** PO1, PO2 and PO4
References to solve the problem:	Steven Holzner, The Complete Reference, Tata McGraw Hill, Page No. 55-74 and 92-104
	Solution / **Viva**
Out of Marks:	**10** / **05**
Secured by Student:	
Teacher's Signature:	
Date:	

Dr. Ronak B. Patel

<table>
<tr><td>Practical No.: 09</td><td colspan="3">Enrolment No.:</td></tr>
<tr><td>Objective:</td><td colspan="3">To understand the concept of foreign key and Ajax.</td></tr>
<tr><td>Practical Exercise:</td><td colspan="3">Consider Product, Category, and Order tables:

Here's how you can design an AJAX-based system for managing products and orders:

Product Management:
• Implement CRUD operations for products (Create, Read, Update, Delete) using AJAX.
• Develop an intuitive user interface that allows administrators to add, edit, and delete products seamlessly.
• Utilize AJAX calls to update and retrieve product data from the server without reloading the entire page, ensuring a dynamic and responsive user experience.
• Implement error handling mechanisms to inform administrators about the success or failure of their CRUD operations.

Order Processing:
• Design an order form that allows customers to select products and specify quantities.
• Utilize AJAX to dynamically update the order summary and total price based on the selected products and quantities without refreshing the page.
• Implement server-side validation to prevent manipulation and ensure data integrity.
• Upon submission, create a new order record in the database, linking it to the corresponding customer and product records.
• Use AJAX to handle the submission of the order form and display success or error messages to the user without reloading the page.

User Authentication and Authorization:
• Implement a secure login system using AJAX to authenticate users and manage sessions.
• Utilize AJAX calls to check the user's credentials against the database and set session variables upon successful login.
• Implement access control mechanisms to restrict certain functionalities (such as CRUD operations on products) to authenticated administrators only.</td></tr>
<tr><td>Time required to implement and debug: 03 Hours</td><td colspan="2">CO Mapped: CO1 and CO2</td><td>PO Mapped: PO1, PO2 and PO4</td></tr>
<tr><td>References to solve the problem:</td><td colspan="3">Steven Holzner, The Complete Reference, Tata McGraw Hill, Page No. 55-74 and 92-104</td></tr>
<tr><td></td><td colspan="2">Solution</td><td>Viva</td></tr>
<tr><td>Out of Marks:</td><td colspan="2">10</td><td>05</td></tr>
<tr><td>Secured by Student:</td><td colspan="2"></td><td></td></tr>
<tr><td>Teacher's Signature:</td><td colspan="2"></td><td></td></tr>
<tr><td>Date:</td><td colspan="2"></td><td></td></tr>
</table>

Dr. Ronak B. Patel