Core Practical 4 For the students admitted from A.Y. 2023-2024 & onwards		
Offering Department: **Computer Application**	Offered to: **Master of Computer Application**	
Semester - I		
Course Code	**Course Title**	**Course Credit and Hours**
23MCACC108	**Core Practical 2 :** Web development using PHP	**2 Credits - 4 hrs/wk**

Course Description:
This course provides an in-depth introduction to PHP programming language with a focus on object-oriented programming (OOP) principles. Students will learn the basics of PHP syntax, control structures, arrays, functions, and handling of HTML forms. The course then delves into more advanced PHP concepts such as object-oriented programming, inheritance, and interfaces. The course also covers the basics of JavaScript including event handling and manipulation of the Document Object Model (DOM). In addition, the course covers the basics of MySQL database management, including SQL language, creating databases and tables, and working with data in tables. By the end of the course, students will have a solid understanding of how to use PHP and related technologies such as JavaScript and MySQL to build dynamic web applications.

Course Purpose:

The purpose of this PHP course is to equip students with the knowledge and skills necessary to develop web applications using PHP, object-oriented programming (OOP) principles, JavaScript, and MySQL. Through a combination of lectures, practical exercises, and project-based assignments, students will learn how to design and develop dynamic, data-driven web applications that can interact with users in real-time. By the end of the course, students will be able to use PHP to build complex web applications that incorporate user authentication, form handling, data storage and retrieval, and dynamic page generation. The course also aims to provide students with a solid foundation in OOP concepts, enabling them to build scalable, modular, and maintainable codebases. Overall, the course is designed to prepare students for a career in web development, by providing them with practical, hands-on experience working with the latest tools and technologies in the field.

Course Outcomes: Upon completion of this course, the learners will be able to

CO No.	CO Statement	Bloom's Taxonomy Level (K_1 to K_6)
CO_1	Demonstrate knowledge of PHP syntax, control structures, and built-in functions	K1, K2
CO_2	Design and implement PHP programs using object-oriented programming principles	K3, K4
CO_3	Use MySQL to create and manage databases and tables, and perform basic CRUD operations	K2, K3
CO_4	Implement JavaScript to create dynamic, interactive web applications	K3, K4, K5
CO_5	Implement best practices for web development, including security and performance optimization	K3, K4, K5

Course Content	Hours
Exercise-I: HTML and JavaScript	**12 hrs**
1. Create a webpage with an input box, and a button. Write a JavaScript function that takes	

Course Content	Hours
the input from the box and displays an alert message with the input. 2. Write a JavaScript function to check if a given string is a palindrome or not. 3. Create a webpage with a form that contains text fields for first name, last name, email, and password. Write a JavaScript function that validates the form by checking if all fields are filled out and the email is in a valid format. 4. Create a function that takes an array of integers as input and returns the sum of all even numbers in the array. 5. Create a JavaScript object representing a car with properties like make, model, year, and color. Write a function that takes a car object as input and displays its properties in an alert message. 6. Create a JavaScript function that takes two numbers as input and returns their average. 7. Create a webpage that displays a random quote each time the page is refreshed. Use an array of quotes and a JavaScript function to select a random quote and display it on the page. 8. Write a JavaScript function that takes an array of strings as input and returns the longest string in the array. 9. Create a webpage that has a button. When the button is clicked, a prompt should appear asking for the user's name. When the user enters their name, the page should display a greeting with their name. 10. Write a JavaScript function that takes a number as input and returns its factorial. 11. Create a webpage with a table that has rows and columns. Write a JavaScript function that highlights the row and column of a cell when it is clicked. 12. Write a JavaScript function that takes a number as input and returns true if it is a prime number, and false otherwise.	
Exercise-II: Installation and Structure of PHP, Functions and Scope of Variables	**12 hrs**
1. Write a PHP program to print "Hello, World!" on the web page using the echo command. 2. Create a PHP variable called $name and assign your name to it. Print the value of the variable on the web page. 3. Write a PHP program to calculate and print the area of a rectangle. The length and width of the rectangle should be stored in variables. 4. Write a PHP program to check whether a given number is even or odd using the modulus operator. 5. Create a PHP function to convert a temperature from Celsius to Fahrenheit. The function should accept a temperature in Celsius as an argument and return the temperature in Fahrenheit. 6. Write a PHP program to create a constant that stores the value of pi (3.14159265). 7. Create a PHP program that uses the if statement to check if a given number is positive or negative. Print a message indicating the result. 8. Write a PHP program that declares a global variable called $counter and increments it each time the program is run. Print the value of $counter on the web page. 9. Write a PHP program to calculate the factorial of a given number using a recursive function. **10.** Create a PHP program that uses the switch statement to print a different message based on the day of the week. For example, if it's Monday, print "Have a great start to your week!"	
Exercise-III: PHP Control Flow, Functions and Including Files in PHP.	**12 hrs**
Expressions: 1. Write a PHP program that assigns a value to a variable and uses the comparison operator to determine whether the value is equal to a specific number. If it is, display a message saying that the value is equal to the number. Otherwise, display a message saying that it is not equal. 2. Write a PHP program that takes two numbers as input from the user and performs some arithmetic operation on them (such as addition, subtraction, multiplication, or division)	

Course Content	Hours
based on a user-selected option. Use if-else statements to handle the different options. **Control Flow:** 1. Write a PHP program that uses a for loop to display the numbers from 1 to 10 on the screen. 2. Write a PHP program that uses a while loop to display the even numbers from 2 to 20 on the screen. 3. Write a PHP program that uses a switch statement to determine the appropriate message to display based on the day of the week (e.g. "It's Monday!" or "It's Friday!"). **PHP Functions:** 1. Write a PHP function that takes two numbers as input and returns the sum of those numbers. 2. Write a PHP function that takes an array of numbers as input and returns the average of those numbers. 3. Write a PHP function that takes a string as input and returns the string with all of the vowels removed. 4. Including and Requiring Files: 5. Write a PHP program that includes an external file containing a function, and then calls that function from the main program. 6. Write a PHP program that includes multiple external files, each containing a different function, and then calls all of those functions from the main program.	
Unit-IV: PHP Arrays, Functions and File Handling	**12 hrs**
PHP Arrays 1. Write a PHP script to create and display a numerically indexed array of 5 elements. 2. Create an associative array of car brands and models. Use a foreach loop to display the brands and their corresponding models. 3. Create a multidimensional array of student names and their grades in 3 subjects. Use a loop to display each student's name and their average grade. **Array Functions** 1. Write a PHP script to create an array of numbers and then use the count function to display the number of elements in the array. 2. Create an array of strings and use the sort function to sort them in ascending order. 3. Use the explode function to split a string into an array of words. Then use the compact function to create variables with the array keys as their names and the array values as their values. **File Handling** 1. Write a PHP script to check whether a file exists. If it does, display its contents. If it doesn't, create the file and write some contents to it. 2. Use PHP to copy an existing file to a new location and rename it. 3. Write a PHP script to upload a file to the server and display a success message if the upload was successful. **Cookies, Sessions and Auth** 1. Write a PHP script to set a cookie with a user's name and display a personalized greeting on subsequent page visits. 2. Use PHP sessions to store and retrieve a user's shopping cart contents across different pages. 3. Implement a login system in PHP using sessions to authenticate users and restrict access to certain pages.	
Unit-V: Object Oriented PHP and MySQL with PHPMyAdmin	**12 hrs**
PHP Objects	

Course Content	Hours
Declare a class named "Car" with properties like model, year, and color. • Create an object of the "Car" class and set its properties. • Access the properties of the object. • Create a constructor for the "Car" class that accepts parameters. • Create a destructor for the "Car" class. • Write a static method for the "Car" class that returns the number of objects created. • Create a subclass of "Car" named "SUV" with an additional property like "4WD". • Use a trait in the "SUV" class and write a method that uses the trait. • Implement an interface in the "SUV" class and define its methods. **MySQL and PHPMyAdmin** Create a database named "mydb" using PHPMyAdmin. • Create a table named "users" in the "mydb" database with columns like "id", "name", and "email". • Add a primary key and auto-increment property to the "id" column. • Add some sample data to the "users" table. • Use SQL queries to retrieve data from the "users" table and display it on a web page. • Write a PHP script that connects to the "mydb" database using the $_POST or $_GET array and inserts data into the "users" table. • Write a PHP script that updates data in the "users" table based on the user's input using $_POST or $_GET array. • Write a PHP script that deletes data from the "users" table based on the user's input using $_POST or $_GET array. • Use mysqldump to backup the "mydb" database and restore it to a new database. • Use SQL queries to create a new table named "orders" with foreign key relation to the "users" table.	

Text books:

- Nixon, Robin. Learning PHP, MySQL & JavaScript: With jQuery, CSS & HTML5. O'Reilly Media, Inc., 2014.
- DuBois, Paul. MySQL, 5th Edition. Addison-Wesley Professional, 2013.
- Flanagan, David. JavaScript: The Definitive Guide. O'Reilly Media, Inc., 2020.

Reference books:

- Lerdorf, Rasmus, Kevin Tatroe, and Peter MacIntyre. Programming PHP: Creating Dynamic Web Pages. O'Reilly Media, 2013.
- Zandstra, Matt. PHP Objects, Patterns, and Practice. Apress, 2017.
- Sklar, David. Learning PHP 7: A Pain-Free Introduction to Building Interactive Web Sites. O'Reilly Media, 2016.
- Flanagan, David. JavaScript: The Definitive Guide. O'Reilly Media, 2020.
- Duckett, Jon. JavaScript and jQuery: Interactive Front-End Web Development. Wiley, 2014.
- Eich, Brendan. JavaScript: The Good Parts. O'Reilly Media, 2008.
- DuBois, Paul. MySQL, 5th Edition. Addison-Wesley Professional, 2013.
- Yarger, Jon, Kevin Kline, and Bob Beauchemin. Pro MySQL. Apress, 2013.
  Banks, Matthew. MySQL Cookbook: Solutions for Database Developers and Administrators. O'Reilly Media, 2014.

Pedagogic tools:

- Chalk and Board
- Power point presentation
- Seminar
- Videos

Methods of Assessment & Tools:

Components of CIA: 40 marks

Sr. No.	Component	Content	Duration (if any)	Marks	Sub Total
A	Practical Skill Assessment	All Experiments	3 hours	30 (Set for 30)	30
B	Observation and Book of Records	All Experiments	-	10 (10 marks)	10
				Grand Total	**40**
Lab Methods:	• Hands-on exercises • Programming assignments • Demonstrations • Code reviews				
Assessment Tools:	• Self-assessments • Peer assessments • Code documentation				

The K-Levels (Knowledge Levels) are based on the Knowledge Dimension of Bloom's Taxonomy, which relates to the depth and complexity of knowledge required to achieve a particular learning outcome. The K-Levels are defined as follows:

K1: Recall of information
K2: Basic comprehension of concepts and ideas
K3: Application of knowledge and concepts to solve problems
K4: Analysis and synthesis of information to develop new ideas or solutions
K5: Evaluation and creation of new knowledge or products based on existing information and concepts

Note: Any other assessment tools or methods can be adopted as per requirement of the course.