M.Sc. (Computer Science)
Semester-3
MSCS-3-01P: Web Programming Lab

Total Marks: 50
External Marks: 15
Internal Marks: 35
Credits: 2
Pass Percentage: 40%

Course: Web Programming Lab	
Course Code: MSCS-3-01P	
Course Outcomes (COs)	
After the completion of this course, the students will be able to:	
CO1	Develop fully working applications that can be used on cross-platforms.
CO2	Create forms and check for data accuracy
CO3	Apply intermediate and advanced web development practices.
CO4	Create Web Pages that function using external data.
CO5	Develop a fully functioning website and deploy on a web server.

Detailed List of Programs:

Program No.	Name of Program
P1	Install Java Development Kit (JDK) on your system and set up the Java runtime environment. Verify the installation using command-line tools
P2	Write a simple Java program and compile it using the Java compiler. Run the compiled program and observe the output.
P3	Compare and contrast Java and C++ programming languages, highlighting key differences.
P4	Create a Java program that demonstrates the concept of byte code and explain how it is executed by the Java Virtual Machine (JVM).
P5	Implement a Java program that explores the usage of constants, variables, data types, and operators.
P6	Create a Java program to illustrate different types of inheritance, including single, multiple, and multilevel inheritance.
P7	Implement a program that showcases the use of the 'super' keyword to call

	superclass constructors.
P8	Develop a Java application that demonstrates method overriding and dynamic method dispatch.
P9	Write a program to read input from the console and display it using predefined input/output streams.
P10	Create a Java program that utilizes one-dimensional and two-dimensional arrays. Perform operations such as sorting or searching on these arrays.
P11	Implement a program that explores string handling using both the String and String Buffer classes.
P12	Define a package and demonstrate its usage in different Java classes. Import and utilize classes from other packages.
P13	Create interfaces with variables and implement them in Java classes to achieve multiple inheritance.
P14	Develop a program that uses both interfaces and abstract classes, showcasing their differences.
P15	Write a Java program that demonstrates the different types of exceptions. Implement try-catch blocks to handle exceptions effectively.
P16	Create a program with multiple try and catch clauses and observe how the program behaves in different scenarios.
P17	Implement a custom exception class and use it in your program to handle specific error conditions.
P18	Develop a Java program with multiple threads, each performing a different task. Use thread priorities and observe the execution order.
P19	Implement synchronization mechanisms in a multi-threaded program to avoid data race conditions.
P20	Create a program that demonstrates inter-thread communication and includes features like deadlock, thread suspension, resumption, and stopping.