AMITY UNIVERSITY
MADHYA PRADESH
Established vide Government of Madhya Pradesh Act No. 27 of 2010

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING
Course Handout
Course : ADVANCED JAVA PROGRAMMING LAB
Course Code : CSE524, Crédits : 01, Session: 2023-27(Odd Sem.), Class : B.Tech. 3rd Year

A. **Introduction:** The objective is to equip the students with the advanced feature of contemporary java which would enable them to handle complex programs relating to managing data and processes over the Enterprise network. The major objective of thiscourse is to provide a sound foundation to the students on the concepts, precepts andpractices, in a field that is of immense concern to the industry and business.

B. **Course Outcomes:** At the end of the course, students will be able to:

CO1: Ability to design and develop Java Applets, Beans programming

CO2: Ability to design and structure the Server Side Programming Concepts

CO3: Ability to Create and design Dynamic web Application.

CO4: Write the structured code for JDBC (back end database).

CO5: Ability to develop and design the enterprise level applications.

C. **Syllabus**

1. Servlet Basics

i) Create a simple Servlet to handle HTTP requests and responses.
ii) Implement doGet() and doPost() methods to handle different types of requests.
iii) Demonstrate Servlet lifecycle methods such as init(), service(), and destroy().
iv) Develop a web application to display user input using Servlets.

2. JavaServer Pages (JSP)

i) Create a basic JSP page to generate dynamic content.
ii) Use JSP scriptlets, expressions, and declarations to embed Java code in JSP.
iii) Implement JSP actions like include, forward, and useBean.
iv) Develop a web application combining Servlets and JSP to manage user authentication.

3. Spring Dependency Injection

i) Implement Dependency Injection (DI) using Spring framework.
ii) Configure Spring beans using XML and annotations.
iii) Demonstrate IoC (Inversion of Control) principle in Spring.
iv) Develop a Spring-based application to manage employee data with DI.

4.Spring MVC Architecture

i) Create a simple Spring MVC application with controllers and views.
ii) Define RequestMapping for handling HTTP requests.
iii) Implement form handling and validation using Spring MVC.

5.Spring Boot Application Development

i) Create a Spring Boot application using Spring Initializr.
ii) Explore Spring Boot starters and auto-configuration.
iii) Configure database connectivity using Spring Boot.
iv) Develop RESTful APIs using Spring Boot for CRUD operations on an entity.

Examination Scheme:

IA			EE			
A	**PR**	**Practical Based Test**	**Major Experiment**	**Minor Experiment**	**LR**	**Viva**
5	10	15	35	15	10	10

Note: IA – Internal Assessment, EE - External Exam, A - Attendance, PR- Performance, LR – Lab Record, V – Viva.

D. Lab Plan

Lab Session	Program Name	Mapped CO	Mode of Assessing CO
1	Create a simple Servlet to handle HTTP requests and responses. Implement doGet() and doPost() methods to handle different types of requests.	CO1	Viva/Quiz & End Sem Practical Exam
2	Demonstrate Servlet lifecycle methods such as init(), service(), and destroy(). Develop a web application to display user input using Servlets	CO1	Viva/Quiz & End Sem Practical Exam
3	Create a basic JSP page to generate dynamic content. Use JSP scriptlets, expressions, and declarations to embed Java code in JSP.	CO1	Viva/Quiz & End Sem Practical Exam
4	Implement JSP actions like include, forward, and useBean. Develop a web application combining Servlets and JSP to manage user authentication.	CO2	Viva/Quiz & End Sem Practical Exam
5	Implement Dependency Injection (DI) using Spring framework. Configure Spring beans using XML and annotations.	CO2	Viva/Quiz & End Sem Practical Exam
6	Demonstrate IoC (Inversion of Control) principle in Spring. Develop a Spring-based application to manage employee data with DI.	CO3	Viva/Quiz & End Sem Practical Exam
7	Create a simple Spring MVC application with controllers and views. Define RequestMapping for handling HTTP requests.	CO3	Viva/Quiz & End Sem Practical Exam
8	Create a Spring Boot application using Spring Initializr. Explore Spring Boot starters and auto-configuration.	CO4	Viva/Quiz & End Sem Practical Exam
9	Configure database connectivity using Spring Boot. Develop RESTful APIs using Spring Boot for CRUD operations on an entity.	CO4	Viva/Quiz & End Sem Practical Exam
10	Use Postman to test RESTful APIs developed in previous experiments. Send GET, POST, PUT, and DELETE requests using Postman.	CO4	Viva/Quiz & End Sem Practical Exam

11	Validate request parameters, headers, and response status codes. Implement error handling and exception responses in RESTful APIs.	CO2	Viva/Quiz & End Sem Practical Exam
12	Understand the concept of microservices and its advantages. Design RESTful APIs for microservices communication. Implement multiple microservices using Spring Boot. Demonstrate communication between microservices using REST calls.	CO5	Viva/Quiz & End Sem Practical Exam
13	Implement caching using Spring Boot annotations. Integrate Spring Data JPA for database operations. Implement logging using Spring Boot logging framework.	CO5	Viva/Quiz & End Sem Practical Exam
14	Deploy Spring Boot applications to a local server (e.g., Tomcat). Monitor and manage Spring Boot applications using Spring Boot Actuator. Configure logging levels and monitor application performance.	CO5	Viva/Quiz & End Sem Practical Exam

A. Course Articulation Matrix (Mapping of COs with POs)

CO	CORRELATION WITH PROGRAMME OUTCOMES												CORRELATIONWITH PROGRAMME SPECIFIC OUTCOMES		
	PO1	PO2	PO3	PO4	PO5	PO6	PO7	PO8	PO9	PO10	PO11	PO12	PSO1	PSO2	PSO3
CO1	3	1	1	1	2				1		1	1			
CO2	3	2	1	2	2				1		1	1			
CO3	3	2	2	2	2				1		1	1			
CO4	3	2	2	2	2				1		1	1			
CO5	2		1						1			1	1		