

MAULANA ABUL KALAM AZAD UNIVERSITY OF TECHNOLOGY, WEST BENGAL
(Formerly West Bengal University of Technology)
Syllabus of BCA
(Effective from 2023-24 Academic Sessions)

SEMESTER: VI

DEFINITION OF CREDIT

1 HR LECTURE PER WEEK	1 CREDIT
1 HR TUTORIAL PER WEEK	1CREDIT
2 HR PRACTICAL PER WEEK	1 CREDIT

SUBJECT NUMBERING SCHEME:

CODE FOR THE DEPT. OFFERING SUBJECT	SUBJECT TYPE	SEM	SUBJECT CODE
--	--------------	-----	--------------

C	CORE MAJOR
---	------------

SUBJECT NAME: Advance Java with Web Application
SUBJECT CODE: BCAC601

Credit: 5 (3L + 2P)

COURSE OBJECTIVE:

The **Advanced Java with Web Application** course aims to provide BCA students with a comprehensive understanding of Java technologies and their application in creating robust, scalable, and dynamic web applications. The course focuses on advanced Java concepts, including servlets, JSP, JDBC, and frameworks, to bridge the gap between core programming and real-world web solutions.

Students will learn to design and develop interactive web applications using Java technologies, emphasizing modular programming, database integration, and web standards. The course explores client-server architecture, multi-threading, session management, and MVC patterns, enabling students to build high-performance, secure, and maintainable web solutions.

This course equips students with practical skills, and problem-solving abilities to address modern challenges in web development and software engineering. It also fosters an understanding of emerging technologies and best practices in Java-based web application development.

Course Outcome	
	DESCRIPTION
CO1	Design and develop interactive, dynamic, and scalable web applications using advanced Java concepts like Servlets, JSP, and JDBC.
CO2	Implement client-server communication, session management, and the MVC architecture for structured web development.
CO3	Integrate database connectivity using JDBC to perform CRUD operations and ensure efficient and secure data handling.
CO4	Deploy web applications on servers (e.g., Apache Tomcat), incorporate RESTful services, and ensure robust application security.

DETAILED SYLLABUS:

Module	Topics Covered	Hours	Marks
M1	Introduction to Java EE: Overview of Java EE Architecture; Difference between Java SE and Java EE; Role of JDBC, JSP, and Servlets in Web Applications; Role of JDBC, JSP, and Servlets in Web Applications	3	5
M2	JDBC (Java Database Connectivity): Introduction to JDBC; JDBC Drivers and Architecture; JDBC Drivers and Architecture; Connecting to Databases; Executing SQL Queries (SELECT, INSERT, UPDATE, DELETE); Use of Statement and Prepared Statement; Result set and Metadata; Transaction management; Batch Processing; commit, rollback, save point	9	12
M3	Java Server Pages: Introduction to JSP; JSP Lifecycle; JSP syntax and directives; Scriptlet, Expression and declaration; JSP Implicit objects; JSP directives; JSP Action Element; Java Beans in JSP; Introduction to JSP Expression Language; Introduction to JSTL Core Tags(<c:if>, <c:choose>, <c:when>, <c:otherwise>, <c:forEach>, <c:forTokens>, <c:param>) and JSTL Functions; Custom Tag Library	12	15
M4	Servlet: Introduction to Servlets; Servlet Life cycle; Handling HTTP request and Responses; Purpose and use Servlet Deployment descriptor File; ServletContext and ServletConfig ; Session Management and Cookie; Servlet chaining and Filters; File Upload and download in servlet	12	15

M5	JSP and Servlet : Combining JSP and Servlets; Model View controller Architecture; Forwarding requests between JSP and Servlet	4	12
M6	Overview of Hibernate Framework; Advantages of ORM over JDBC; Hibernate Architecture and Core Components; Setting Up Hibernate in a Java Application; Mapping Java Classes to Database Tables; Hibernate Configuration (XML and Annotations); Basic CRUD Operations: Save, Update, Delete, Retrieve Hibernate Query Language (HQL) Basics	5	11
	INTERNAL EXAMINATION	3	30
	TOTAL	48	100

Practical:

SUBJECT NAME: Advance Java with Web Application Lab
SUBJECT CODE: BCAC691

Credit: 2

The practical instruction ought duly to encompass the theoretical curriculum in its entirety, and shall further entail the execution of a modest project, thereby affording the scholar an opportunity to apprehend the principles in earnest, and to apply such knowledge toward the resolution of problems encountered in the affairs of the real world.

Suggested Lab question:

Case Study 1: Online Bookstore Application

Scenario:

You are tasked to develop an Online Bookstore web application. The application should allow users to browse books, search for books by title or author, view details, and place orders. Admin users should be able to add, update, or delete book records.

Lab Questions:

1. Servlet Basics

Create a servlet to display the list of books from a database on a web page.

Implement a servlet to handle user searches by book title or author.

2. Session Management

Implement session tracking to maintain the user's cart.

Use HttpSession to store the list of books added to the cart.

3. JDBC Integration

Write a Java program to connect to a database and fetch book records.

Use Prepared Statement to insert, update, and delete book details securely.

4. JSP for Dynamic Pages

Create a JSP page to display the shopping cart's content dynamically.

Use JSTL (Java Server Pages Standard Tag Library) for iteration and conditional rendering.

5. MVC Architecture

Implement the Model-View-Controller pattern for the application.
Separate business logic (Model) from the presentation (View) and control logic (Controller).

6. Form Validation and Error Handling

Create a user registration form with validation (e.g., email, password strength) using JSP and servlets.

Implement error handling for invalid user input or database connection issues.

Case Study 2 : Employee Management System

Scenario:

Develop a web-based Employee Management System where employees can log in to view their profiles, and administrators can manage employee records.

Lab Questions:

1. Login Authentication

Create a login servlet to authenticate users based on username and password stored in a database.

Redirect users to different pages based on their roles (e.g., Employee or Admin).

2. CRUD Operations with JDBC

Write servlets to add, update, view, and delete employee records in a database.

Ensure proper validation and error handling for all database operations.

3. Pagination and Sorting

Implement pagination to display employee records on the admin dashboard.

Add sorting functionality for columns like Name, Department, and Salary.

4. Email Notification

Use JavaMail API to send an email notification to employees upon record creation or updates.

5. Deployment on Apache Tomcat

Package the application as a WAR file and deploy it on the Apache Tomcat server.

Test the application's functionality in a live environment.

SUGGESTED READING:

- Kogent Learning Solutions, JDBC 4.0, Dreamtech Press, 1st Edition
- R. S. Gohil, JDBC Programming, Laxmi Publications, 1st Edition
- S. G. T. Raghavan, Java and JDBC, Oxford University Press, 2nd Edition
- K. S. Shankar, Java Database Programming with JDBC, Wiley India, 1st Edition
- Kogent Learning Solutions, Java Server Programming (JSP, Servlets) Black Book, Dreamtech Press, 1st Edition
- Budi Kurniawan, Java Servlet & JSP, Wrox Press, 2nd Edition
- Hans Bergsten, JavaServer Pages, O'Reilly Media, 3rd Edition
- S. R. S. Sharma, Servlet & JSP Programming, PHI Learning, 1st Edition
- Ramesh F. Gujjula, Mastering Hibernate, BPB Publications, 1st Edition

- S. Vijayalakshmi, Hibernate: Java Persistence Framework, Pearson Education India, 1st Edition
- K. M. S. R. Anjaneyulu, Java Hibernate, Laxmi Publications, 1st Edition

SUBJECT NAME: Unix and Shell Programming
SUBJECT CODE: BCAC602

Credit: 5 (3L +2P)

COURSE OBJECTIVE:

The Unix and Shell Programming course is designed to provide BCA students with a thorough understanding of the Unix operating system and its powerful command-line interface. This course emphasizes the foundational concepts of Unix, including file system structure, process management, and inter-process communication, while focusing on shell scripting for automation and system management.

Students will explore the Unix environment, learning to navigate the file system, manipulate files, and manage processes efficiently. The course also introduces shell programming using popular Unix shells such as Bourne Shell, C Shell, and Korn Shell. Through scripting, students will develop the ability to automate repetitive tasks, schedule jobs, and manage system configurations.

By the end of the course, students will be well-equipped with the knowledge and skills to work in Unix-based environments, enabling them to pursue careers in system administration, software development, and IT support. The course fosters problem-solving abilities and a strong command over Unix tools and scripting techniques.

Course Outcome	
CO1	Demonstrate proficiency in using Unix commands to navigate the file system, manage files, and control processes.
CO2	Develop shell scripts to automate repetitive tasks, schedule jobs, and manage system configurations effectively.
CO3	Utilize Unix utilities such as grep, awk, and sed for file processing, text manipulation, and data extraction.
CO4	Implement inter-process communication and manage process control to optimize system performance and resource usage.
CO5	Apply knowledge of Unix tools and shell programming to solve real-world problems in system administration and IT operations.

DETAILED SYLLABUS:

Module	Topics Covered	Hours	Marks
M1	Introduction to Unix; Discuss about POSIX; Discuss about Linux and most popular distributions of Linux; Compare Between the Unix and Linux; Unix system Architecture; Discuss about the Unix kernel and system call interface; Unix directory structure;	2	5
M2	Unix File commands: create directory, change directory, move and copy directory, create file, remove file; remove directory, listing directory information; discuss about the types of files; change file security; creating the empty file and change the timestamp; discuss the access time, modification time and change time; touch with d and t option; finding the files and directorires; soft link and hard link.discuss about i node; size of file system.Use of > and >>, Use of pipe and tee command.	8	12
M3	row wise and column wise selection from the file with different options, merge lines of files horizontally with different options; split large files into smaller files, counting number of lines, words , characters; sorting the content of the file according to the specific column, numerical sorting, comparing two files line by line, compare two files character by character, compare two sorted files., join the two files, the uniq command, the transformation command.	7	14
M4	Utility command: cal, date, pr, who, bc, echo, zip unzip , gzip commands, Archiving the file	2	
M5	Vi editor: Overview of VI Editor as a text editor in Unix/Linux; Basic mode operations in VI (Command mode, Insert mode, and Last-line mode). Basic Operations in VI Editor (Starting VI: Opening and editing files using vi filename, Navigation: Moving the cursor (h, j, k, l, arrow keys, etc.), Moving by word, line, or page. Text Editing (Inserting text in insert mode (i, I, a, A, o, O), Deleting text (x, dd, dw, D, etc.), Copying and pasting (yy, pp, p, P), Undo and redo changes (u, Ctrl+r).). Working with Files (Saving and Exiting (Saving files: :w, :w filename), Exiting: :q, :q! (force quit without saving), Saving and quitting together: :wq, ZZ.). File Operations (Opening a new file: :e filename.)	2	5

	Search and Replace (Searching forward and backward, using regular expressions for searching.) Replace (Replacing text using :s/old/new/g., Global search and replace in a file.)		
M6	Searching the file with pattern using grep and awk command, Advance searching the file with grep, awk command with print and printf, awk with comparison operator, arithmetic operator, begin and else section, Begin and end section, if else statement, built in variable fs and ofs, awk with string and arithmetic functions, use of loops, Use of searching and substitute function.	8	15
M7	Process in Unix, Discuss the process command with different options, Discuss about the init process and unis login process, discuss briefly the fork() , getpid(), getppid(), wait(0, zombie process, , pipe() and message(), Discuss unix process states and the diagram, Discuss about the scheduler used in unix, swapped memory, discuu about the vmstat and top command, discussion about the nice command	8	9
M8	Shell programming: Introduction of shell and types of shell, use of shell, system variables and user defined variable, use of single and double quote and backslash, command substitution, let: assigning and evaluationg the expression, take input from the user, Command line parameters, use of if statement, use of for, while and until loop, Observe the exit status.	8	10
	INTERNAL EXAMINATION	3	30
	TOTAL	48	100

Practical:

SUBJECT NAME: Unix and Shell Programming Lab

Credit: 2

SUBJECT CODE: BCAC692

The practical sessions shall, with due diligence, reflect and reinforce the theoretical syllabus, and shall moreover include the undertaking of a modest yet purposeful project, designed to furnish the student with a sound comprehension of foundational principles, and to cultivate the capacity to apply such learning to the resolution of real-world exigencies.

List of Sample question for Unix and shell programming Lab:

Direct Lab Questions

1. Basic Unix Commands

List all files in a directory along with their permissions and ownership.

Display the top 10 largest files in a directory using `du` and `sort`.

Find and replace a specific word in a file using `sed`.

2. File and Directory Management

Write a shell script to create a directory structure for a project (e.g., `Project/Docs`, `Project/Src`, `Project/Bin`) and verify the structure.
Create a script to back up all `.txt` files in the current directory into a new directory named `Backup_<date>`.

3. Process Management

Write a script to list all processes owned by the current user.
Create a script to monitor CPU usage and alert the user if it exceeds a specific threshold.

4. Text Processing

Use `awk` to extract and display the second and fourth columns from a CSV file.
Use `grep` to find all lines in a file that contain a specific pattern and count the occurrences.

5. Shell Scripting Basics

Write a shell script to calculate the factorial of a number entered by the user.
Create a script to check if a given file exists and whether it is readable, writable, or executable.

Case Study-Based Lab Questions

System Monitoring Tool

Scenario:

Build a shell script-based tool to monitor and report system performance, including disk usage, memory usage, and active processes.

Lab Questions:

1. Write a script to display the following system details:
 - Total and free memory.
 - Disk usage percentage of each mounted partition.
 - Number of active processes.
2. Implement a feature to save the report to a file with a timestamp in the filename (e.g., `System_Report_<date>.txt`).
3. Extend the script to alert the user if disk usage exceeds 80% or memory usage goes below 10%.

SUGGESTED READING:

- Yashavant Kanetkar, *Unix Shell Programming*, BPB Publications, 1st Edition
- Sumitabha Das, *UNIX: Concepts and Applications*, Tata McGraw-Hill Education, 4th Edition
- Vikas Chawla, *Unix and Shell Programming*, Oxford University Press, 1st Edition
- M. G. Venkatesh, *Advanced Unix Programming*, Wiley India, 1st Edition
- Stephen G. Kochan, *Unix Shell Programming*, Pearson Education India, 2nd Edition
- S. R. Rajesh, *UNIX Shell Programming*, Pearson Education India, 1st Edition
- Brian W. Kernighan, *The UNIX Programming Environment*, Prentice Hall, 1st Edition

SUBJECT NAME: Networking
SUBJECT CODE: BCAC603

Credit: 4

COURSE OBJECTIVE:

The course aims to provide BCA students with a foundational understanding of computer networks, their architecture, and functionality. This course covers essential networking concepts, including network types, topologies, protocols, and models like OSI and TCP/IP. Students will learn how data is transmitted across networks, explore various communication mediums, and understand error detection and correction mechanisms.

The course also introduces addressing schemes, routing algorithms, and transport layer protocols, enabling students to grasp how devices communicate in a networked environment. Emphasis is placed on practical knowledge of network configuration, troubleshooting, and security basics to ensure reliable and secure data transmission.

By the end of the course, students will be able to design, analyze, and implement small-scale networks, preparing them for advanced studies and professional roles in the field of computer networking and IT infrastructure management.

Course Outcome	
CO1	Explain the fundamental concepts of computer networks, including types, topologies, and communication protocols.
CO2	Analyze and compare networking models such as OSI and TCP/IP to understand data flow across layers.
CO3	Demonstrate knowledge of addressing schemes, subnetting, and routing algorithms for efficient data communication.
CO4	Configure and troubleshoot small-scale networks using appropriate tools and techniques.
C05	Apply basic network security principles to ensure secure and reliable data transmission.

DETAILED SYLLABUS:

Module No.	Topics Covered	Duration (Hours)	Marks
M1	Introduction: Definition and Importance of Networking, Components of the network, Network Topology: LAN, MAN, WAN Network Topologies (Star, Ring, Bus, Mesh), Reference Models: OSI and TCP/IP reference model, A comparison of the OSI and TCP/IP reference models	6	6
M2	Physical Layer: Data Transmission Concepts, Transmission Media: Guided Media (Twisted pair, Co-axial cable, Optical fiber); Unguided Media (Radio, VHF, microwave, satellite, Infrared), Encoding Schemes: NRZ, Manchester, Differential Manchester; Multiplexing Techniques: FDM, TDM, WDM, Switching Techniques: Circuit, Packet, and Message Switching	10	12
M3	Data link layer: Data Link layer issues, Flow control (stop and wait protocol, sliding window Flow control), Error Detection Algorithms: Parity Check, CRC, Checksum, Error Correction: Hamming Code Error control (Stop and wait ARQ, Go-back-N ARQ, Selective reject ARQ), HDLC, Multiple Access Protocols (ALOHA, Collision Free Protocols). IEEE Standards for LAN (IEEE802.3, IEEE802.4, IEEE802.5).	11	15
M4	Network Layer: Network Layer: Routing algorithm (Shortest path algorithm, Flow based Routing, Distance vector routing, Broadcast routing, Multicast routing); IP Addressing: IPv4 and IPv6, Subnetting and CIDR, Routing Protocols: RIP, OSPF, BGP, NAT and ICMP.	10	12
M5	Transport Layer: Functions of Transport Layer, Protocols: TCP (3-Way and 4-way Handshaking), UDP, Port Addressing, Congestion Control Mechanisms: Leaky Bucket, Token Bucket	8	10
M6	Application Layer Protocols: DNS, HTTP, HTTPS, FTP, SMTP, POP3, IMAP; Network Applications: Remote Login (Telnet, SSH), File Sharing (NFS, SMB), Domain Name server, Simple network management Protocol.	8	10
M7	Emerging Networking concepts: Cloud Networking Basics, Internet of Things (IoT) Networking Protocols: MQTT, CoAP; Software-Defined Networking (SDN) Concepts	4	5
	INTERNAL EXAMINATION	3	30
	TOTAL	60	100

SUGGESTED READING:

- S. Tanenbaum, Computer Networks, Pearson Education India, 5th Edition
- William Stallings, Data and Computer Communications, Pearson Education India, 9th Edition
- James F. Kurose, Computer Networking: A Top-Down Approach, Pearson Education India, 7th Edition
- Larry L. Peterson, Computer Networks: A Systems Approach, Elsevier, 5th Edition
- Behrouz A. Forouzan, Data Communications and Networking, McGraw-Hill Education, 5th Edition
- Rajendra Prasad, Introduction to Computer Networks, Wiley India, 1st Edition