



Subject: Advanced Object-Oriented Programming-DETCE14211

Type of course: Minor Stream

Prerequisite: Basic Knowledge of JAVA and HTML

Rationale:

This course offers a comprehensive exploration of web-based Java programming, covering essential concepts and practical techniques. Students delve into technologies like JDBC and Hibernate for seamless database integration, server-side tools for robust Java applications, and web services for cross-platform communication. The course emphasizes client-service architecture's importance in web-based applications, addressing limitations of basic Java with advanced tools. Servlets is introduced for creating dynamic web applications. Web socket programming facilitates real-time communication between web clients and servers, providing a scalable alternative to HTTP. In summary, this course equips students with a strong foundation in web-based Java programming, empowering them to develop sophisticated applications and enable seamless internet communication.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks					Total Marks
CI	T	P	C	Theory Marks		Practical Marks		CA	
				ESE	MSE	V	P	ALA	
3	0	2	4	60	30	10	20	30	150

Legends: CI-ClassRoom Instructions; T – Tutorial; P - Practical; C – Credit; ESE - End Semester Examination; MSE- Mid Semester Examination; V – Viva; CA - Continuous Assessment; ALA- Active Learning Activities.



Course Content:

Sr. No	Course Content	Hrs.	% Weightage
1	Exception Handling & Multithreading: Types of errors, exceptions, try...catch statement, multiple catch blocks, throw and throws keywords, finally clause, uses of exceptions, user defined exceptions. Concept of Multithreading, creating thread, extending Thread class, implementing Runnable interface, life cycle of a thread, Thread priority, Thread exception handling in threads	07	15%
2	File Handling Explain basics of streams, stream classes, creation, reading and writing files in context to file handling. Network Programming with Java: Network Programming With java.net Package- Inet Address class, URL class, URL Connection class. Establishing two-way communication between Server and Client - TCP/IP client sockets, TCP/IP server sockets.	07	15%
3	Java Database Connectivity (JDBC) Describe the basics of JDBC and its connectivity. The JDBC API. The Statement Interface, Prepared Statement, Callable Statement, the Result Set Interface, Transaction processing – commit, rollback, save point Explain different types of JDBC drivers and their advantages and disadvantages. database operations like creating tables, operations using SQL CRUD operation.	11	25%
4	Hibernate: Overview of Hibernate, Hibernate Architecture, Hibernate Mapping Types, Hibernate O/R Mapping, Hibernate Annotation Hibernate Query Language	09	20%
5	Servlets: Introduction to Servlets, Life Cycle of Servlet. Creating, configuring and deploying echo servlet on Tomcat Server Parameters and Attributes – HttpServletRequest Interface, ServletContext and ServletConfig Interface, Request Delegation – RequestDispatcher Interface. Exploring Session Tracking Mechanisms. Connecting and reading database/table records and displaying them using servlet, Advantages and Disadvantages of Servlet.	11	25%



Continuous Assessment:

Sr. No	Active Learning Activities	Marks
1	Empowering Code: Unlocking the Potential of Java Students (Group of two students) are required to design a poster that illustrates the concept of "Code to Connection" using Java programming. This poster should effectively convey how Java code can be utilized to create and establish connections between various systems, devices, and applications, highlighting key principles such as network programming, client-server models, and the integration of different technologies. Upload Final poster on GMIU Web Portal.	10
2	Make one project using core JAVA(JDBC) with the help CRUD operation Users can perform CRUD operations on student data, such as adding new students, reading student details, updating student information, and deleting student records. The project utilizes core Java for logic implementation and JDBC (Java Database Connectivity) for database interaction. Upload solution (code) and output (animation screenshot) on GMIU Web Portal.	10
3	Make one project using web-based java (Servlet HTML/CSS) with the help of CRUD operation. Users can perform CRUD operations on student data, such as adding new students, reading student details, updating student information, and deleting student records. The project utilizes core Java for logic implementation and Servlet HTML/CSS (Java Database Connectivity) for database interaction with UX and UI. Upload solution (code) and output (animation screenshot) on GMIU Web Portal.	10
Total		30

Suggested Specification table with Marks (Theory): 60

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage %	25%	35%	20%	10%	05%	05%



Course Outcome:

After learning the course the students should be able to:	
CO1	Implement thread and learn about how to handle the exception in java.
CO2	Develop Java Application using file Handling, network protocols, socket programming, and related technologies
CO3	Implement basic database operations using JDBC.
CO4	Understand database-driven Java applications using Hibernate ORM framework.
CO5	Learn server-side programs using Servlets.

List of Practical:

Sr. No	Description	Unit No	Hrs.
1	Write a program in Java to develop user defined exceptions for 'Divide by Zero' error.	1	2
2	Write a program that executes two threads. One thread will print the even numbers and another thread will print odd numbers from 1 to 200.	1	2
3	Write a program that reads words from a text file (passed as a command-line argument) and displays all the non-duplicate words in descending order.	2	4
4	Write a client server program where the client sends two numbers and the server responds with a square of them.	2	2
5	Develop a chat application using socket programming.	2	4
6	Develop a program to present a set of choices for users to select a product and display the price of the product.	3	4
7	Develop a simple hibernate Web Application that displays all records stored in a student table having attributes student_id, student_name and student branch.	4	2
8	a) Write an HTML code to create a login form having one submit button, two textboxes labeled as Login name and Password as respectively. b) Write a Servlet class named as ReadParameter to read these two parameters and display entered parameters values on the page using the doGet() or doPost() method when the user clicked on the submit button.	5	4
9	Create a web form which processes servlet and demonstrates use of cookies and sessions.	5	4
10	Develop a student login application using MVC architecture (HTML & servlet). display whether a student gets Successfully logged in or not.	5	2
		Total	30



Instructional Method:

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory.

The internal evaluation will be done on the basis of Active Learning Assignment.

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.

Reference Books:

- [1] Java server programming javaee-7 j2ee 1.7), black book , dreamtech press.
- [2] J2ee: the complete reference , james edward keogh, mcgraw hill education.
- [3] The complete reference jsp 2.0, phil hanna, bpb publication.
- [4] Complete reference java 2, herbert schildt, mcgraw hill education
- [5] Web technology with advance java, soumadip ghosh, university press.





Gyanmanjari
Innovative University

Course Syllabus
Gyanmanjari Diploma Engineering College
Semester-4 (Diploma)

Subject: Advance Database Management System-DETCE14212

Type of course: Minor Stream

Prerequisite: Prior knowledge in computer science fundamentals, including programming.

Rationale:

In today's digital economy, businesses, governments, and other organizations rely heavily on databases to store and process large volumes of critical data. Advanced DBMS provides the necessary tools and techniques to handle this growing demand efficiently and effectively. By studying advanced DBMS concepts, students are equipped with the knowledge to build high-performance, scalable, secure, and reliable database systems that power modern applications. This training ensures that they are ready to meet the challenges of working with big data, distributed systems, and complex data models, making them highly competitive in the job market.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks					Total Marks
CI	T	P	C	Theory Marks		Practical Marks		CA	
				ESE	MSE	V	P	ALA	
3	0	2	4	60	30	10	20	30	150

Legends: CI-Class Room Instructions; T – Tutorial; P - Practical; C – Credit; ESE - End Semester Examination; MSE- Mid Semester Examination; V – Viva; CA - Continuous Assessment; ALA- Active Learning Activities.

Course Content:

Sr. No	Course content	Hrs.	% Weightage
1	Database Design Concepts: Introduction, Data and Information, Metadata, Normalization - 1NF, 2NF, 3NF, BCNF, 4NF, 5NF, De-normalization, Entity-Relationship (E-R) modeling, Schema design and implementation.	07	15%



2	SQL and Advanced Query Techniques: Data Definition Language (DDL)- CREATE, ALTER, TRUNATE, DROP, Data Manipulation Language(DML)-INSERT, SELECT, UPDATE, DELETE, Data Control Language (DCL)- Grant and revoke, Transaction Control Language(TCL)- Commit, save point, Rollback, Database Integrity constraints- Domain constraints, Entity integrity constraints, Group by, Having and Order by Clause, Sub queries and joins, set operations, Cursor static - Implicit and Explicit, fundamental of database Triggers – creating Triggers and types of Triggers.	11	25%
3	Transaction Management: ACID properties (Atomicity, Consistency, Isolation, Durability),Concurrency control techniques – Lock based protocol, Granting of lock, 2PL,Deadlock detection and resolution, introduction to parallel database, parallel database system architecture, types of parallelism, parallel database implementation, introduction to distribute database, apply parallel and distributed database technique, distributed database system architecture, benefits of distributed database system, issues with distributed database system.	09	20%
4	Database Security and Authorization: User roles and privileges, Encryption techniques, Data masking and auditing, DAC, MAC and RBAC models, SQL Injection.	07	15%
5	Distributed Databases: Architecture of distributed databases, Data replication and fragmentation, Consistency models.	04	10%
6	Database Backup and Recovery: Backup strategies (full, incremental, differential), Recovery Model - Simple Recovery Model, Full Recovery Model, Bulk-Logged Recovery Model, Recovery Techniques - Restore Techniques, Recovery Scenarios, Backup Strategies.	07	15%

Continuous Assessment:

Sr. No	Active Learning Activities	Marks
1	DDL Command Exploration: In this activity Students use an online SQL simulator (e.g., DB Fiddle or SQL Zoo) to practice the CREATE, ALTER, TRUNCATE and DROP DDL commands. They create tables, alter existing structures, truncate data, and drop tables while observing the effects in real-time. Students document their observations on how each command impacts the database schema and upload it on GMIU web portal.	10
2	Triggers and Trigger Types: In this activity Students create triggers (e.g., BEFORE INSERT, AFTER UPDATE) on tables in an online simulator. They simulate data modifications and examine how triggers automatically execute actions (e.g., logging changes, enforcing business	10



	rules). Students experiment with different types of triggers and reflect on their uses in automating database operations and upload it on GMIU web portal.	
3	Recovery Model Analysis: In this activity Students research different recovery models (e.g., full, bulk-logged, simple) and their impact on transaction logging and database performance. They configure and simulate recovery models on a database and observe the differences in transaction log size and restore behaviour and upload it on GMIU web portal.	10
Total		30

Suggested Specification table with Marks (Theory):60

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage %	25%	35%	20%	10%	05%	05%

Note: This specification table shall be treated as a general guideline for students and teachers. The actual distribution of marks in the question paper may vary slightly from above table.

Course Outcome:

After learning the course, the students should be able to:	
CO1	Design and Implement Efficient Databases.
CO2	Apply in SQL and Advanced Query Techniques.
CO3	Implement and Manage Database Security Measures.
CO4	Design and Optimize Distributed and Parallel Database Systems.
CO5	Understand of Backup and Recovery Strategies in Database Systems.



List of Practical

Sr. No	Description	Unit No	Hrs.
1	Implement DDL (CREATE, ALTER, DROP, TRUNCATE and DML (INSERT, UPDATE, DELETE) SQL commands.	2	2
2	Insert sample data into tables using INSERT command.	2	2
3	Apply all database Entity Integrity constraints (i.e. Primary key, Foreign key, NOT NULL, Unique and CHECK).	2	2
4	Apply all database Domain constraints.	2	2
5	Perform Group by, having and Order by clause.	2	2
6	Perform query for TCL(ROLLBACK, Commit and Save point) and DCL (Grant and Revoke) commands.	2	2
7	Implement query using joins(i.e. Inner Join and Outer Join)	2	2
8	Implement query using sub queries (3 layer)	2	2
9	Perform queries involving predicates LIKE, BETWEEN, IN etc.	2	2
10	Retrieve data spread across various tables or same table using various joins.	2	2
11	Implement programming using Cursor	2	2
12	Implement programming using Triggers.	2	2
13	Implement SQL Query using set operations like: Union, Union all, Minus etc.	2	2
14	Apply concept of security and privileges.	4	2
15	Implement locking protocols.	4	2
		Total	30



Instructional Method:

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory.

The internal evaluation will be done on the basis of Active Learning Assignment.

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.

Reference Books:

- [1] Database system concepts", 6th Edition by Abraham silberschatz, Henry F. Korth, S. Sudarshan, McGraw-Hill.
- [2] Database Management System by Ramakrishnan, Gehrke, Tata McGraw-Hill.
- [3] SQL-PL/SQL, Ivan bayross.
- [4] Database Systems: The complete book, Hector Garcia-Molina, Jeff Ullman, and Jennifer Widom.
- [5] SQL and Relational theory: How to write accurate SQL code: C.J. Date.





Course Syllabus
Gyanmanjari Diploma Engineering College
Semester-4 (Diploma)

Subject: Python Programming-DETCE14208

Type of course: Major (Core)

Prerequisite: Prior knowledge in computer science fundamentals, including programming.

Rationale:

Python is a versatile and powerful programming language that is crucial for developing software applications, data analysis, and automation. This subject covers fundamental programming concepts, object-oriented design, libraries, and frameworks necessary for building efficient and maintainable code. Proficiency in Python is essential for students pursuing careers in software development, data science, machine learning, and automation.

Teaching and Examination Scheme:

Teaching Scheme			Credits	Examination Marks					Total Marks
CI	T	P	C	Theory Marks		Practical Marks		CA	
				ESE	MSE	V	P	ALA	
3	0	2	4	60	30	10	20	30	150

Legends: CI-ClassRoom Instructions; T – Tutorial; P - Practical; C – Credit; ESE - End Semester Examination; MSE- Mid Semester Examination; V – Viva; CA - Continuous Assessment; ALA- Active Learning Activities.

Course Content:

Sr. No	Course content	Hrs.	% Weightage
1	Fundamentals of Python: Introduction to Python- Python as scripting Language, Programming language Vs Scripting Language, Python's building blocks- Identifiers, Keywords, Variables, Constants, Indentation, Comments in python, Python's Data Types, Input and Output statements in python, Operators in Python, Type Casting, Installation of python.	5	12%



2	Control Statements in Python: Types of Control Statements – Decision making statements, Looping statement: - while loop, for loop, nested loop, Manipulating Loops: - use of break, continue and pass statements.	9	20%
3	Data Structures in Python: List- Defining List, Creating list, Accessing values from list, Updating the elements of a list, Concatenation of two lists, Tuples- Defining Tuple, Creating Tuples, Accessing the Tuple elements, Inserting elements in a Tuple, modifying elements of a Tuple, Deleting elements from a Tuple, Sets- Defining Set, Creating a Set, Accessing elements from set, Add and update Set, Remove an elements from a Set, Dictionaries- Defining Dictionary, Creating Dictionary, Accessing elements from Dictionary, Add and update Dictionary, Delete an element from a Dictionary.	9	20%
4	Functions and Modules: Python Functions- Use of python built in functions, User defined function- Function definition, function calling, function arguments and parameter passing, Return statement, scope of variables.	11	24%
5	String Processing: Introduction to strings, accessing string elements via index operators, exploring various string functions such as basic functions, testing functions, searching functions, manipulation functions.	11	24%

Continuous Assessment:

Sr. No	Active Learning Activities	Marks
1	Debugging Detective: Faculty Provide students with Python program containing deliberate errors containing syntax and logic errors. They must identify and fix the errors to make the code run successfully. This activity enhances students' debugging skills and their understanding of common Python errors. Students need to upload the corrected Python code to the GMIU web portal.	10
2	Python Deep Dive: Interview Intensive Faculty will provide a set of Python interview-style questions covering topics like control flow, data types, functions, and error handling. These questions are designed to help students prepare for job interviews by testing their practical knowledge of Python. Students need to solve the questions individually and submit their answers on the GMIU web portal.	10



3	Code Translation Challenge: Java/C to Python Faculty will provide students with a program written in Java or C. Students will then convert the code into Python using advanced Python libraries and frameworks where applicable. This activity encourages students to understand code structure across languages and utilize Python's powerful libraries for efficient implementation. For marks, students need to upload the converted Python script to the GMIU web portal.	10
Total		30

Suggested Specification table with Marks (Theory): 60

Distribution of Theory Marks (Revised Bloom's Taxonomy)						
Level	Remembrance (R)	Understanding (U)	Application (A)	Analyze (N)	Evaluate (E)	Create (C)
Weightage %	30%	40%	20%	10%	-	-

Course Outcome:

After learning the course, the students should be able to:	
CO1	Understand the fundamental concepts of Python, including its syntax, structure, and basic programming constructs.
CO2	Apply control structures, loops, and functions to solve programming problems in Python.
CO3	Demonstrate the use of Python's built-in data structures (lists, tuples, sets, and dictionaries) for efficient data management and manipulation.
CO4	Implement modular programs using functions, standard libraries, and user-defined modules in Python.
CO5	Illustrate the process of file handling, string manipulation, and working with external modules in real-world applications.

List of Practical:

Sr. No	Description	Unit No	Hrs.
1	Install and configure the Python environment. Run basic Python, commands to verify the Python environment.	1	2
2	Write a program to read your name, contact number, email, and birthdate and print those details on the screen.	1	1
3	Write a program to convert temperature from Celsius to Fahrenheit. Equation to convert Celsius to Fahrenheit: $F = (9/5) * C + 32$.	1	2



4	Write a program to find a maximum of given three numbers.	1	1
5	A year is a Leap year if it is divisible by 4, unless it is a century year that is not divisible by 400 (1800 and 1900 are not leap years, 1600 and 2000 are leap years). Write a program that calculates whether a given year is a leap year or not.	2	2
6	Write a program to read the marks and assign a grade to a student. Grading system: A (≥ 90), B (80-89), C (70-79), D (60-69), E (50-59), F (< 50). (Use the Switch case)	2	2
7	Write a program to read n numbers from users and calculate the average of those n numbers.	2	1
8	Write a program that prompts the user to enter 10 integers and displays all the combinations of picking two numbers from the 10.	2	2
9	Write a program to perform the below operations on the list: <ul style="list-style-type: none"> • Create a list. • Add/Remove an item to/from a list. • Get the number of elements in the list. • Access elements of the list using the index. • Sort the list. • Reverse the list. 	3	2
10	Write a program to perform below operations on tuple: <ul style="list-style-type: none"> • Create a tuple with different data types. • Print tuple items. • Convert tuple into a list. • Remove data items from a list. • Convert list into a tuple. • Print tuple items. 	3	2
11	Write a program to print Fibonacci sequence up to n numbers using recursion.	4	2
12	Write a program that plays the popular scissor-rock-paper game. (A scissor can cut a paper, a rock can knock a scissor, and a paper can wrap a rock.) The program randomly generates a number 0, 1, or 2 representing scissor, rock, and paper. The program prompts the user to enter a number 0,1, or 2 and displays a message indicating whether the user or the computer wins, loses, or draws.	4	2
13	Write a program to check whether a given string is palindrome or not.	5	1
14	Write a program to perform the below operations on files: <ul style="list-style-type: none"> • Create a text file and write a string to it. • Read an entire text file. • Read a text file line by line. • Write a string to a file. • Write a list of strings to a file. • Count the number of lines, words in a file. 	5	2
15	Write a program to perform the following operations on matrices: <ul style="list-style-type: none"> • Create two matrices. 	5	2



	<ul style="list-style-type: none"> • Add the matrices. • Subtract the matrices. • Multiply the matrices. • Transpose the matrices. 		
16	Write a program to: <ul style="list-style-type: none"> • Check if a given number is a prime number. • Generate all prime numbers within a given range (input from the user). 	5	2
17	Write a program to perform the following operations on a dictionary: <ul style="list-style-type: none"> • Create a dictionary with at least 5 key-value pairs. • Add a new key-value pair. • Delete a key-value pair. • Access elements using a key. • Update the value of an existing key. • Print all keys, values, and key-value pairs. 	5	2
Total			30

Instructional Method:

The course delivery method will depend upon the requirement of content and need of students. The teacher in addition to conventional teaching method by black board, may also use any of tools such as demonstration, role play, Quiz, brainstorming, MOOCs etc.

From the content 10% topics are suggested for flipped mode instruction.

Students will use supplementary resources such as online videos, NPTEL/SWAYAM videos, e-courses, Virtual Laboratory.

The internal evaluation will be done on the basis of Active Learning Assignment.

Practical/Viva examination will be conducted at the end of semester for evaluation of performance of students in laboratory.

Reference Books:

- [1] Python for Dummies, Stef Maruch and Aahz Maruch, Wiley India, New Delhi.
- [2] Learning Python, Mark Lutz, O'Reilly Media.
- [3] Python Crash Course, Eric Matthes, No Starch Press.



