

Java Programming

Course Code: 4343203

Course Information

| Field | Details |
|------------|--|
| Program | Information and Communication Technology |
| Level | Diploma |
| Semester | Fourth |
| Curriculum | Competency-focused Outcome-based Green Curriculum-2021 (COGC-2021) |

Rationale

This course is designed to teach object-oriented programming concepts, techniques, and applications using the Java programming language. Object-oriented programming emphasis on the fundamentals of the structured design with classes, including development, testing, implementation and documentation also includes object-oriented programming techniques, classes and objects.

Java is a simple, portable, distributive, robust, secure, dynamic, architecture neutral, object oriented programming language. Java programming language is designed to enable the development of a small, reliable, portable, distributed, real-time operating platform, high-performance applications for the widest range of computing platforms possible as well as cross-platform interaction. By making applications available across heterogeneous environments, businesses can provide more services, boost end-user productivity, communication and collaboration to enterprise and consumer applications.

The Java programming language originated as part of a research project to develop advanced software for a wide variety of network devices and embedded systems. The Java programming language is used as the teaching vehicle for this course.

The aim of this course is that student should learn platform independent object oriented programming and java as base language for advanced technology like three tier architecture applications, cloud computing and web development. Java is also used to program electronic hardware devices such as air conditioners, televisions, refrigerators, washing machines and many others.

Competency

The aim of this course is to help the students to attain the following industry identified competency through various teaching-learning experiences:

Develop java application using object-oriented approach.

Course Outcomes (COs)

The practical exercises, the underpinning knowledge and the relevant soft skills associated with this competency are to be developed in the student to display the following COs:

Course Outcomes

- Write simple java programs for a given problem statement.
- Use object oriented programming concepts to solve real world problems.
- Develop an object-oriented program using inheritance and package concepts for a given problem statement.
- Develop an object oriented program using multithreading and exception handling for a given problem statement.
- Develop an object-oriented program by using the files and collection framework.

Teaching and Examination Scheme

| Teaching Scheme (Hours) | | | | Assessment Pattern (Marks) | | | | |
|-------------------------|---|----|---|----------------------------|-----------|--------------|---------------|-------|
| L | T | PR | C | Theory ESE | Theory CA | Practical CA | Practical ESE | Total |
| 3 | 0 | 4 | 5 | 70 | 30 | 25 | 25 | 150 |

Suggested Practical Exercises

| Sr. No | Practical Outcomes (PrOs) | Unit No. | Hrs. |
|--------|---|----------|------|
| 1 | *Install JDK, write a simple 'Hello World' or similar java program, compilation, debugging, executing using java compiler and interpreter. | I | 2 |
| 2 | Write a program in Java to find maximum of three numbers using conditional operator. | I | 1 |
| 3 | Write a program in Java to reverse the digits of a number using while loop | I | 1 |
| 4 | Write a program in Java to add two 3*3 matrices. | I | 2 |
| 5 | Write a program in Java to generate first n prime numbers. | I | 2 |
| 6 | Write a program in Java which has a class Student having two instance variables enrollmentNo and name. Create 3 objects of Student class in main method and display student's name. | II | 1 |
| 7 | *Write a program in Java which has a class Rectangle having two instance variables height and weight. Initialize the class using constructor. | II | 1 |
| 8 | Write a program in Java demonstrate the use of 'this' keyword. | II | 2 |
| 9 | Write a program in Java to demonstrate the use of 'static' keyword. | II | 2 |
| 10 | Write a program in Java to demonstrate the use of "final" keyword. | II | 2 |
| 11 | *Write a program in Java which has a class Shape having 2 overloaded methods area(float radius) and area(float length, float width). Display the area of circle and rectangle using overloaded methods. | II | 2 |
| 12 | Write a program in Java to demonstrate the constructor overloading. | II | 2 |

| Sr. No | Practical Outcomes (PrOs) | Unit No. | Hrs. |
|--------|---|----------|------|
| 13 | Write a java program to demonstrate use of 'String' class methods: <code>charAt()</code> , <code>contains()</code> , <code>format()</code> , <code>length()</code> , <code>split()</code> | II | 1 |
| 14 | Write a program in Java to demonstrate single inheritance | III | 1 |
| 15 | Write a program in Java to demonstrate multilevel inheritance | III | 2 |
| 16 | Write a program in Java to demonstrate hierarchical inheritance. | III | 2 |
| 17 | Write a program in Java to demonstrate method overriding. | III | 2 |
| 18 | *Write a program in Java which has a class Car having two instance variables <code>topSpeed</code> and <code>name</code> . Override <code>toString()</code> method in Car class. Create 5 instances of Car class and print the instances. | III | 2 |
| 19 | Write a program in Java to implement multiple inheritance using interfaces. | III | 2 |
| 20 | *Write a program in Java which has an abstract class Shape having three subclasses: Triangle, Rectangle, and Circle. Define method <code>area()</code> in the abstract class Shape and override <code>area()</code> method to calculate the area. | III | 4 |
| 21 | Write a program in Java to demonstrate use of final class. | III | 2 |
| 22 | Write a program in Java to demonstrate use of package. | III | 2 |
| 23 | Write a program in Java to develop user defined exception for 'Divide by Zero' error. | IV | 2 |
| 24 | *Write a program in Java to develop Banking Application in which user deposits the amount Rs 25000 and then start withdrawing of Rs 20000, Rs 4000 and it throws exception "Not Sufficient Fund" when user withdraws Rs. 2000 thereafter. | IV | 2 |
| 25 | *Write a program that executes two threads. One thread displays 'Thread1' every 1000 milliseconds, and the other displays 'Thread2' every 2000 milliseconds. Create the threads by extending the Thread class | IV | 2 |
| 26 | 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. | IV | 2 |
| 27 | *Write a program in Java to perform read and write operations on a Text file. | V | 2 |
| 28 | Write a program in Java to demonstrate use of List. 1) Create ArrayList and add weekdays (in string form) 2) Create LinkedList and add months (in string form) Display both List. | V | 2 |
| 29 | Write a program in Java to create a new HashSet, add colors(in string form) and iterate through all elements using for-each loop to display the collection. | V | 2 |
| 30 | *Write a Java program to create a new HashMap, add 5 students' data (enrolment no and name). retrieve and display the student's name from HashMap using enrolment no. | V | 2 |

Underpinning Theory

Introduction to Java Programming Language

Unit Outcomes:

- 1a Describe java features and applications and environment setup of Java programming language
- 1b Install Java Components
- 1c Write simple program using java programming language
- 1d Describe data types, identifiers, constants and variables
- 1e Write programs using arrays
- 1f List types of operators
- 1g Write simple java programs using decision and control structures

Object Oriented Programming Concepts

Unit Outcomes:

- 2a Differentiate between POP and OOP
- 2b List object oriented programming concepts
- 2c Develop simple java program using class
- 2d Use this and final keyword
- 2e Write object oriented program using constructor
- 2f Write java program using String class

Inheritance, Packages & Interfaces

Unit Outcomes:

- 3a List types of inheritance
- 3b Write program to implement single, multilevel, hierarchical inheritance
- 3c Write programs to implement method overriding
- 3d Write programs to implement overriding using Object class
- 3e Write programs to implement multiple inheritance
- 3f Create a user-defined package and use that package

Exception Handling & Multithreading

Unit Outcomes:

- 4a Describe errors and types of exceptions
- 4b List types of errors
- 4c Write user-defined exceptions
- 4d Define thread, creating threads, multithreading, thread priority & synchronization

File Handling and Collections Framework

Unit Outcomes:

- 5a Describe basics of streams, stream classes, creation, reading and writing files in context to file handling
- 5b Describe Collections framework
- 5c Write programs using ArrayList and LinkedList
- 5d Write programs to Map classes