GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT

COURSE CURRICULUM
COURSE TITLE: JAVA PROGRAMMING
(COURSE CODE: 3350703)

Diploma Program in which this course is offered	Semester in which offered
Computer Engineering/ Information Technology	5th Semester

1. RATIONALE:

Open source platforms play significant role in the corporate world and are gaining popularity because these are freeware and ease of access. Java is a simple, portable, distributive, robust, secure, dynamic, architecture neutral, object oriented programming language. This technology allows the software designed and developed once for an idealized 'virtual machine' and run on various computing platforms. Companies of all sizes are using Java as the main programming platform to develop various applications/projects worldwide. 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. Many commercial applications as well as developing mission critical applications are using Java Technologies. This necessitates the corporate sectors to hire highly skilled Java developers. So, after learning this course, student can float themselves as Java developer in the software industry as well this course works as foundation course for advance Java programming for the forthcoming semester.

2. LIST OF COMPETENCY:

The course content should be taught and implemented with the aim to develop required skills so that students are able to acquire following competency:

- **Develop software applications using object oriented concept in an Java SDK environment**

3. COURSE OUTCOMES:

The theory should be taught and practical should be carried out in such a manner that students are able to acquire different learning out comes in cognitive, psychomotor and affective domain to demonstrate following course outcomes.

i. Explain object oriented programming concepts of java.
ii. Comprehend building blocks of OOPs language, inheritance, package and interfaces.
iii. Identify exception handling methods.
iv. Develop multithreading object oriented programs.
v. Develop an object oriented program handling data file.

4. TEACHING AND EXAMINATION SCHEME

Teaching Scheme (In Hours)			Total Credits (L+T+P)	Examination Scheme				
				Theory Marks		Practical Marks		Total Marks
L	T	P	C	ESE	PA	ESE	PA	200
3	0	4	7	70	30	40	60	

Legends: L - Lecture**; T** - Tutorial/Teacher Guided Student Activity**; P** - Practical**; C** - Credit**; ESE** - End Semester Examination**; PA** - Progressive Assessment

5. COURSE CONTENT DETAILS

Unit	Major Learning Outcomes (in cognitive domain)	Topics and Sub-topics
Unit – I Introduction to Java	1a. Describe Internet role, advantages and, environment setup of Java.	1.1 Basics of Java, Background/History of Java, Java and the Internet, Advantages of Java 1.2 Java Virtual Machine & Byte Code 1.3 Java Environment Setup 1.4 Java Program Structure
	1b. Differentiate between POP and OOP	1.8 Procedure-Oriented vs. Object-Oriented Programming concept
	1c. List important OOP fundamentals	1.9 Basics of OOP: Abstraction, Inheritance, Encapsulation, Classes, subclasses and super classes, Polymorphism and Overloading, message communication
	1d. Write simple programs using java	1.10 Compiling and running a simple "Hello World" program: Setting Up Your Computer, Writing a Program, Compiling, Interpreting and Running the program, Common Errors
Unit – II Building Blocks of the Language	2a. Explain Data types: constant and variables	2.1 Primitive Data Types : Integers, Floating Point type, Characters, Booleans etc 2.2 User Defined Data Type 2.3 Identifiers & Literals 2.4 Declarations of constants & variables 2.5 Type Conversion and Casting 2.6 Scope of variables & default values of variables declared 2.7 Wrapper classes 2.8 Comment Syntax 2.9 Garbage Collection
	2b. State the steps to implement programs for Arrays and String Handling	2.10 Arrays of Primitive Data Types 2.11 Types of Arrays 2.12 Creation, concatenation and conversion of a string, changing case of string, character extraction, String

		Comparison, String Buffer
	2c. List different types of operators	2.13 Different Operators: Arithmetic, Bitwise, Rational, Logical, Assignment, Conditional, Ternary, Increment and Decrement, Mathematical Functions
	2d. State the steps to implement small programs using Decision & Control Structures	2.14 Decision & Control Statements: Selection Statement (if, if...else, switch), Loops (while, do-while, for), Jump statements (break, continue, return & exit)
Unit – III Object Oriented Programming Concepts	3a.Define Objects and Classes and methods	3.1 Defining classes, fields and methods, creating objects, accessing rules, this keyword, static keyword, method overloading, final keyword,
	3b.Explain Constructors & its types, Object as a parameter, constructor overloading	3.2 Constructors: Default constructors, Parameterized constructors, Copy constructors, Passing object as a parameter, constructor overloading
Unit– IV Inheritance, Packages & Interfaces	4a. Describe Inheritance and method overriding 4b. List the types of Inheritance	4.1 Basics of Inheritance, Types of inheritance: single, multiple, multilevel, hierarchical and hybrid inheritance, concepts of method overriding, extending class, super class, subclass, dynamic method dispatch & Object class
	4c. Describe Creating package, importing package, access rules for packages, class hiding rules in a package 4d. Define interface. 4e. Explain inheritance on interfaces, implementing interface, multiple inheritance using interface	4.2 Creating package, importing package, access rules for packages, class hiding rules in a package. 4.3 Defining interface, inheritance on interfaces, implementing interface, multiple inheritance using interface
	4f.Describe Abstract & final classes	4.4 Abstract class and final class
Unit – V Exception Handling & Multithreaded	5a. Explain errors, & exceptions 5b. List types of errors	5.1 Types of errors, exceptions, try..catch statement, multiple catch blocks, throw and throws keywords, finally clause, uses of exceptions, user defined exceptions

Programming	5c. Define thread, creating threads, multithreading, thread priority & synchronization	5.2 Creating thread, extending Thread class, implementing Runnable interface, life cycle of a thread, Thread priority & thread synchronization, exception handing in threads
Unit – VI File Handling	6a. Explain basics of streams, stream classes, creation, reading and writing files in context to file handling	6.1 Stream classes, class hierarchy, useful I/O classes, creation of text file, reading and writing text files

6. SUGGESTED SPECIFICATION TABLE WITH HOURS & MARKS (THEORY)

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks			
			R Level	U Level	A Level	Total
1.	Introduction to Java	04	4	3	0	7
2.	Building blocks of the Language	08	4	4	6	14
3.	Object Oriented Programming Concepts	06	4	4	6	14
4.	Inheritance, Packages and Interfaces	10	4	4	6	14
5.	Exception Handling, Multithreaded Programming	10	4	4	6	14
6.	File Handling	04	0	3	4	07
	Total	**42**	**20**	**22**	**28**	**70**

Legends: R = Remember; U = Understand; A = Apply and above levels (Bloom's revised taxonomy)

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

7. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises are designed to develop different types of skills of the competency. Following is the list of practical problems.

Sr. No	Unit No.	Exercise/Practical (Outcomes in Psychomotor Domain)	Approximate Hrs.
1	1	Install JDK, write a simple "Hello World" or similar java program, compilation, debugging, executing using java compiler and interpreter.	2
2	2	Write a program in Java to generate first n prime numbers.	2
3	2	Write a program in Java to find maximum of three numbers using conditional operator	1
4	2	Write a program in Java to find second maximum of n numbers without using arrays	2
5	2	Write a program in Java to reverse the digits of a number using while loop	1
6	2	Write a program in Java to convert number into words & print it	2
7	2	Write programs in Java to use Wrapper class of each primitive data types	4
8	2	Write a program in Java to multiply two matrix	2
9	3	Write a static block which will be executed before main( ) method in a class.	1
10	3	Write a program in Java to demonstrate use of **this** keyword. Check whether **this** can access the private members of the class or not.	1
11	3	Write a program in Java to develop overloaded constructor. Also develop the copy constructor to create a new object with the state of the existing object.	2
12	3	Write a program in Java to demonstrate the use of private constructor and also write a method which will count the number of instances created using default constructor only.	2
13	3	Write a program in Java to demonstrate the use of **'final'** keyword in the field declaration. How it is accessed using the objects.	1
14	3	Develop minimum 4 program based on variation in methods i.e. passing by value, passing by reference, returning values and returning objects from methods.	2
15	4	Write a program in Java to demonstrate single inheritance, multilevel inheritance and hierarchical inheritance.	3
16	4	Create a class to find out whether the given year is leap year or not. (Use inheritance for this program)	2
17	4	Write an application that illustrates how to access a hidden variable. Class **A** declares a static variable **x**. The class **B** extends **A** and declares an instance variable **x**. **display( )** method in **B** displays both of these variables.	2
18	4	Write a program in Java in which a subclass constructor invokes the constructor of the super class and instantiate the values.	2
19	4	Write a program that illustrates interface inheritance. Interface **P12** inherits from both **P1** and **P2**. Each interface declares one constant and	4

		one method. The class **Q** implements **P12**. Instantiate **Q** and invoke each of its methods. Each method displays one of the constants.	
20	4	Write an application that illustrates method overriding in the same package and different packages. Also demonstrate accessibility rules in inside and outside packages.	4
21	4	Describe **abstract** class called **Shape** which has three subclasses say **Triangle**, **Rectangle**, **Circle**. Define one method **area()**in the abstract class and override this **area()** in these three subclasses to calculate for specific object i.e. area() of Triangle subclass should calculate area of triangle etc. Same for **Rectangle** and **Circle**	2
22	4	Write a program in Java to demonstrate implementation of multiple inheritance using interfaces.	2
23	4	Write a program in Java to demonstrate use of final class.	1
24	5	Write a program in Java to develop user defined exception for 'Divide by Zero' error.	2
25	5	Write a program in Java to demonstrate multiple try block and multiple catch exception	1
26	5	Write an small application in Java to develop Banking Application in which user deposits the amount Rs 1000.00 and then start withdrawing of Rs 400.00, Rs 300.00 and it throws exception "Not Sufficient Fund" when user withdraws Rs. 500 thereafter.	2
27	5	Write a program that executes two threads. One thread displays “Thread1” every 2,000 milliseconds, and the other displays “Thread2” every 4,000 milliseconds. Create the threads by extending the **Thread** class	2
28	5	Write a program that executes two threads. One thread will print the even numbers and the another thread will print odd numbers from 1 to 50.	2
29	5	Write a program in Java to demonstrate use of synchronization of threads when multiple threads are trying to update common variable.	2
30	6	Write a program in Java to create, write, modify, read operations on a Text file.	2
Total			60

8. SUGGESTED LIST OF STUDENT ACTIVITIES

Following is the list of proposed student activities like:

i. Study available small Java application on internet and reuse in your application
ii. Develop Java object oriented application programs
iii. Present the application developed

9. SUGGESTED LEARNING RESOURCES

(A) List of Books:

Sr.No	Authors	Title of Books	Publication
1	Herbert Schildt	Java: The Complete Reference, Seventh Edition	Tata McGraw Hill
2	E Balagurusamy	Programming with Java	Tata McGraw Hill
3	Cay S. Horstmann, Gray Cornell	Core Java, Vol I-Fundamentals	Java Series, Sun MicroSystem
4	Sachin Malhotra & Saurabh Choudhary	Programming in JAVA, Second Edition	Oxford

(B) List of Major Equipment/Materials

i. Computer System with latest configuration and memory
ii. Multimedia projector
iii. Internet Access
iv. Access to library resources

(C) List of Software/Learning Websites

i. Java Development Kit: http://www.oracle.com/technetwork/java/javase/downloads/index.html
ii. http://docs.oracle.com/javase/specs/jls/se7/html/index.html
iii. http://docs.oracle.com/javase/tutorial/java/index.html
iv. http://www.tutorialspoint.com/java/
v. http://www.learnjavaonline.org/
vi. http://www.c4learn.com/javaprogramming/
vii. http://www.learn-java-tutorial.com/
viii. http://www.tutorialspoint.com/javaexamples/

10. SPECIAL INSTRUCTIONAL STRETEGIES (If Any)

The course activities include Lectures and Practical Exercises as per teaching scheme.

i. Conceptual knowledge will be shared interactively using multimedia projector.
ii. Student should be given environment to develop sample applications using JAVA under guidance of Teachers.

11. COURSE CURRICULUM DEVELOPMENT COMMITTEE

Faculty members from Polytechnics

- **Prof. R. M. Shaikh**, H.O.D Computer Department, K. D. Polytechnic, Patan
- **Prof. K. N. Raval**, H.O.D Computer Department, R. C. Technical Institute, Ahmedabad
- **Prof. M. P. Mehta**, Sr. Lecturer in Computer Technology, K. D. Polytechnic, Patan
- **Prof. H. P. Chauhan**, Lecturer(IT), Government Polytechnic, Himmatnagar
- **Prof A. S. Galathiya**, Lecturer in Computer Department, R. C. Technical Institute, Ahmedabad
- **Prof. H.J. Prajapati,** Lecturer(IT), Government Polytechnic, Himmatnagar
- **Prof. J. S. Upadhyay,** Lecturer and Head, IT, K P T I T, Viramgam

Coordinator and Faculty Members from NITTTR Bhopal

- **Dr. Shailendra Singh,** Professor & Head, Dept. of Computer Engineering and Applications.
- **Dr. James K. Mathai**, Associate Professor, Dept. of Computer Engineering and Applications.