

**GUJARAT TECHNOLOGICAL UNIVERSITY, AHMEDABAD, GUJARAT****COURSE CURRICULUM****COURSE TITLE: JAVA PROGRAMMING  
(COURSE CODE:3350703)**

<b>Diploma Programme in which this course is offered</b>	<b>Semester in which offered</b>
Computer Engineering/ Information Technology	5 <sup>th</sup> Semester

**1. RATIONALE :**

Open source platforms play significant role in the corporate world. Which are gaining popularity because it is freeware and ease of access. Java is a simple, portable, distributed, 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. This course is having aim 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 using Java technologies. This necessitates the corporations to hire highly skilled java developers. So after learning this course student can float themselves as java developer in the software industry as well foundation course for advance java programming for the forthcoming semester.

**2. COMPETENCY:**

The course content should be taught and implemented with the aim to develop various types of related skills leading to the achievement of the following competency

- **Be able to use Java SDK environment to create, debug and run simple Java programs.**

**3. Course Outcomes:**

1. Understand concept of class, object and methods.
2. Explain Inheritance.
3. Create Packages.
4. Identify Exception handling method.
5. Understand multithreading Programming.

#### 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	
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

**Note:** It is the responsibility of the institute heads that marks for **PA of theory & ESE and PA of practical** for each student are entered online into the GTU Portal at the end of each semester within the dates specified by GTU.

#### 5. COURSE DETAILS

Unit	Major Learning Outcomes	Topics and Sub-topics
<b>Unit – I Introduction to Java</b>	1a. Explain the advantages and basic features of Java.	1.1 What is Java? 1.2 Background/History of Java 1.3 The Internet and Java's place in it 1.4 Applications and Applets 1.5 Java Virtual Machine 1.6 Byte code - not an executable code
	1b. Differentiate between POP and OOP	1.7 Procedure-Oriented vs. Object-Oriented Programming
	1c. List important OOP fundamentals	1.7 Basics of OOP: Abstraction, Inheritance, Encapsulation, Classes, subclasses and super classes, Polymorphism and Overloading
	1d. Write simple programs using java	1.8 Compiling and running a simple "Hello World" program: Setting Up Your Computer, Writing a Program, Compiling, Interpreting and Running the program, Common errors
<b>Unit – II Introduction to language basics in java</b>	2a. List the data types available in Java and utilize them in applications.	2.1 Primitive Data Types : Integers, Floating Point type, Characters, Booleans 2.2 User Defined Data Type 2.3 Declarations 2.4 Constants 2.5 Identifiers & Literals 2.6 Type of Conversion and Casting 2.7 Objects and Wrapper Classes 2.8 Variables : Variable Definition and Assignment, Default Variable Initializations
	2b. Implement programs for Array and wrapper	

	classes	2.9 Command Line Arguments 2.10 Arrays of Primitive Data Types 2.11 Comment Syntax 2.12 Garbage Collection
	2c. List different types of operators	2.13 Using Operators: Arithmetic, Bitwise, Rational, Logical, Assignment, Conditional, Ternary, Auto Increment and Decrement
	2d. Implement small programs using Control Structures	2.14 Using Control Statements: Selection Statement (If, Switch), Loops (While, Do-while, for), Jump statements (Break, Continue, return)
<b>Unit – III Object Oriented Programming Concepts</b>	3a. Explain three important concepts of OOP	3.1 Abstraction 3.2 Encapsulation 3.3 Polymorphism & Overloading
	3b. Define Objects and Classes and the relationship between them	3.4 Fundamentals of Classes: A simple class, Creating Class Instances, Adding methods to a class, Calling Functions/Methods 3.5 Using 'this' keyword
	3c. Explain types of constructor with example	3.6 Constructors: Default constructors, Parameterized constructors 3.7 More on methods: Passing by Value by Reference, Access Control, Methods that Return Values, Method Overloading, Recursion 3.8 Nested and Inner classes
<b>Unit– IV Inheritance</b>	4a. Explain importance of Inheritance	4.1 Inheritance: Using 'extends' keyword, Subclasses and Superclasses, 'super' keyword usage, Overriding Methods, Dynamic Method Dispatch
	4b. Explain method overriding and incorporate in developed programs	4.2 The Object class
	4c. Distinguish final and abstract class, implement programs of abstract class and final class	4.3 abstract and final Classes
	4d. Describe Packages & interfaces with example	4.4 Packages: Defining a package, Importing a package, Access Controls 4.5 Interfaces: Defining an interface, Implementing and applying interfaces
<b>Unit – V</b>	5a. List types of error in Java	5.1 Basic Exceptions

<b>Exception Handling &amp; Multithreaded Programming</b>	& Incorporate exception handling in developed application	5.2 Proper use of exceptions 5.3 User defined Exceptions 5.4 Catching Exception: try, catch 5.5 Throwing and re-throwing: throw, throws 5.6 Cleaning up using the finally clause
	5c. Discuss the life cycle of Thread 5d. Distinguish to create thread using Runnable and Thread class, write steps to create thread using both the techniques	5.7 Create/instantiate/start new threads: .Extending java.lang.Thread, Implementing java.lang. Runnable Interface 5.8 Understand thread execution 5.9 Thread Priorities
	5d. Explain concept of synchronization 5e. Implement small application based on Thread	5.10 Synchronization  5.11 Inter-thread communication, Deadlock
<b>Unit – VI Handling Strings</b>	6a. Write program using string functions in java programming	6.1 Creation, Concatenation and conversion of a string 6.2 Changing case 6.3 Character Extraction 6.4 String Comparison 6.5 Searching strings 6.6 Modifying strings 6.7 String Buffer 6.8 java.io package

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

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks (Duration – 42 Hours)			
			R Level	U Level	A Level	Total
1.	Introduction to Java	05	4	3	0	7
2.	Introduction to language basics in java	06	4	4	4	12
3.	Object Oriented Programming Concepts	12	4	4	8	16
4.	Inheritance	08	4	4	6	14
5.	Exception Handling & Multithreaded Programming	08	4	4	6	14
6.	Handling Strings	03	0	3	4	07

Unit No.	Unit Title	Teaching Hours	Distribution of Theory Marks (Duration – 42 Hours)			
			R Level	U Level	A Level	Total
	<b>Total</b>	<b>42</b>	<b>20</b>	<b>22</b>	<b>28</b>	<b>70</b>

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

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

## 7. SUGGESTED LIST OF EXERCISES/PRACTICAL

The practical/exercises should be properly designed and implemented with an attempt to develop different types of skills so that students are able to acquire the competency.

Following is the list of experiments for guidance.

Sr. No	Unit No.	Experiment	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	Develop minimum five programs to explore java data types, operators, control structures and decision statements.	4
3	2	Develop minimum 2 programs to explore labelled loop.	2
4	2	Develop minimum 2 programs to explore arrays.	2
5	2	Develop minimum 2 programs to explore wrapper classes.	2
6	3	Define minimum 5 different classes such as Box, student, distance, shape, employee, feet, time etc. with member variables & methods. Also Develop programs to test those classes functionality.	6
7	3	Develop minimum 4 program based on variation in methods i.e. passing by value, passing by reference, returning values from methods.	4
8	3	Develop minimum 4 programs based on overloading methods and recursion methods.	4
9	3	Modify created classes in during the previous practical sessions to provide constructor and retest all classes.	4
10	4	Develop minimum 5 programs based on inheritance.	6
11	4	Develop minimum 2 programs based on package and also test all the visibility modifiers.	4
12	4	Develop minimum 2 programs to explore the multiple inheritance concept using interfaces.	4
13	5	Develop minimum 4 programs using exception handling class.	4
14	5	Develop minimum 2 programs using simple thread concept.	4
15	5	Develop minimum 2 programs using multiple thread concepts.	4
16	5	Develop minimum 2 programs using synchronized threading.	4
17	6	Develop minimum 3 programs to explore all string functionality.	4

18	6	Develop minimum 2 programs to explore knowledge of java.io.	4
<b>Total</b>			<b>68</b>

## 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 related small applications
- 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 software :  
<http://www.oracle.com/technetwork/java/javase/downloads/index.html>
- ii. <http://docs.oracle.com/javase/tutorial/java/index.html>
- iii. <http://www.tutorialspoint.com/java/>
- iv. <http://www.learnjavaonline.org/>
- v. <http://www.c4learn.com/javaprogramming/>
- vi. <http://www.learn-java-tutorial.com/>
- vii. <http://www.tutorialspoint.com/javaexamples/>

## 10. INSTRUCTIONAL STRETEGIES

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. M. A. Rizvi**, Associate Professor, Dept. of Computer Engineering and Applications.
- **Dr. R. K. Kapoor**, Associate Professor, Dept. of Computer Engineering and Applications, NITTTR