Lok Jagruti Kendra University
University with a Difference

Diploma in Cloud Computing & Big Data

LJ Polytechnic

Course Code: 025100502

Java Programming

Programmme / Branch Name	Diploma in Cloud Computing & Big Data					
Course Name	Java Programming				**Course Code**	025100502
Course Type	HSSC	BSC	ESC	PCC	OEC	PEC

Legends: HSSC: Humanities and Social Sciences Courses BSC: Basic Science Courses
ESC: Engineering Science Courses PCC: program Core Courses
OEC: Open Elective Courses PEC: program Elective Courses

1. Teaching and Evaluation Scheme

Teaching Hours / Week / Credits				**Evaluation Scheme**			
L	**T**	**P**	**Total Credit**	**CCE**	**SEE (Th)**	**SEE (Pr)**	**TOTAL**
3	0	4	5	50	50	50	150

Legends:
L: Lectures T: Tutorial P: Practical
CCE: Continuous & Comprehensive Evaluation
SEE (Th): Semester End Evaluation (Theory)
SEE (Pr): Semester End Evaluation (Practical)

2. Prerequisites

- ✓ Basic knowledge of C programming
- ✓ Basic knowledge of computer.

3. 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 a Java developer in the software industry as well. This course works as foundation course for advance Java programming for the forthcoming semester.

4. Objectives

✓ 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.

- Explain object-oriented programming concepts of Java.
- Comprehend building blocks of OOPs language, inheritance, package and interfaces.
- Identify exception handling methods.
- Develop multithreading object-oriented programs.

5. Contents

Unit No.	Topics	Sub-Topics	Learning Outcomes	% Weightage	Hours
1	**Fundamentals of Java Programming**	1.1. Introduction to Java 1.2. History of Java and Features 1.3. Java Virtual Machine and Byte Code 1.4. Types of Java program 1.5. Basic Concept of OOP 1.6. Procedure Oriented v/s Object Oriented 1.7. Write Simple Program 1.8. Basic Data Types and Operators	• Basics and history of Java • Various types of Java program • Basic OOP Concepts • Differentiate between POP and OOP • Knowledge about compiling and running a program. • Knowledge of data types and operators	15	6
2	**Control Flow and Array**	2.1. Variable and Types of Variable 2.2. Types Casting and Conversion 2.3. Wrapper Class 2.4. Decision and Control Statements 2.5. Array and Types of Array 2.6. Garbage Collection 2.7. Command Line Arguments	• Knowledge of basic programming language concepts • Basic of Wrapper class • Implement small program using decision and control statements	20	8
3	**Object Oriented Programming Concepts**	3.1. Class and Object 3.2. Constructor and Types of Constructors 3.3. Method and Method Overloading 3.4. This Keyword 3.5. Static Keyword	• Define class, object and method of class • Implement constructor overloading program.	25	10

		3.6. String Class and Its Methods 3.7. I/O Classes and File Handling	• Implement a string program using String class. • Knowledge of I/O Stream classes and File Handling		
4	**Inheritance, Packages and Interfaces**	4.1. Basic of Inheritance 4.2. Types of Inheritance 4.3. Method Overriding 4.4. Super Keyword 4.5. Dynamic Method Dispatch 4.6. Final Keyword 4.7. Abstract Class and Method 4.8. Interface 4.9. Packages	• Describe Inheritance and Method Overriding • List types of Inheritance • Implementation of program using super and final keywords • Define Interface • Creating Package and importing Package, access rules of Package.	20	10
5	**Exception Handling and Multithreaded Programming**	5.1. Types of Error 5.2. Basic Concepts of Exception Handling 5.3. Try and Catch Block 5.4. Throw and Throws 5.5. User Define Exception 5.6. Introduction of Thread 5.7. Implementation of Thread 5.8. Thread Life Cycle. 5.9. Thread Method 5.10. Multithreading	• Explain exception and errors • List types of error and exceptions • Define throw and throws • Implement program for user defined exception. • Define thread and create thread. • Explain thread life cycle • Implement thread program	20	8
				Total Hours	**42**

6. List of Practicals / Exercises

The practical/exercises should be properly designed and implemented in an attempt to develop different types of skills so that students can acquire the competencies/Programme outcomes. Following is the list of practical exercises for guidance.

Sr. No.	Practical / Exercises	Key Competency	Hours
1	Install JDK, write a simple "Hello World" or similar Java program, compilation, debugging, executing using Java compiler and interpreter.	To be able to install JDK	2
2	Write a Java program to find maximum of three numbers.	Understanding basic condition statements	2
3	Write a simple Java program using loops.	To be able to use basic loops	4
4	Write a Java program for matrix multiplication.	To be able to use an array	2
5	Write a Java program to use of Wrapper class of each primitive data types.	To be able to use Wrapper class	2
6	Write a Java program to use of command line arguments.	To be able to use Command Line Arguments	2
7	Write a Java program to demonstrate class and object.	To understand and demonstrate the class and object	2
8	Write a Java program for method overloading.	To be able to use method overloading	2
9	Write a Java program for constructor overloading.	To be able to use constructor overloading	2
10	Write a program in Java to demonstrate use of **this** keyword.	To be able to use "this" keyword	2
11	Write a program in Java to demonstrate use of **static** keyword.	To be able to use "static" keyword	2
12	Develop minimum four programs based on variation in methods i.e. passing by value, passing by reference, returning values and returning objects from methods.	To be able to use of variation in method	2
13	Write a program in Java perform create, write and read operations on a text file.	To be able to use text file operations	2
14	Write a program in Java to demonstrate single inheritance, multilevel inheritance and hierarchical inheritance.	To be able to perform all inheritance types	4
15	Write a program in Java to demonstrate method overriding.	To be able to use method overriding	4
16	Write a program in Java to demonstrate the use of **'super'** keyword.	To be able to use the "super" keyword	2
17	Write a program in Java to demonstrate the use of **'final'** keyword.	To be able to use the "final" keyword	2

18	Write a program in Java for use of abstract method and class.	To be able to use abstract method and class	2
19	Write a program that illustrates interface inheritance. Interface P12 inherited from both P1 and P2. Each interface declares one constant and one method. The class Q implements P12. Instantiate Q and invoke each of its methods. Each method displays one of the constants.	To understand how to illustrate interface inheritance	2
20	Write an application that illustrates method overriding in the same package and different packages. Also demonstrate accessibility rules in inside and outside packages.	To understand how to illustrate method overriding in same package and in different packages	2
21	Write a program in Java to develop user defined exception for 'Divide by Zero' error.	To be able to use user defined exception	2
22	Write a program in Java to demonstrate multiple try block and multiple catch exceptions.	To be able to use multiple try and catch blocks	2
23	Write a small application in Java to develop Banking Application in which user deposits the amount Rs 1000.00 and then starts withdrawing of Rs 400.00, Rs 300.00 and it throws exception "Not Sufficient Fund" when user withdraws Rs. 500 thereafter.	To understand how to develop small applications in Java programming	2
24	Write a program that executes two threads. One thread will print the even numbers and another thread will print odd numbers between 1 to 50.	To be able to use thread	2
25	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.	To be able to use threads by extending the Thread class	2
		Total Hours	**56**

7. Suggested Specification Table with Hours

Unit No.	Chapter Name	Teaching Hours	Distribution of Topics According to Bloom’s Taxonomy					
			R %	U %	App %	C %	E %	An %
1	Fundamentals of Java programming	6	40	30	20	-	5	5
2	Control Flow and Array	8	30	30	20	10	5	5
3	Object Oriented programming Concepts	10	20	30	30	10	5	5
4	Inheritance, Packages and Interfaces	10	20	20	20	10	15	15
5	Exception Handling and Multithreaded programming	8	20	20	30	10	10	10

Legends: R: Remembering U: Understanding
App: Applying C: Creating
E: Evaluating An: Analyzing

8. Textbooks

1) programming with Java, E Balagurusamy.
2) Core Java Vol I : Fundamentals, Cay S. Horstmann, Gray Cornell.
3) programming in Java, Sachin Malhotra andSaurabh Choudhary.

9. Reference Books

1) Java: The Complete Reference, Herbert Schildt, Seventh Edition.

10. Open Sources (Website, Video, Movie)

1) http://www.tutorialspoint.com/Java/
2) http://www.learnJavaonline.org/
3) http://www.tutorialspoint.com/Javaexamples/