## Title of Paper :Core Java

	Heading	Particulars	
1	Description the course :	Core Java course focuses on tea	ching students
	Including but Not limited to:	how to design, develop, and ma	_
	_	software applications using the	Java
		programming language. The cou	urse covers
		fundamental to advanced conce	epts of Java,
		enabling students to understand	d object-
		oriented programming (OOP) pr	rinciples, data
		structures, algorithms, and real-	-world
		application development.	
2	Vertical :	Major	
3	Type:	Theory	
4	Credits:	2 credits (1 credit = 15 Hours fo	or Theory in a
		semester, Total 30 hours)	,
5	Hours Allotted :	30 Hr	
6	Marks Allotted:	50	
7	Course Objectives(CO):		
	CO 1: Understand and Apply Object-Oriented Programming (OOP)		
	Concepts.		
	CO 2: Identify the key components of a class and object in Java, including		
	attributes (fields), methods, and constructors.		
	CO 3: Apply sound software engineering principles in Java by organizing		
	code into classes and methods with proper access control identifiers		
	CO 4: Use tools and techniques like unit testing, as well as IDE		
	debugging tools to find and fix iss	ues within Java programs.	
	CO 5. Effectively use Java's collection framework (e.g., Lists, Sets, Maps) to		
	manage and process groups of related objects.		
	CO 6. Use OOP concepts in designing and building solutions to real-world		
	nrobleme encuring the application	n is modular, maintainable, and	
		ii is inioaaiai, maintamasie, ana	
	reusable.	mis modular, manitamasic, and	
8	reusable. Course Outcomes (OC):		
8	reusable.  Course Outcomes (OC):  OC1. Understand the basics of	Java and its runtime environmen	ıt.
8	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Ja	Java and its runtime environmen	it. atements, and
8	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in	Java and its runtime environmen ava's data types, control flow standeritance, and exception handling	it. atements, and
8	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacobs OOP principles such as classes, in OC3. Creating own classes and	Java and its runtime environmen ava's data types, control flow sta heritance, and exception handlin d objects	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using James OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using the course of th	Java and its runtime environmen ava's data types, control flow standeritance, and exception handling	nt. atements, and ng.
9	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:-	Java and its runtime environmen ava's data types, control flow sta heritance, and exception handlin d objects	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:-  Module 1:	Java and its runtime environmend ava's data types, control flow standeritance, and exception handling objects and caption handling Class, Interface and exception	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:-  Module 1:  Introduction to Java Programmin	Java and its runtime environmen ava's data types, control flow stanheritance, and exception handling displayed objects ang Class, Interface and exception	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:-  Modules:-  Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platform)	Java and its runtime environment ava's data types, control flow standeritance, and exception handling dobjects and Class, Interface and exception g-History of Java and its form Independence, Object-	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:-  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfor Oriented), Data Types and Variables	Java and its runtime environmen ava's data types, control flow stanheritance, and exception handling displayed objects ang Class, Interface and exception	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob OOP principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:-  Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfo Oriented), Data Types and Variable Literals, Type Casting	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and caption exception g-History of Java and its form Independence, Objectes, Operators Constants and	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfo Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and Class, Interface and exception g-History of Java and its form Independence, Objectes, Operators Constants and Statements, Switch Statement,	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfo Oriented), Data Types and Variable Literals, Type Casting  Decision Making and Loops :If-else Loops (For, While, Do-While), Break and	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and class, Interface and exception g-History of Java and its form Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements	it. atements, and ng. handling
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platford Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else Loops (For, While, Do-While), Break and Classes and Objects: Array, Arrays Stri	Java and its runtime environment ava's data types, control flow standeritance, and exception handling dobjects and Class, Interface and exception of Java and its form Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements and String methods,	nt. atements, and ng.
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfor Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else Loops (For, While, Do-While), Break and Classes and Objects: Array, Arrays Stristing Buffer and String Builder, Object-O	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and Class, Interface and exception ge-History of Java and its form Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements and class and String methods, riented Programming Concepts,	it. atements, and ng. handling
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfor Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else Loops (For, While, Do-While), Break and Classes and Objects: Array, Arrays Stri String Buffer and String Builder, Object-O Defining Classes and Creating Objects, In	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and class, Interface and exception g-History of Java and its form Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements and String methods, riented Programming Concepts, instance Variables and Methods,	it. atements, and ng. handling
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfor Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else Loops (For, While, Do-While), Break and Classes and Objects: Array, Arrays Stri String Buffer and String Builder, Object-O Defining Classes and Creating Objects, In Constructors, this Keyword, super key	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and Class, Interface and exception ge-History of Java and its orm Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements and Class and String methods, riented Programming Concepts, astance Variables and Methods, word, Types of Classes, Scope	it. atements, and ng. handling
	reusable.  Course Outcomes (OC):  OC1. Understand the basics of OC2. Be proficient in using Jacob Principles such as classes, in OC3. Creating own classes and OC4. Develop mini projects using Modules:  Modules:- Module 1:  Introduction to Java Programmin Evolution, Features of Java (Platfor Oriented), Data Types and Variable Literals, Type Casting Decision Making and Loops: If-else Loops (For, While, Do-While), Break and Classes and Objects: Array, Arrays Stri String Buffer and String Builder, Object-O Defining Classes and Creating Objects, In	Java and its runtime environment ava's data types, control flow standeritance, and exception handling objects and Class, Interface and exception ge-History of Java and its orm Independence, Objectes, Operators Constants and Statements, Switch Statement, Continue Statements and Class and String methods, riented Programming Concepts, astance Variables and Methods, word, Types of Classes, Scope	it. atements, and ng. handling

	Inheritance: Its types, Superclass and methods	d Subclass, Final classes and	
	Polymorphism: Compile-time and Runtime Polymorphism		
	Module 2:  Interfaces: Defining and Implementing Interfaces, Abstract Classes and Methods, Multiple Interface Implementation  Packages: Introduction to predefined packages, User Defined Packages, Access specifier, Java Built-in packages		
	<b>Exception handling</b> - Try, Catch, and Fina Keywords	Ily Blocks, Throw and Throws	
	Introduction to Threads: Creating a	nd Running Threads,Thread	
	Lifecycle		
10	Books and References:		
	1. Java: The Complete Reference Herbert Schildt MC-Graw HILL 12th EDITION		
	2022		
	2. Core Java, Volume I: Fundamentals Hortsman Pearson 9th 2013		
	3. Core Java, Volume II: Advanced F	eaturesGary Cornell and Hortsman	
	Pearson 8th 2008		
12	Internal Continuous Assessment: 40%	Semester End Examination: 60%	
13	Continuous Evaluation through:	Format of Question Paper: External	
	Class test of 1 of 15 marks	Examination (30 Marks)— 1 hr duration	
	Class test of 2 of 15 marks		
	Average of the two: 15 marks		
	Quizzes/ Presentations/		
	Assignments: 5 marks		
	Tatal: 30 manulus		
4.4	Total: 20 marks	d Franciscotions 20 Mardo Duration 4 have	
14	Format of Question Paper: (Semester Er	nd Examination: 30 Marks. Duration:1 hour)	
14	Format of Question Paper: (Semester Er Q1: Attempt any two (out of four) from I	Module 1 (15 marks)	
14	Format of Question Paper: (Semester Er Q1: Attempt any two (out of four) from I Q2: Attempt any two (out of four) from I	Module 1 (15 marks)	
14	Format of Question Paper: (Semester Er Q1: Attempt any two (out of four) from I Q2: Attempt any two (out of four) from I Or	Module 1 (15 marks) Module 2 (15 marks)	
14	Format of Question Paper: (Semester Er Q1: Attempt any two (out of four) from I Q2: Attempt any two (out of four) from I	Module 1 (15 marks) Module 2 (15 marks)  Module 1 (15 marks)	