SGK GOVERNMENT DEGREE COLLEGE
Department of Computer Science, Lesson Plan

Name of the Department : Computer Science	
Name of the Lecturer : D.SIVA PHANINDRA	
Group / Course : II B.Sc.(comp)., Semester IV	
Paper : Object Oriented Programming Using Java	
Name of the Topic	**FUNDAMENTALS OF OBJECT – ORIENTED PROGRAMMING && Java Language**
Hours Required	14
Learning Objectives	**FUNDAMENTALS OF OBJECT – ORIENTED PROGRAMMING:** Object Oriented paradigm, Basic Concepts of OOP, Benefits of OOP, Applications of OOP, Java features. **OVERVIEW OF JAVA LANGUAGE**: Introduction, Simple Java program structure, Java tokens, Java Statements, Implementing a Java Program, Java Virtual Machine, Command line arguments. **CONSTANTS, VARIABLES & DATA TYPES:** Introduction, Constants, Variables, Data Types, Declaration of Variables, Giving Value to Variables, Scope of variables, Symbolic Constants, Type casting, Getting Value of Variables, Standard Default values; **OPERATORS & EXPRESSIONS**.
Previous Knowledge to be reminded	Basic programming concepts
Topic Synopsis	Object-oriented programming (OOP) is a programming paradigm based upon objects (having both data and methods) that aims to incorporate the advantages of modularity and reusability. The basic concepts of OOP are: Class, Object, Inheritance, Encapsulation, Polymorphism, Dynamic Binding, message communication and Data abstraction. Modularity, reusability, flexibility and effective problem solving gained by using oops. Some applications of OOPs are: Client-Server Systems, Object-Oriented Databases, Object-Oriented Databases, Real-Time System Design, Simulation and Modeling System, Hypertext and Hypermedia, Neural Networking and Parallel Programming and Office Automation Systems etc. Java is Object Oriented, Platform Independent, Simple, Secure, Architecture-neutral, Portable, Robust, Multithreaded, Interpreted, High Performance, Distributed and Dynamic Language. A typical structure of a Java program contains the following elements: Documentation Section Package Declaration Import Statements Interface Section Class Definition Class Variables and Variables Main Method Class Methods and Behaviors

	Java tokens are smallest elements of a program which are identified by the compiler. Java token includes the following: Keywords, Identifiers, Literals, Operators, Separators and Comments. In Java, a statement is an executable instruction that tells the compiler what to perform. Java statements can be broadly classified into the following categories: Expression Statements, Declaration Statements and Control Statements. In Java, a statement is an executable instruction that tells the compiler what to perform. Java statements can be broadly classified into the following categories: Expression Statements, Declaration Statements and Control Statements. JVM (Java Virtual Machine) is an abstract machine. It is a specification that provides runtime environment in which java byte code can be executed. Command line arguments are nothing but simply arguments that are specified after the name of the program in the system's command line, and these argument values are passed on to your program during program execution. Constant is a value that cannot be changed after assigning it. In Java, The final modifier represents that the value of the variable cannot be changed. It also makes the primitive data type immutable or unchangeable. Variable in Java is a data container that saves the data values during Java program execution. Data types specify the different sizes and values that can be stored in the variable. There are two types of data types in Java. The primitive data types include boolean, char, byte, short, int, long, float and double, etc.,. The non-primitive data types include Classes, Interfaces, and Arrays, etc. To **declare** (create) a variable, you will specify the type, leave at least one space, then the name for the variable and end the line with a semicolon (;). Java. **Type casting** is when you assign a value of one primitive data type to another type. In Java, there are two types of casting: Widening Casting (automatically) - converting a smaller type to a larger type size byte -> short -> char -> int -> long -> float -> double Narrowing Casting (manually) - converting a larger type to a smaller size type double -> float -> long -> int -> char -> short -> byte The standard default values for the Data Type, Byte, Short, int, long are: 0, float and Double are 0.0, boolean is false and Char is '\u0000' or null. Operator in Java is a symbol that is used to perform operations. For example: +, -, *, / etc.

	There are many types of operators in Java which are: Unary Operator, Arithmetic Operator, Shift Operator, Relational Operator, Bitwise Operator, Logical Operator, Ternary Operator and Assignment Operator. An expression is a construct made up of variables, operators, and method invocations, which are constructed according to the syntax of the language, that evaluates to a single value.
Thrust Areas	
Skills to be learnt by Students	
Examples / Illustrations	
Additional Inputs	
Teaching Aids Used	Yes
References Cited	E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.
Student Activity Planned after Teaching	Group discussion
Any other Activities	Assignment

K.R.K GOVERNMENT DEGREE COLLEGE
Department of Computer Science, Lesson Plan

Name of the Department : Computer Science	
Name of the Lecturer : D.SIVA PHANINDRA	
Group / Course : II B.Sc(comp)., Semester IV	
Paper : Object Oriented Programming Using Java	
Name of the Topic	**DECISION MAKING & BRANCHING**
Hours Required	10
Learning Objectives	Problem solving with various DECISION MAKING & BRANCHING Statements CLASSES, OBJECTS & METHODS: Defining a class, Adding variables, Adding methods, Creating objects, Accessing class members, Constructors, Method overloading, Static members, Nesting of methods
Previous Knowledge to be reminded	
Topic Synopsis	The statements inside your source files are generally executed from top to bottom, in the order that they appear. Control flow statements, however, break up the flow of execution by employing decision making, looping, and branching, enabling your program to conditionally execute particular blocks of code. This section describes the decision-making statements (if-then, if-then-else, switch), the looping statements (for, while, do-while), and the branching statements (break, continue, return) supported by the Java programming language. main object-oriented programming terms. Class: A class describes the contents of the objects that belong to it. It describes an aggregate of data fields (called instance variables), and defines the operations (called methods). Object: An object is an element (or instance) of a class; objects have the behaviors of their class. The object is the actual component of programs, while the class specifies how instances are created and how they behave. Method: A method is an action which an object is able to perform. Sending a message: sending a message to an object means asking the object to execute or invoke one of its methods. The object is a basic building block of an OOPs language. In Java, we cannot execute any program without creating an object. Java provides five ways to create an object. Using new Keyword Using clone() method Using newInstance() method of the Class class Using newInstance() method of the Constructor class Using Deserialization Accessing data members and member functions: The data

	members and member functions of class can be accessed using the dot('. ') operator with the object. A **constructor** in Java is a special method that is used to initialize objects. The constructor is called when an object of a class is created. It can be used to set initial values for object attributes. If a class has multiple methods having same name but different in parameters, it is known as Method Overloading. In Java, static members are those which belongs to the class and you can access these members without instantiating the class. The static keyword can be used with methods, fields, classes (inner/nested), blocks. A method of a class can be called only by an object of that class using the dot operator. So, there is an exception to this. A method can be called by using only its name by another method of the same class that is called **Nesting of Methods.**
Thrust Areas	
Skills to be learnt by Students	Database design through ER-model and reducing the ER diagram into tables
Examples / Illustrations	Few the ER diagram into tables
Additional Inputs	
Teaching Aids Used	Yes
References Cited	E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.
Student Activity Planned after Teaching	Group discussion
Any other Activities	Assignment

K.R.K GOVERNMENT DEGREE COLLEGE
Department of Computer Science, Lesson Plan

Name of the Department : Computer Science	
Name of the Lecturer : D.SIVA PHANINDRA	
Group / Course : II B.Sc(comp)., Semester I	
Paper : Object Oriented Programming Using Java	
Name of the Topic	**INHERITANCE & ARRAYS, STRINGS AND VECTORS**
Hours Required	12
Learning Objectives	**INHERITANCE**: Extending a class, Overloading methods, Final variables and methods, Final classes, Abstract methods and classes; **ARRAYS, STRINGS AND VECTORS:** Arrays, One-dimensional arrays, Creating an array, Two – dimensional arrays, Strings, Vectors, Wrapper classes. INTERFACES: MULTIPLE INHERITANCE: Introduction, Defining interfaces, Extending interfaces, Implementing interfaces, Assessing interface variables
Previous Knowledge to be reminded	Basic data and file structures, basic database concepts, mathematical knowledge
Topic Synopsis	The extends keyword extends a class (indicates that a class is inherited from another class). In Java, it is possible to inherit attributes and methods from one class to another. Overloading happens when you have two methods with the same name but different signatures (or arguments). In a class we can implement two or more methods with the same name. If we initialize a variable with the final keyword, then we cannot modify its value. If we declare a method as final, then it cannot be overridden by any subclasses. And, if we declare a class as final, we restrict the other classes to inherit or extend it. Abstract class: it is a restricted class that cannot be used to create objects (to access it, it must be inherited from another class). Abstract method: it can only be used in an abstract class, and it does not have a body. The body is provided by the subclass (inherited from). Normally, an array is a collection of similar type of elements which has contiguous memory location. Array in Java is index-based, the first element of the array is stored at the 0th index, 2nd element is stored on 1st index and so on. Unlike C/C++, we can get the length of the array using the length member. A One-Dimensional Array in Java programming is a special type of variable that can store multiple values of only a single data type such as int, float, double, char, structure, pointer, etc. at a

	contagious location in computer memory. Two – dimensional array is the simplest form of a multidimensional array. A two – dimensional array can be seen as an array of one – dimensional array for easier understanding. In Java, string is basically an object that represents sequence of char values. An array of characters works same as Java string. Java String class provides a lot of methods to perform operations on strings such as compare(), concat(), equals(), split(), length(), replace(), compareTo(), intern(), substring() etc. The java.lang.String class implements Serializable, Comparable and CharSequence interfaces. String objects immutable. The Vector class implements a growable array of objects. Vectors fall in legacy classes, but now it is fully compatible with collections. It is found in java.util package and implement the List interface. The wrapper class in Java provides the mechanism to convert primitive into object and object into primitive. Since J2SE 5.0, autoboxing and unboxing feature convert primitives into objects and objects into primitives automatically. The automatic conversion of primitive into an object is known as autoboxing and vice-versa unboxing. In the Java programming language, an interface is a reference type, similar to a class, that can contain only constants, method signatures, default methods, static methods, and nested types. An interface can extend other interfaces, just as a class subclass or extend another class. However, whereas a class can extend only one other class, an interface can extend any number of interfaces. The interface declaration includes a comma-separated list of all the interfaces that it extends. To declare a class that implements an interface, you include an implements clause in the class declaration. Your class can implement more than one interface, so the implements keyword is followed by a comma-separated list of the interfaces implemented by the class. By convention, the implements clause follows the extends clause. All the fields of an interface are static by default, you can access them using the name of the interface.
Thrust Areas	

Skills to be learnt by Students	
Examples / Illustrations	
Additional Inputs	
Teaching Aids Used	Yes
References Cited	E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.
Student Activity Planned after Teaching	Group discussion
Any other Activities	Assignment

K.R.K GOVERNMENT DEGREE COLLEGE
Department of Computer Science, Lesson Plan

Name of the Department : Computer Science	
Name of the Lecturer : D.SIVA PHANINDRA	
Group / Course : II B.Sc(comp)., Semester IV	
Paper : Object Oriented Programming Using Java	
Name of the Topic	**MULTITHREADED PROGRAMMING & MANAGING ERRORS AND EXCEPTIONS**
Hours Required	12
Learning Objectives	**MULTITHREADED PROGRAMMING:** Introduction, Creating Threads, Extending the Threads, Stopping and Blocking a Thread, Lifecycle of a Thread, Using Thread Methods, Thread Exceptions, Thread Priority, Synchronization, Implementing the 'Runnable' Interface. **MANAGING ERRORS AND EXCEPTIONS:** Types of errors : Compile-time errors, Runtime errors, Exceptions, Exception handling, Multiple Catch Statements, Using finally Statement.
Previous Knowledge to be reminded	Programming fundamentals in Java

Topic Synopsis	Multithreading in Java is a process of executing multiple threads simultaneously. A thread is a lightweight sub-process, the smallest unit of processing. Multiprocessing and multithreading, both are used to achieve multitasking.
	There are two ways to create a thread:1. By extending Thread class, 2. By implementing Runnable interface.
	In Java, a thread always exists in any one of the 5 states. These states are: New, Active, Blocked / Waiting, Timed Waiting and Terminated.
	A thread can also be temporarily suspended or blocked from entering into the runnable and subsequently running state by using either of the following thread methods:
	1. sleep(t) // blocked for 't' milliseconds.
	2. suspend() // blocked until resume() method is invoked.
	3. wait() // blocked until notify () is invoked.
	3 constants defined in Thread class:
	public static int MIN_PRIORITY
	public static int NORM_PRIORITY
	public static int MAX_PRIORITY
	Default priority of a thread is 5 (NORM_PRIORITY). The value of MIN_PRIORITY is 1 and the value of MAX_PRIORITY is 10.
	Synchronization in Java is the capability to control the access of multiple threads to any shared resource.
	
	The synchronization is mainly used to: To prevent thread interference and To prevent consistency problem.

	There are two types of synchronization: Process Synchronization and Thread Synchronization. Java runnable is an interface used to execute code on a concurrent thread. It is an interface which is implemented by any class if we want that the instances of that class should be executed by a thread. The runnable interface has an undefined method run() with void as return type, and it takes in no arguments. An error can be of 2 types. 1. A compile-time error generally refers to the errors that correspond to the semantics or syntax. 2. A runtime error refers to the error that we encounter during the code execution during runtime. In Java, an exception is an event that disrupts the normal flow of the program. It is an object which is thrown at runtime. Exception Handling is a mechanism to handle runtime errors such as ClassNotFoundException, IOException, SQLException, RemoteException, etc. Java provides five keywords that are used to handle the exception. The "try" keyword is used to specify a block where we should place an exception code. It means we can't use try block alone. The try block must be followed by either catch or finally. The "catch" block is used to handle the exception. It must be preceded by try block which means we can't use catch block alone. It can be followed by finally block later. The "finally" block is used to execute the necessary code of the program. It is executed whether an exception is handled or not. The "throw" keyword is used to throw an exception. The "throws" keyword is used to declare exceptions. It specifies that there may occur an exception in the method. It doesn't throw an exception. It is always used with method signature.
Thrust Areas	Programming Design
Skills to be learnt by Students	SQL Commands
Examples / Illustrations	SQL Queries on Company and banking databse
Additional Inputs	
Teaching Aids Used	Yes
References Cited	DBMS by Korth, DBMS by Navathe, DBMS by Raghu Rama Krishnan
Student Activity Planned after Teaching	Group discussion
Any other Activities	Assignment

K.R.K GOVERNMENT DEGREE COLLEGE
Department of Computer Science, Lesson Plan

Name of the Department : Computer Science	
Name of the Lecturer : D.SIVA PHANINDRA	
Group / Course : II B.Sc(comp)., Semester I	
Paper : Object Oriented Programming Using Java	
Name of the Topic	**Applet Programming, Packages and managing I/O in Java**
Hours Required	12
Learning Objectives	**APPLET PROGRAMMING:** local and remote applets, Applets and Applications, Building Applet code, Applet Life cycle: Initialization state, Running state, Idle or stopped state, Dead state, Display state. **PACKAGES:** Introduction, Java API Packages, Using System Packages, Naming conventions, Creating Packages, Accessing a Package, using a Package. **MANAGING INPUT/OUTPUT FILES IN JAVA:** Introduction, Concept of Streams, Stream classes, Byte Stream Classes, Input Stream Classes, Output Stream Classes, Character Stream classes: Reader stream classes, Writer Stream classes, Using Streams, Reading and writing files.
Previous Knowledge to be reminded	Java Programming Fundamentals
Topic Synopsis	A special type of Java program that runs in a Web browser is referred to as Applet. There are two types of applets that a web page can contain. Local Applet: Local Applet is developed locally and stored in the local system. Remote Applet: It is located or available on a remote computer that is connected to the internet. In order to run the applet stored in the remote computer, our system is connected to the internet then we can download run it. Applet is a small java program that can be run by a Java-compatible web browser while application is a standalone program that can directly run on the machine. Torun the applet, Create an HTML page that references the applet. Open the HTML page in a browser. The applet life cycle can be defined as the process of how the object is created, started, stopped, and destroyed during the entire execution of its application. It basically has five core methods namely init(), start(), stop(), paint() and destroy().These methods are invoked by the browser to execute. A java package is a group of similar types of classes, interfaces and sub-packages. Package in java can be categorized in two form, built-in package and user-defined package. The package keyword is used to create a package in java.

	In Java, streams are the sequence of data that are read from the source and written to the destination. An input stream is used to read data from the source. And, an output stream is used to write data to the destination. The Stream class defines objects which accept a sequence of characters. Java encapsulates Stream under java.io package. Java defines two types of streams. They are, Byte Stream : It provides a convenient means for handling input and output of byte. Character Stream : It provides a convenient means for handling input and output of characters. Character stream uses Unicode and therefore can be internationalized. Byte stream is defined by using two abstract class at the top of hierarchy, they are InputStream and OutputStream. Character stream is also defined by using two abstract class at the top of hierarchy, they are Reader and Writer.
Thrust Areas	
Skills to be learnt by Students	
Examples / Illustrations	
Additional Inputs	
Teaching Aids Used	Yes
References Cited	E.Balaguruswamy, Programming with JAVA, A primer, 3e, TATA McGraw-Hill Company.
Student Activity Planned after Teaching	Group discussion
Any other Activities	Assignment