

GUJARAT TECHNOLOGICAL UNIVERSITY

Master in Computer Application (Integrated MCA)

Year II – (Semester-III) (W.E.F. July 2014)

Subject Name: Java Programming

Subject Code: 4430601

1. Learning Objectives:

- To develop proficiency in creating console based and GUI based applications using the Java Programming Language.
- To interpret the concepts of Object Oriented Programming Language and easily use Java.
- To implement multi-threaded applications using the Java Programming Language.
- To implement Applets for embedding in a web page.
- To create GUI based applications
- To implement Applets for embedding in a web page

2. Prerequisites: Knowledge of the C programming language.

2. Contents:

Unit No.	Course Content	No of Lectures
1	Object Oriented Programming and Introduction to Java, Data types, operators, statements and defining classes in Java: <ul style="list-style-type: none">o Features of the Java Language,o Object-oriented Programmingo Creating an Application in Javao Compiling and executing Applications in Javao Program commentso Primitive data typeso Integer Data Types, Floating Point Data Typeso Reference Data typeso Arrays, single and multi-dimensional arrayso Other reference types, classes, interfaces, enums and annotationso Unicode escapes in Java source codeo Understanding supertypes and subtypes.	8

	<ul style="list-style-type: none"> o Operators - Arithmetic, String concatenation, Relational, Logical, Bitwise, increment-decrement, conditional, assignment, cast and instanceof operators. o Understanding the narrowing and widening conversions of numeric data types. o Statements - if, if-else, switch-case, for, while, do-while, break, continue and return statements. o Various members within a class o instance variables o methods and their overloading o constructors and their overloading o Garbage collector and finalize method o static variables and methods o Initializer blocks and the class initialize blocks 	
2	<p>Inheritance and subclassing in Java, packages and use of access specifiers, using common classes from the java.lang package</p> <ul style="list-style-type: none"> o Defining subclasses o Using super to use constructor of a super-class o Method overriding and use of super o Variable shadowing and use of super. o Method and variable binding o Using final with variables, methods and classes o Abstract classes and interfaces o Abstract classes and abstract methods o Single inheritance of classes o Interfaces o Object class as the super class of all classes o Methods inherited from the Object class o Uses of package and import statements o use of static imports o use of CLASSPATH for class loading o Access specifiers o Access specifiers for members of a package o Access specifiers for members of a class o Access specifiers for overriding methods o Using the Java APIs o Commonly used classes from the java.lang package o Comparable and Comparator interfaces o String, StringBuffer and the StringBuilder classes o Understanding pass by value and pass by reference for Java o Wrapper classes o Math class constants and methods 	8
3	<p>Exceptions, Nested enum types, Generic Programming and Collection framework:</p> <ul style="list-style-type: none"> o Runtime stack and execution of application o The return and the throw statements o The return type and throws declaration in methods o Checked and the Unchecked exception classes o The Throwable class o Exception chaining o Handling exceptions with try and catch o Use of the finally block o Creating custom exception classes o Member Types o Top level nested classes and Inner classes o The local class and anonymous classes 	10

	<ul style="list-style-type: none"> o The enum type o Why Generic Programming? o Generic Class o Generic Methods o Generic Code and the Virtual Machine o Restrictions and Limitations o Inheritance Rules for Generic Types o WildCard Types o classes from java.util package o Date, TimeZone, Calendar and the GregorianCalendar classes o Collection Framework o Collection interface o Set and List interfaces o Map Interface o Generics in the Collection Framework o Regular Expressions, Pattern and Matcher classes o Scanner class o Varargs and the Formatter class 	
4	<p>Stream based I/O and Multi-threading:</p> <ul style="list-style-type: none"> o Stream classes o OutputStream and the Writer classes o InputStream and the Reader classes o Bridge classes OutputStreamWriter and the InputStreamReader o Writing and reading from files using FileOutputStream and the FileInputStream o Piped Streams o Array based streams o Filter streams o Buffered streams o PrintStream and the PrintWriter classes o Data and Object streams o RandomAccessFile o Multi-threading o Thread class and thread of execution o Creating a new Thread of execution o ThreadGroup o properties of Thread instance o Daemon Threads o Thread states o Synchronization o Another way of creating a thread of execution 	10
5	<p>GUI Programming, Building Applets and Introduction to Annotations:</p> <ul style="list-style-type: none"> o Comparing AWT and swing features o AWT Components o Overview of the AWT components o Component properties o Graphics context o Containers o Container class o Layout Managers o Top-level containers o Window class o Decorated windows Frame and Dialog 	12

	<ul style="list-style-type: none"> o Panel class o Events o Event Delegation Model o AWTEvents o Adapter classes o Swing and MVC o JComponent o JOptionPane o Showing Message, Confirm and Input Dialogs o Browser as a container for Applets o Life-cycle of an Applet o AppletStub and <applet> tag o AppletContext and the URL class. o Applet class o Defining a new annotation o Annotating a programming element o Annotation element names and usage o meta-annotations 	
--	--	--

Notes:

Term work is to be carried out as per the above syllabus.

4. Reference Book(s):

- 1) Pravin Jain, “The class of Java” Pearson Education, (2010).
- 2) Cay S Horstmann, Gary Cornell, “Core Java 2, Volume 1 – Fundamentals”, Pearson Education(8th edition – 2008).

5. Suggested Additional Reading:

- 1) Ivor Horton's “Beginning Java 2” JDK 5 Edition, Wiley Computer Publishing, (2007).
- 2) Ken Arnold, James Gosling, David Holmes, “The Java Programming Language”, Addison-Wesley Pearson Education (4th Edition – 2005).
- 3) Raj Kumar Buyya, S. Thamarai Selvi, & Xing Chen Chu, “Object-Oriented Programming with Java: Essentials & Applications”, Tata McGraw Hill
- 4) Cay Horstmann, “Big Java”, Wiley Computer Publishing (2nd edition – 2006).
- 5) Hari Mohan Pandey, “Java Programming”, Pearson
- 6) Sharan Zakhour, Scott Hommel, Jacob Royal, Isaac Rabinovitch, Tom Risser, Mark Hoeber “The Java Tutorial”, Addison-Wesley Pearson Education(4th Edition), available for online reference at <http://java.sun.com/docs/books/tutorial/index.html>
- 7) James Gosling, Bill Joy, Guy Steele, Gilad Bracha, “The Java Language Specifications”, Addison-Wesley Pearson Education(3rd edition), available for download at <http://java.sun.com/docs/books/jls/download/langspec-3.0.pdf>

6. Chapter wise Coverage from Main Reference Book(s):

1	Book #1 ->	Ch. 1, Ch. 2 Except for section on Various tools in JDK and Using java doc, Ch. 3, Ch. 4, Ch. 5
2	Book #1 ->	Ch. 6, Ch. 7, Ch. 8, Ch. 9 Except for sections on generating javadoc, commonly used packages from Java APIs, Ch. 10 Except for sections on Unicode characters and using supplementary characters and Numeric values requiring more than 64 bits
3	Book #1 ->	Ch. 11 Except for section on assertions, Ch. 12, Ch. 13 Except for sections on Arrays class, Collections class and StringTokenizer class
4	Book #1 ->	Ch. 14 Except for sections on Pushback streams and SequenceInputStream, Ch. 16
5	Book #1 ->	Ch. 17 Except for section on Toolkit, Ch. 18 Except for section on Scroll Pane class, Ch. 19, Ch. 20 Except for sections on JlayeredPane class and MVC and using the models in JTable and JTree., Ch. 21, Ch. 23
6	Book #2 ->	Ch. 12

7. Accomplishments of the student after completing the Course:

- Ability to create appropriate classes using the Java Programming Language to solve a problem using Object Oriented Approach.
- Ability to write console based and GUI based applications in the Java Programming Language.
- Ability to develop to multi-threaded applications using the Java Programming Language
- Ability to create Applets using the Java Programming Language