

OBJECT ORIENTED PROGRAMMING

COURSE OBJECTIVE :

This syllabus describes about object oriented programming through Java. It describes the syntax, key words and fundamental programming principles. Significant Java API libraries are also examined.

JNTU SYLLABUS :

UNIT I:

Object oriented thinking :- Need for oop paradigm, A way of viewing world – Agents, responsibility, messages, methods, classes and instances, class hierarchies (Inheritance), method binding, overriding and exceptions, summary of oop concepts, coping with complexity, abstraction mechanisms.

UNIT II:

Java Basics :History of Java, Java buzzwords, datatypes, variables, scope and life time of variables, arrays, operators, expressions, control statements, type conversion and casting, simple java program, classes and objects – concepts of classes, objects, constructors, methods, access control, this keyword, garbage collection, overloading methods and constructors, parameter passing, recursion, nested and inner class, exploring string class.

UNIT III:

Inheritance – Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance. Member access rules, super uses, using final with inheritance, polymorphism- method overriding, abstract classes,the object classes.

UNIT IV :

Packages and Interfaces : Defining, Creating and Accessing a Package, Understanding CLASSPATH, importing packages, differences between classes and interfaces, defining an interface, implementing interface, applying interfaces, variables in interface and extending interfaces. Exploring packages – Java.io

UNIT V :

Exception handling - Concepts of exception handling, benefits of exception handling, Termination or resumptive models, exception hierarchy, usage of try, catch, throw, throws and finally, built in exceptions, creating own exception sub classes. String handling, Exploring java.util

UNITVI :

Multithreading Differences between multi threading and multitasking, thread life cycle, creating threads, thread priorities,synchronizing threads, inter thread communication, thread groups, daemon threads.Enumerations,autoboxing,annotations,generics

UNIT VII:

Event Handling : Events, Event sources, Event classes, Event Listeners, Delegation event model, handling mouse and keyboard events, Adapter classes.

The AWT class hierarchy, user interface components- labels, button, canvas, scrollbars, text components, check box, check box groups, choices, lists panels – scrollpane, dialog, menubar, graphics, layout manager – layout manager types – boarder, grid, flow, card and grid bag.

UNIT VIII:

Applets – Concepts of Applets, differences between applets and applications, life cycle of an applet, types of applets, creating applets, passing parameters to applets.

Swing – Introduction, limitations of AWT, MVC architecture, components, containers, exploring swing- JApplet, JFrame and JComponent, Icons and Labels, text fields, buttons – The JButton class, Check boxes, Radio buttons, Combo boxes, Tabbed Panes, Scroll Panes, Trees, and Tables.

TEXT BOOKS :

1. Herbert schildt , “*Java; the complete reference*”, 7th edition, , TMH.
2. T. Budd, “*Understanding OOP with Java*”, updated edition, pearson education.

REFERENCES :

1. J.Nino and F.A. Hosch “*An Introduction to programming and OO design using Java*”, John wiley & sons.
2. T. Budd, “*An Introduction to OOP, second edition*”, pearson education.
3. Y. Daniel Liang “*Introduction to Java programming*” 6th edition, , pearson education.
4. R.A. Johnson “*An introduction to Java programming and object oriented application development*”, - Thomson.
5. Cay.S.Horstmann and Gary “*Core Java 2*”, Vol 1, Fundamentals, Cornell, seventh Edition, Pearson Education.
- 6 . Cay.S.Horstmann and Gary Cornell “*Core Java 2*”, Vol 2, Advanced Features, , Seventh Edition, Pearson Education
7. P. Radha Krishna “*Object Oriented Programming through Java*”, University Press.
8. John hunt, “Java and Object orientation, an introduction” 2 nd edition Springer.
9. Maurach, “Beginning Java2”, JDK 5, SPD
10. JM Slack, “Programming and Problem Solving with java”, B.S.Publications

SESSION PLAN:

Topics in each unit as per JNTU syllabus	Lecture No.	Modules/sub modules for each Topic	Text Books/ Reference Books
UNIT-I			
	1	Course objective and scope, Differences between structured and object oriented programming, Explain the paradigm to understand the oops concepts and its features	T2: pgno:3
Agents, responsibility, messages, methods ,Classes and Instances, Class Hierarchies	2	Classes, Responsibilities, Objects, Inheritance, Messages ,Methods	T2: 1.1.1-1.1.5
Method binding, Over riding and Exceptions	3	Method binding ,Over riding , Exceptions	T2: 1.1.6
Summary of OOP concepts, copying with complexity Abstraction Mechanisms	4	Summary of OOP concepts, Abstraction Mechanisms, Copying with complexity	T2: 1.1.7
	5	Tutorial	
	6	Revision: Unit-I	
UNIT-II			
History of Java, Java Buzzwords, Data types, Variables, scope and lifetime of variables	7	Creation of java, Byte code, java is simple, object oriented, robust, multithreaded, distributed , dynamic, primitive types, floating point types , characters , Booleans, Declaring & creation of variables	T1: 6-13,33-38 T1:41-42 R1: 2.1-2.2 R3: pgno:2-8,32-33, 35-39 R5: 4-11,48-52 R1:14.7 R3:pgno:29-31 R5:52-53
Arrays, Operators, Expressions ,control statements	8	One dimensional, multi dimensional, arrays,Arithmetic, Bitwise, relational, logical, assignment operators, operator precedence, control statements	T1: 48-55,57-74 R1: 6.1-6.2 R3: pgno:40,52-67 R5: 92-98,56-59, 74-84
Automatic type conversion, Casting Incompatible types, simple programs	9	Automatic type conversion,Casting ,incompatible types,simple programs	T1: 45-47 R3: pgno:35
Concepts of classes, class fundamentals, Declaring objects, assigning object reference values	10	General form of class & object, A simple Class	T1: 105-111 R1: 2.3-2.4 R3: pgno:110-113 R5:pgno: 109-111
Constructors, Methods, Access Control	11	Parameterized constructors, adding methods to the class, access controls	T1: 117-120,112-138 R1:4.2.3 R3:pgno:78-80,114,115
this keyword, garbage collection, Overloading methods, overloading constructors Call by value.	12	this keyword , garbage collection, Overloading methods, overloading constructors Call by value ,	T1: 120,121,125-130 R1:10.4.2,23.5 R1:Pgno:342-344 R3:pgno:83-85,187 R5:141-142,139-140
Passing objects, Recursion, Nested class, inner class, Exploring String	13	Passing objects, Recursion,	T1:pgno:132-134,134-135,135-137

class		Nested class, inner class, Exploring String class	R3:pgno:81-83,166-172 R5:90-92,60-67
	14	Tutorial	
	15	Revision: Unit-II	
Unit-III			
Hierarchical abstractions, Base class object, subclass, subtype, substitutability, forms of inheritance-specialization, specification,	16	Hierarchical abstractions, Base class object, Subclass, Subtype, Substitutability, forms of inheritance-specialization, specification	T1: 6,157-170 R1:14.5 R3:pgno:182,183 R5:164-170
Construction, extension, limitation, combination, benefits of inheritance, costs of inheritance	17	Construction, extension, limitation, combination, benefits of inheritance, costs of inheritance	R3: 7
Member access rules, super uses, using final with inheritance	18	Member access rules, Super uses, using final to prevent overriding, To prevent Inheritance	T1: pgno:159,163-167,180-181 R1:14.2.3 R3: pgno:184-187, 189-190 R5:165,172-173
Polymorphism-method overriding, abstract classes the object class	19	Dynamic Method Dispatch, method overriding, abstract classes, object class	T1: pgno:171-177, 177-181 R1:14.2.2,14.3 R3: pgno:189-190 R5:176
	20	Tutorial	
	21	Revision: Unit-III	
Unit IV			
Defining a Package, Creating and Accessing the package, understanding CLASSPATH, access protection	22	Defining a Package, Creating and Accessing the package, understanding CLASSPATH, access protection example	T1: pgno:183-190 R3: pgno:133-138 R5:153-255
How to import a package, Differences between classes and interfaces	23	How to import a package, Differences between classes and interfaces	T1:pgno:190-192 R1:pgno:174 R5:155-157
Defining an interface, Access implementations and Partial implementations of an interface	24	Defining an interface, Access implementations and Partial implementations of an interface	T1:pgno:193 T1: pgno:195-196 R1:15.1.2 R3: pgno:210-215 R5:226 R5:229
Applying interfaces, Variables in interfaces and extending interfaces	25	Applying interfaces, Variables in interfaces and extending interfaces	T1:pgno:197-203
Exploring Java. IO package	26	Java i/o classes&interfaces, stream classes byte stream, character streams,console class, serialization,stream benefits	T1: pgno:555-598
	27	Tutorial	
	28	Revision: Unit-IV	
Unit V			
Concepts of exception Handling, benefits of exception handling	29	Way to handle and invoke exceptions and its uses	T1: pgno: 205-206 R1:pgno: 444 R3: pgno:364-366 R5: pgno: 583-585
Termination or resumptive models	30	Termination or resumptive models	T1: pgno: 207

Exception hierarchy, usage of try, catch, throw, throws and finally	31	Exception hierarchy, usage of try, catch, throw, throws and finally	T1: pgno: 221, 207-216 R1:446-449 R3:pgno:366-380 R5:586-587,592-597
Built in exceptions, creating own exception sub classes	32	Run time Exceptions, Checked Exceptions, Unchecked Exceptions, Creating own exceptions by using Exception class	T1: 217-219 R1: 18.2.4
String Handling	33	String constructors,special string operations,character extraction,modifying string,string buffer	T1:359-384
Exploring java util	34-35	String tokenizer,bit set, calendar,formatter, java.util.sub packages	T1:pgno:503-554
	36	Tutorial	
	37	Revision: Unit-V	
Unit-VI			
Multi threading	38-39	Thread model,ImplementingRunnable, Extending thread,creating Multiple Threads, synchronized methods ,synchronized statement ,Inter Thread Communication, Deadlock	T1:Pgno:224-237 R3:pgno:394-403 T1:pgno:238-253 R3:pgno:407-411
Daemon threads, thread groups	40	Daemon thread,Thread Group	T1:pgno:424-429 R3:pgno:406-407
Enumerations,Auto boxing,annotations,generics	41	Enumerations,Auto boxing,annotations,generics	T1:Pgno:255-284
	42	Tutorial	
	43	Revision: Unit-V	
Unit-VII			
Events, Event Sources, Event Classes, Event listeners	44	Two event handling mechanisms, Declaring event model, Event sources, Event Classes, Event listeners	T1: pgno: 637-648 R3:pgno:227-230
delegation event model, handling mouse events	45	delegation event model, handling mouse events	T1: pgno 653-656 R3:pgno:233-235
Handling key board events	46	Handling key board events	T1: pgno 656-659 R3:pgno:337-339
Adapter classes	47	Adapter classes	T1: pgno 659 R1:328-330
Inner classes	48	Anonymous Inner classes	T1: pgno 662 R5:238-255
AWT class hierarchy	49	Window fundamentals,creating a applet, Handling events,Working with graphics	T1: pgno :666-683 R3:pgno:314-320 R5:pgno:330-332
User interface components-labels, buttons, canvas, scroll bars, text components, check box	50	User interface components-canvas, scroll bars, text components, check box	T1: pgno: 667,702-708 R3: pgno:246-248, 270-278
Check box groups, choices, list panels	51	Choice controls,handling choice lists, handling lists	T1: 709-714 R3:pgno:289-292,278-286
Scrollpane, dialogs, menubar, graphics	52	Scrollpane, dialogs, menubar, graphics	T1: 716-718,737-749 R3:pgno:293-303
Layout manager, layout manager types, boarder and grid layouts	53	Understanding Layout managers layout manager types, boarder and grid	T1:723,725,728-729 R1: pgno:479-481

		layouts	R3:pgno:239-242 R5:383,465-468
Flow Layout, Card Layout and Grid bag	54	Flow Layout, Grid bag, Card Layout	T1: pgno: 724,732,730 R1: pgno:478,580 R3:pgno:238-239 R5: pgno: 474-475
	55	Tutorial	
	56	Revision: Unit-VI	
Unit VIII			
Concept of Applets difference between Applets and applications, Life cycle of Applets	58	Applet fundamentals, Applet class, Applet architecture, Applet initialization and termination	T1:pgno:296-299, 618-628 R1: pgno: 669-670 R3:pgno:314-319, 323-331 R5: pgno: 532-547
Types of Applets, Creating Applets, passing parameters to applets	59	HTML applet tag, passing parameters to applets	T1: pgno:617,629-636 R1: 3.5-3.7 R3:pgno:320-323 R5: pgno: 548-559
Introduction, limitations of AWT, MVC Architecture	60	Describing what is the use of Abstract Window Toolkit, Model-view-controller connection	T1: 24 T1: pgno:861 R1: pgno: 20.1-20.2 R5:374-379
Containers, exploring swing JApplet, JComponent, Icons and Labels	61	Creating a swing Applet, Painting fundamentals, Creating Icons and Labels	T1: pgno:863,871-879 T1: pgno:864-866 R1: 19.5.6 R1:19.5.2 T1: pgno:874-878 R1: 19.5.5 T1: pgno:879-881 R5:pgno:404-407
Text fields, buttons, JButton class, Checkboxes	62	Text fields, buttons, JButton class, Checkboxes	T1: pgno:881-888 R1: R5:pgno:387-393,408-411
Radio buttons, comboboxes, Tabbed panes, scroll panes, Trees and Tables	63	Designing Radio buttons, comboboxes, Designing Tabbed panes scroll panes, Designing Trees and Tables	T1:pgno:889,898-900,900-906 R5:411-415,438-441 T1: pgno:891-895 R5: pgno: 446-455
	64(T)	Tutorial	
	65	Revision: Unit-VII	

Referred to by faculty:-

- T1.** Herbert schildt , “*Java; the complete reference*”, 7th edition, , TMH.
T2. T. Budd, “*Understanding OOP with Java*”, updated edition, pearson education.
R1. J.Nino and F.A. Hosch “*An Introduction to programming and OO design using Java*”, John wiley & sons.
R3. Y. Daniel Liang “*Introduction to Java programming*” 6th edition, , pearson education.
R5. Cay.S.Horstmann and Gary “*Core Java 2*”, Vol 1, Fundamentals, Cornell, seventh Edition, Pearson Education.

WEBSITES

1. <http://java.sun.com/docs/books/tutorial/java/concepts/index.html>-- Introduction to object oriented concepts
2. <http://java.sun.com/docs/books/tutorial/java/javaOO/index.html>-- objects and classes
3. <http://java.sun.com/docs/books/tutorial/java/concepts/inheritance.html>--Inheritance
4. <http://java.sun.com/docs/books/tutorial/java/package/packages.html>-- packages
5. <http://java.sun.com/docs/books/tutorial/essential/exceptions/>--Exception handling
6. <http://java.sun.com/docs/books/tutorial/essential/concurrency/>--Multithreading
7. <http://www.javaworld.com/javaworld/jw-07-1996/jw-07-awt.html>--AWT
8. <http://java.sun.com/docs/books/tutorial/networking/overview/index.html>--Networking

JOURNALS

1. Dr.Dobb's Journal-- Software tools and techniques for global software development. **Dr. Dobb's** features articles, source code, blogs,forums,video tutorials, and audio podcasts
2. Journal of Object Oriented Programming--- An object is actually a discrete bundle of functions and procedures, all relating to a particular real-world concept such as a bank account holder or hockey player in a computer game. Other pieces of software can access the object only by calling its functions and procedures that have been allowed to be called by outsiders. Isolating objects in this way makes their software easy to manage and keep track of.
3. Journal of Object Technology-- **Object Technology** Solutions Inc. (OTSI) is a leading Global Information Technology (IT) Services and Solutions company offering a wide array of Solutions.
4. Java Pro--- In this article, an excerpt from **Pro Java** Programming (Apress, June 2005), Brett Spell explains step-by-step how to locate print services, create a print.
5. Java World-- Search the **JavaWorld** Archives for older columns and series that have stood the test of time, especially for those just learning about **Java** technology.

STUDENT SEMINAR TOPICS

1. Title: Developing in Java5
Journal: Java Developer's Journal, Dec-2005, Pg No:44 to 47
Author: Roberto Scaramuzzi
2. Title: Table Layout
Journal: Java Developer's Journal,Dec-2005, Pg No: 54 to 58
Author: Phil Herold
3. Title: Signaling Integer overflows in Java
Journal: Dr.Dobb's Journal, Pg No: 54 to 58
Author: Frederic and Francis
4. Title: Building a toolbar from a menu
Journal: Java Developer's Journal,Oct-2005, Pg No: 54 to 57
Author: Mauro Micalizzi

5. Title: ArrayList Model
Journal: Java Developer's Journal, Oct-2005, Pg No: 50 to 52
Author: Phil Herold
6. Title: Disentangling concepts in object oriented systems
Journal: Dr.Dobb's Journal, Pg No: 46 to 49
Author: Stephen

GUEST LECTURE TOPICS:

1. Remote Method Invocation
2. Web services
3. Entity Java beans
4. Networking

ASSIGNMENT QUESTIONS

Unit-I

1. What is the difference between structured programming and object oriented programming?
2. What do you mean by paradigm. what is the use of it?
3. Take the paradigm "Sending Flowers to the friend saying abroad" and explain what is the agents and communities in that?
4. What do you mean by Responsibilities?
5. What do you mean by messages and methods in oops?
6. Describe inheritance using the paradigm?
7. Define a class and object ?
8. What do you mean by exception explain using the paradigm?
9. Explain method overriding using the paradigm?
10. Explain method binding using the paradigm?
11. What do you mean late binding? is it mean run time polymorphism?
12. Explain different abstraction mechanisms?
13. Mention the oops concepts summary?
14. What do you mean copying with complexity?
15. Take any paradigm and explain all the features of oops?

Unit –II

1. Why we wont prefer C and C++ for internet applications? What is importance of java to internet Explain?
2. Explain Java Buzzwords
3. Explain encapsulation, inheritance, polymorphism
4. Explain the datatypes in Java
5. Explain about the scope and life time of variable
6. Explain how to declare variables and how is initialize it dynamically
7. Explain arrays with an example
8. Explain the various operators in Java?
9. Explain different control statements in java?
10. Explain type conversion and casting?
11. Explain Class with an example?
12. What is an object how do u declare it

13. Explain what is a constructor with an example
14. Explain access control methods
15. Explain about this keyword? example
16. Explain garbage collection in java?
17. Explain the usage of static with data and methods
18. Explain overloading methods with an example
19. Explain about different parameter passing methods?
20. Explain the concept of recursion and string handling functions?

UNIT -III

1. Explain about inheritance concept?
2. Explain about Object class
3. Explain about subclass and sub type?
4. Explain the various forms of inheritance
5. What are the benefits of inheritance?
6. Explain about cost calculation of inheritance
7. Explain the member access rules
8. Explain the usage of super key word
9. Explain the use of final with inheritance
10. what is the meaning if we use finally with variables
11. Explain dynamic method dispatch with an example
12. Explain method overriding with an example
13. Explain the difference between overloading and overriding
14. Explain the concept of abstract classes with an example
15. Explain the differences between class and abstract class

UNIT -IV

1. Define package
2. How to create and access a package
3. Depending on access control how the methods in the package can be accessed
4. Explain about CLASSPATH
5. Explain about importing the packages
6. Explain the differences between classes and interface
7. What is an interface Explain with an example
8. Explain how to implement interface
9. Explain about applying interfaces
10. Explain how to declare variables in interfaces
11. Explain how to extend interfaces
12. Explain the difference between package and interface?
13. Explain about byte stream, character streams,
14. Explain console class, serialization
15. What are the benefits of stream

Unit V

1. What is exception handling and explain types of exceptions
2. What are benefits of Handling exceptions in programming

3. Explain about exception hierarchy
4. Write a program using try and catch
5. Write a program using throw, throws and finally key words
6. Explain about built in exception
7. How to create our own exceptions in java
8. Explain Termination models.
9. Write about String comparison methods.
10. Explain Data conversion using valueOf method.
11. Write String Buffer class and its methods.
12. Explain about accessing a collection via Iterator.
13. Explain about working with maps.
14. Explain the use of generic collection.
15. Give the collection of classes in java.util

UNIT-VI

1. Write notes on multi threading
2. Write notes on multi Tasking
3. Explain about thread life cycle and draw the diagram
4. Write the difference between process and thread
5. How to create multiple threads using thread class
6. Write notes on
 - a) Synchronization
 - b) Thread Priorities
 - c) Inter thread communication
7. Write notes on
 - a) Daemon Threads
 - b) Dead locks
 - c) Thread groups
8. What are daemon threads? Write a program to illustrate thread synchronization in java?
9. Explain about thread priorities with an example program.
10. Explain how enumeration inherit enum.
11. Write a short notes on
 - a) Annotation basics.
 - b) Using default vales
12. Explain with an example how autoboxing /unboxing occurs in expressions.
13. Write about annotations at run time by use of reflection.
14. Write a short notes on built-in annotations.
15. Give an example to demonstrate the use of value() and valueof() methods.

Unit VII

1. What is an event? Explain event classes
2. Write notes on
 - a) event sources
 - b) event classes
 - c) event listeners
3. Write a program using events sources , event classes
4. Write the difference between event sources , event classes, event listeners
5. Explain about Delegation Event model?

6. Write a program to handle keyboard and mouse events?
7. Write notes on adapter classes
8. Explain about anonymous and inner class
9. Draw the AWT hierarchy
10. What is AWT and explain AWT components
11. Write about labels and buttons in AwT
12. Describe about Choice and list using a program
13. Write about Check box groups
14. Explain about menus
15. Explain what is container, panel, window
16. Write notes on frames, canvas
17. Explain about layout manager and what is the use of it
18. Explain boarder and grid layout
19. Explain card and Grid bag layout
20. Write a program using events

Unit VIII

1. Create an applet and explain life cycle of an applet?
2. Write the difference between applets and applications and mention the types of applets?
3. How you pass parameters to applets
4. What is AWT & explain AWT controls?
5. Explain Model-view-controller architecture
6. Write a program JComponent fields?
7. Explain about components and containers
8. What is swing & write program using swings?
9. Explain JApplet , JFrame & JComponent,
10. Expalin about Icons and labels?
- 11 Write notes on buttons?
12. Explain about check boxes
13. Explain about Radio buttons and combo boxes
14. What is Tabbed Panes, Scroll panes,
15. Explain about trees with an example
16. Describe about tables with an example

Unit VIII

1. What is networking & write notes on basics of networking?
2. Explain Inet address methods?
3. Explain Inet 4 address and Inet 6 address?
4. What is URL and give the URL format& URL connection?
5. Write notes on TCP/IP sockets?
6. Write notes on java library (string handling)?
7. Write about multiple clients
8. Write notes on Java I/o classes and interfaces?
9. Write notes on Closeable and flushable interfaces?
 - c) Stream classes?
10. Write a client/server application using datagram?
11. Write about String tokenizer
12. What is collection Frame work and write notes on collection interfaces and classes?

13. Explain collection Algorithms?
14. Write about Date class with an example
15. Write about Calendar, Time zone class

QUESTION BANK

UNIT-I

1. a) Explain importance of Java?
b) Explain encapsulation, inheritance, polymorphism and data types? (JNTU FEB 2010)
2. Explain various control structures and lexical issues in Java. (JNTU FEB 2010)
3. Discuss about.
a) Need for OOP paradigm.
b) Methods, classes and instances
c) Summary of OOP concepts. (JNTU FEB 2010)
3. Discuss about
a) Summary of OOP concepts
b) Overriding and exceptions
c) Abstraction mechanisms (JNTU FEB 2010)
4. a) Why is java more secured than other languages.
b) Write short notes on abstraction mechanism, and exceptions.
c) Write a program to find factorial of a given number? (JNTU FEB 2010)
5. Distinguish the following terms.
(a) Objects and classes.
(b) Data abstraction and Data encapsulation. (JNTU Aug/Sep 2009)
6. Explain different abstraction mechanisms possible in object oriented programming. (JNTU Aug/Sep 2009)
7. Distinguish the following terms.
(a) Inheritance and Polymorphism.
(b) Dynamic binding and message passing. (JNTU Aug/Sep 2009)
8. If one is familiar with two or more distinct computer programming languages, give an example of showing how one language would direct the programmer to one type of solution, and a different language would encourage an alternative solution. (JNTU Aug/Sep 2009)
9. (a) Describe the structure of a typical Java Program.
(b) Enumerate the rule for creating identifiers in Java (JNTU Aug/Sep 2008)
10. (a)What is data abstraction? What is information hiding? Contrast them.
(b)How event driven programming is different from Procedure oriented programming. (JNTU Nov 2006)
11. Describe areas of application of Object Oriented technology. (JNTU March 2006)
12. List out any five major C++ features that were intentionally removed from Java. (JNTU March 2006)
13. How data and methods organized in an object oriented program? (JNTU Mar 2006)
14. What are the drawbacks of structured procedural languages? Explain how Java solved

- these problems. (JNTU Nov/Dec 2005)
15. Write any five applications of OOP technology. (JNTU Nov/Dec 2005)
16. What are unique advantages of an object oriented paradigm?
(JNTU Nov/Dec 2005)
17. Distinguish between structured oriented Programming and Objected Oriented Programming. (JNTU Nov/Dec 2005)
18. Discuss about procedure oriented and object oriented programming. Give examples. (JNTU Nov/Dec 2005/Nov 04)
19. What are the drawbacks of structured procedural languages? Explain how Java solved these problems. (JNTU Nov 2005)
20. Explain the benefits of Objected Oriented Programming. How does Java differ from C language? (JNTU Nov 2005)
21. Write short notes on the following:
Data abstraction
Data binding
(JNTU Nov 2005)

UNIT-II

1. a) Explain java buzzwords?
b) Explain how is java more secured than other languages? [8+8] (JNTU FEB 2010)
2. a) Explain constructors and constructor overloading.
b) Write an example program for constructor and method of over loading?
(JNTU FEB 2010)
- 3 a) Explain all scope and life time variables, arrays, operators and control statements in Java?
b) Write syntax for compilation and running of a simple java program? (JNTU FEB 2010)
4. a) Write a short notes on exploring the string class with examples?
b) Explain about nested and inner classes? (JNTU FEB 2010)
5. Explain the object classes and its method? (JNTU FEB 2010)
6. Explain the following ones with example.
a) Usage of static with data and methods.
b) Final keyword with data, access controls.
c) This keyword. (JNTU FEB 2010)
7. a) Explain all scope and life time variables, arrays, operators and control statements in Java?
b) Write syntax for compilation and running of a simple java program? (JNTU FEB 2010)
8. Write short notes on the following:
a) Inheritance
b) Static class variables
c) Package
d) Constructors (JNTU FEB 2010)
9. a) What is an array? Why arrays are easier to use compared to a bunch of related variables?
b) Write a program for transposition of a matrix using array copy command?
(JNTU FEB 2010)
10. What does toString() method returns? How is it handled in Java? Explain with an example. (JNTU Aug/Sep 2009)
11. How many Data types are in java? Explain with ranges. (JNTU Aug/Sep 2009)

12. (a) Write a Java program that implements Linear Search?
 (b) What does System.out.println in Java? (JNTU Aug/Sep 2009)
13. (a) List the eight data types used in Java. Give examples.
 (b) Write a while loop to find the smallest n such that n^2 is greater than 10,000.
 (JNTU Aug/Sep 2008)
14. (a) What is statement? How do the Java statements differ from those of C and C++?
 (b) Given a number, write a program using while loop that reverses the digits of the number.
 (JNTU Aug/Sep 2008)
15. Write a program that will read an unspecified number of integers and will determine how many positive and negative values have been read. Your program ends when the input is 0.
 (JNTU Apr/May 2008)
16. Write a program to convert the given temperature in Fahrenheit to Celsius using the following conversion formula $C = (F - 32)/1.8$ And display the values in a tabular form.
 (JNTU Apr/May 2008)
17. (a) What is class? How does it accomplish data hiding?
 (b) How do classes help us to organize our programs?
 (c) Compare and contrast overloading and overriding methods.
 (JNTU Aug/Sep 2008)
18. Explain about Object class in detail. (JNTU Apr 2007/Aug/Sep 2008)
19. (a) What is the difference between equality of objects and equality of objects and equality of references that refer to them?
 (b) What is the difference between a public member and a private member of a class?
 (c) Write an application that computes the value of e^x by using the formula:

$$e^x = 1 + x/1! + x^2/2! + x^3/3! + \dots$$

 (JNTU Aug/Sep 2008/sep 07)
20. (a) What is the difference overriding and overloading a method?
 (b) Write an application that finds the smallest of several integers. Assume that the first value read specifies the number values to input the user.
 (JNTU Aug/Sep 2008)
21. Explain the following methods of StringBuffer class and write a Java Program illustrating these. Length(), Capacity(), SetLength(), EnsureCapacity().
 (JNTU Aug/Sep 2008)
22. Write a program to illustrate the usage of the following methods of StringBuffer class. Explain the output in each case. Delete(), SetCharAt(), deleteCharAt(), append(), charAt(), getChars().
 (JNTU Aug/Sep 2008/ Nov 2004)
23. (a) What are conventional styles for class names, method names, constants and variables?
 (b) Can a java run on any machine? What is needed to run java on a computer?
 (c) Explain the concept of keywords. List some java keywords. (JNTU Apr 2007)
24. Write short notes on
 (a) Java applets and
 (b) Byte code.
 (c) What is the difference between a program compiled in Java and other languages like C,C++.
 (JNTU Aug/Sep 2006)

25. Write a program to demonstrate the concept of class scope and method scope. Explain the lifetime of a variable (JNTU March 2006)
26. What is an array? Why arrays are easier to use compared to a bunch of related variables? (JNTU Nov 2006, Apr 2007)
27. Write a program for transposition of a matrix using arraycopy command. (JNTU Nov, Apr 2006, 2007)
28. (a) What is a constructor? What are its special properties?
(b) How do we invoke a constructor?
(c) What are objects? How are they created from a class? (JNTU Apr 2007)
29. (a) How does String class differ from the StringBuffer class?
(b) Write program to do the following
i. To output the question "who is the inventor of java?"
ii. To accept an answer
iii. To print out "Good" and then stop, if the answer is correct.
iv. To output the message "try again", if the answer is wrong
v. To display the correct answer when the answer is wrong even at the third attempt. (JNTU Sep 2007)
30. (a) When dealing with very small or very large numbers, what steps would you take to improve the accuracy of the calculations
(b) What are symbolic constants? How are they useful in developing programs? (JNTU Sep 2007)
31. Write a program to create a class with a non default constructor and no default constructor. Create a second class that has a method which returns a reference to the first class. Create the object to return by making an anonymous inner class that inherits from the first class. (JNTU Sep 2007)
32. Briefly explain the constructors and methods of StringTokenizer class. (JNTU Nov 2006, Apr 2007)
- 33.** Explain the overloaded versions of the append() method of the StringBuffer class. (JNTU Nov 2006)
34. Describe different levels of access protections available in java. (JNTU Nov 2006)
35. Write a Java program to read set of strings and find out number of Palindromes in the given set. Use Buffered Reader. (JNTU Aug/Sep 2006)
36. Write program in Java to evaluate sine series. (JNTU Aug/Sep 2006)
37. Write a Java program to illustrate the usage of replace() method of String (JNTU Mar 2006)
38. Write a Java program to read a set of sentences from a text file and check the following:
Number of vowels
Number of Words
Number of characters
And produce a formatted report (JNTU Mar 2006)
39. Compare and contrast the String class with StringBuffer class. (JNTU Mar 2006)

40. Write a Java program to illustrate the usage of replace() method of StringBuffer class. Give output. (JNTU Mar 2006)
41. What is the purpose of String Buffer class? Explain briefly its constructors? (JNTU Mar 2006)
42. How many types of constructors are there in Java? Explain. (JNTU Mar 2006)
43. List and explain the control statements used in java. Also describe the syntax of the control statements with suitable illustration. (JNTU Apr 2006)
44. Using the following:
(a) nested if statements
(b) else if statements, and
(c) Conditional operator? (JNTU Aug/Sep 2006)
45. Write a Java program to read a text and find out whether the given Pattern is present in the text and if it is present find out no of occurrence. (JNTU Aug/Sep 2006)
46. Write a program to find factorial of a given number. (JNTU Nov 2006)
47. (a) Java is freeform language. Comment
(b) Describe in detail the steps involved in implementing a stand-alone program.
(c) What are command line arguments? How are they useful?
(JNTU Sep/Nov/Dec 2005/ 2007/ Nov 04)
48. What is a class? How does it accomplish data hiding? (JNTU Mar 2006)
49. Explain the difference between an object and a class. (JNTU Mar 2006)
50. Explain briefly the main concepts of object-oriented programming. (JNTU Mar 2006)
51. Write a program that takes a list of numbers and sorts them in descending order. (JNTU March 2006)
52. What is WWW? What is the contribution of Java to WWW? (JNTU March 2006)
53. Discuss the Lexical issues in Java. (JNTU Nov/Dec 2005)
54. Write a Java program for multiplying two matrices. (JNTU Nov 2005/Nov 04)
55. Write a Java program to find the sum of the series $1 + x + x^2 + x^3 + \dots$ (JNTU Nov/Dec 2005)
56. What is meant by mixed mode arithmetic? Briefly explain with an example. Explain bitwise operators available in Java. (JNTU Nov/Dec 2005)

57. How is portability achieved in Java? (JNTU Nov/Dec 2005)
58. Write a program segment that accomplishes each of the following.
(a) Calculate the integer part of Quotient when integer a is divided by integer b.
(b) Calculate integer remainder when integer a is divided by integer b.
(c) Use the program pieces developed in a) and b) to write a method displayDigit that receives an integer between 1 and 99999 and prints it as a series of digits, separated by one space and two successive integers by two spaces. For example, integers 4562 and 4563 should be printed as 4 5 6 2 5 6 2. (JNTU Nov/Dec 2005)
59. Write a Java program that uses overloaded constructors and this pseudovisible. (JNTU Nov 2005)
60. Write about Java API package. (JNTU March 2005)

UNIT-III

1. a) Explain the concept of inheritance with an example?
b) Explain the usage of super keyword with example? (JNTU FEB 2010)
2. a) Explain abstract classes with an example and also explain how to access abstract class?
b) Explain about final classes, final methods and final variables. (JNTU FEB 2010)
3. a) Explain different types of inheritance?
b) Explain the overriding with example? (JNTU FEB 2010)
4. a) Compare and contrast innerclass vs nested class?
b) Write a Java program to illustrate the usage of inner class.
c) What is the purpose of inheritance? (JNTU FEB 2010)
5. a) Explain about final classes, final methods and final variables?
b) Explain about the abstract class with example program? (JNTU FEB 2010)
6. Discuss about
a) Forms of inheritance
b) Using final with inheritance (JNTU FEB 2010)
7. What is the mechanism by which a call to an overridden method is resolved at runtime? Explain with an example. (JNTU AUG/SEP 2009)
8. How the keyword 'final' can be used to prevent Overriding ? explain With an example. (JNTU AUG/SEP 2009)
9. What is an abstract class? What is its importance? How is it designed in java. (JNTU AUG/SEP 2009)

10. Add a new method in the base class of shapes. Java that prints a message, but don't override it in the derived classes. Explain what happens. Now override it in one of the derived classes but not the others, and explain what happens. Finally, override it in all the derived classes; explain in detail about each situation. (JNTU AUG/SEP 2008)

11. Describe the following terms:

(a) final and abstract

(b) Passing parameter-call by value.

(JNTU AUG/SEP 2008)

12. How super class members are accessed in Java?

(JNTU AUG/SEP 2008, AUG/SEP 2006)

13. (a) Explain about final classes, final methods and final variables?

(b) Explain about the abstract class with example program?

(JNTU AUG/SEP 2008/NOV 04 AUG/SEP 2006)

14. Add a new method in the base class of shapes. Java that prints a message, but don't override it in the derived classes. Explain what happens. Now override it in one of the derived classes but not the others, and explain what happens. Finally, override it in all the derived classes; explain in detail about each situation. (JNTU AUG/SEP 2008)

15. (a) What is class? How does it accomplish data hiding?

(b) How do classes help us to organize our programs?

(c) Compare and contrast overloading and overriding methods.

(JNTU SEP 2007)

16. What is inheritance? What are the types of inheritances in java? Explain the member access mechanism in inheritance with an example. (JNTU SEP 2007/NOV 2006)

17. Create an inheritance hierarchy of Rodent: Mouse, Gerbil, Hamster, etc. In the base class, provide methods that are common to all Rodents, and override these in the derived classes to perform different behaviors depending on the specific type of Rodent. Create an array of Rodent, fill it with different specific types of Rodents, and call your base-class methods. Explain the output. (JNTU SEP 2007/ NOV 2006)

Is there any alternative solution for Inheritance. If so explain the advantages and disadvantages of it. (JNTU APR 2007)

18. Briefly explain following:

(a) garbage collection

(b) passing parameter-call by value

(JNTU SEP 2007)

19. What are the types of inheritances in java? Explain each of them in detail

(JNTU SEP 2007/NOV/DEC 05)

20. Create an abstract class with no methods. Derive a class and add a method. Create a static method that takes a reference to the base class, downcasts it to the derived class, and calls the method. In main(), demonstrate that it works. Now put the abstract declaration for the method in the base class, thus eliminating the need for the downcast.

(JNTU APR 2007)

21. What is Multiple Inheritance? Explain how it can be implemented in Java with the help of an example. (JNTU NOV 2006)
22. How can you prevent a class from instantiation.? (JNTU NOV 2006)
23. Explain the following.
(a) Virtual base class
(b) Pure virtual functions. (JNTU NOV 2006)
24. What are the uses of final and super keyword. Explain with example programs. (JNTU NOV 2006)
25. How to implement multiple inheritance in Java and explain with a suitable example (JNTU MAR 2006)
26. What is runtime polymorphism. Write a program to illustrate the concept of run-time polymorphism in Java? (JNTU NOV 2005)

UNIT-IV

1. a) Write the difference between classes and interfaces?
b) Define an interface.
c) Explain how multiple inheritances is implemented by using interfaces with example? (JNTU FEB 2010)
2. Write short notes on the following:
a) Inheritance
b) Static class variables
c) Package
d) Constructors (JNTU FEB 2010)
3. What are the Packages provided by java API? Explain. (JNTU AUG/SEP 2009)
4. Write Java code to obtain Date and Time. (JNTU AUG/SEP 2009)
5. Explain Buffered Reader and Buffered Writer classes from IO package with an example- (JNTU AUG/SEP 2009)
6. Explain FileInputStream and FileOutputStream with an example for each. (JNTU AUG/SEP 2009)
7. Write a sample program to illustrate packages. (JNTU AUG/SEP 2008)
8. Create an interface with at least one method, and implement that interface by defining an inner class within a method, which returns a reference to your interface. (JNTU AUG/SEP 2008)
9. Write a program to create a class with a non default constructor and no default constructor. Create a second class that has a method which returns a reference to the first class. Create the

object to return by making an anonymous inner class that inherits from the first class.
(JNTU AUG/SEP 2008)

10. Write a program create an interface U with three methods. Create a class A with a method that produces a reference to a U by building an anonymous inner class. Create a second class B that contains an array of U. B should have one method that accepts and stores a reference to a U in the array, a second method that sets a reference in the array (specified by the method argument) to null and a third method that moves through the array and calls the methods in U. In main(), create a group of A objects and a single B. Fill the B with U references produced by the A objects. Use the B to call back into all the A objects. Remove some of the U references from the B.
(JNTU APR/ SEP 2007)

11. Prove that the fields in an interface are implicitly static and final.
(JNTU SEP 2007)

12. Create three interfaces, each with two methods. Inherit a new interface from the three, adding a new method. Create a class by implementing the new interface and also inheriting from a concrete class. Now write four methods, each of which takes one of the four interfaces as an argument. In main(), create an object of your class and pass it to each of the methods.
(JNTU SEP 2007)

13. Create an interface with at least one method, in its own package. Create a class in a separate package. Add a protected inner class that implements the interface. In a third package, inherit from your class and, inside a method, return an object of the protected inner class, upcasting to the interface during the return. (JNTU APR 2007)

14. Write an interface called shape with necessary methods. Derive classes circle, rectangle, triangle, cone, sphere and cube with appropriate constructors and methods for area, volume also setting & displaying.
(JNTU NOV 2006)

15. Write a program to create an interface containing a static inner class. Implement this interface and create an instance of the inner class.
(JNTU NOV 2006)

16. What is a package and how do you design a package? (JNTU NOV 2006)

17. What are packages? How they can be accessed? (JNTU NOV 2006)

18. Explain about data abstraction? (JNTU NOV 2006)

19. Name any four Java packages. Give any two classes of a package. (JNTU MAR 2006)

20. What is an interface? What are the major differences and similarities between an interface and class?
(JNTU NOV/DEC 2005)

21. Discuss about the FileInputStream and FileOutputStream in java with examples.
(JNTU APR/MAY /NOV 2005)

22. Discuss about Object input class. (JNTU NOV 2005)

23. With an example explain Object Input Stream. List out the benefits of stream oriented I/O.
(JNTU NOV 2005)
24. Write a program that uses a SequenceInputStream to output the contents of two files.
(JNTU NOV/DEC /2005)
25. Describe the various forms of implementing interfaces. Give examples of java code for each case.
(JNTU NOV/DEC 2005)
26. Explain about Stream I/O. (JNTU NOV/DEC 2005/NOV 2004)
27. Discuss how I/O operations are performed in java with suitable examples.
(JNTU NOV 2005)
28. Discuss about the package of Java language. (JNTU NOV 2005)
29. Give general form of a multileveled package statement. What is the significance of the CLASSPATH environment variable in creating/using a package? (JNTU NOV 2005)
30. Discuss about Print Stream. (JNTU NOV 2005/NOV 2004)
31. Write a Java Program to find the second largest among given set of numbers using Sequence Input Stream.
(JNTU NOV 2005)
32. Discuss about the methods available in the Character Streams.
(JNTU APRIL/MAY 2005)
33. Write a Java Program to create a file with Student information such as name, roll no and six subject marks using File Write class.
(JNTU APRIL/MAY 2005)
34. With a program explain the use of File Reader. (JNTU APRIL/MAY 2005)

UNIT-V

1. Discuss various classes of java.util. Package. (JNTU FEB 2010)
2. Explain FileInputStream and FileOutputStream with an example for each.
(JNTU AUG/SEP 2009)
3. What are Checked Exceptions and Unchecked Exceptions? Explain some of these exceptions with an example and also give the differences between them.
(JNTU AUG/SEP 2009)
4. What is Error? What is Exception? Are they totally different or related? As a programmer what is the difference in handling an error and an exception. With the help of a simple java program explain the concepts error and exception.
(JNTU AUG/SEP 2008)
5. What is the necessity of exception handling? Explain exception handling taking “divide-by-zero” as an example.
(JNTU AUG/SEP 2008)

6. What is the difference between throw and throws statement. (JNTU AUG/SEP 2008)
7. What are the limitations of exception handling feature of java.
(JNTU SEP 2007)
8. In JAVA, is exception handling implicit or explicit or both. Explain with the help of example java programs. (JNTU SEP 2007)
9. Explain the following exceptions with the help of examples:
(a) Arithmetic Exception
(b) Null Pointer Exception
(JNTU APR 2007)
10. What is the role of stack in exception handling? (JNTU APR 2007)
11. Explain in detail any two unchecked exceptions. (JNTU APR 2007/NOV 2006)
12. How is an Exception handled in JAVA. (JNTU AUG/SEP 2006/NOV /DEC 2005)
23. Discuss about the nested try statement and how such a program may be executed?
(JNTU AUG/SEP 2006/NOV/DEC 2005/NOV 04)
14. List out by an example the way to create a user defined exception.
(JNTU AUG/SEP 2006/NOV/DEC 2005)
15. What are advantages of using Exception handling mechanism in a program?
(JNTU AUG/SEP 2006)
16. Give a note about how multitasking is implemented in Java? (JNTU MAR 2006)
17. Explain with an example, why exception handling is an effective means for dealing with constructor failure. (JNTU MAR 2006)
18. Explain how an exception handling mechanism can be used for debugging a program
(JNTU MAR 2006)
19. Create a try block that is likely to generate three types of exceptions and then incorporate necessary catch blocks to catch and handle them appropriately. (JNTU MAR 2006)
20. A program throws an exception and the appropriate exception handler begins execution, and this exception handler in turns throws the same exception. Is this above approach creating an infinite recursion? Justify your answer with an example.
(JNTU NOV/DEC 2005)
21. Why exception handling is considered as one of the important features in OOPS
(JNTU NOV/DEC 2005)
22. Describe the syntax of throw, throws, try, finally.

(JNTU NOV/DEC 2005/ NOV 2004)

23. What do you mean by an exception and error? Give the hierarchy of the exceptions in java. (JNTU NOV 2005)

24. List out the various java built in exception handlers. (JNTU NOV 2005)

25. When should a program throw an Exception? Explain. When do we use multiple catch handlers? (JNTU APR/MAY /NOV/DEC 2005)

26. List some of the most common types of exceptions that might occur in java. Give examples. (JNTU NOV 2005)

27. List out the various categories of compile time errors. (JNTU NOV 2005)

28. Write a program that illustrates the application of multiple catch statements. Write a program that demonstrates how certain exception types are not allowed to be thrown. (JNTU NOV 2005)

29. Write a program that illustrates the application of multiple catch statements? Write a program that demonstrates how certain exception types are not allowed to be thrown. (JNTU APR/MAY /NOV 2005)

30. Explain different types of Event methods and interfaces. (JNTU APR/MAY 2005)

31. When should a program throw an Exception? Explain. (JNTU APR/MAY 2005)

UNIT-VI

1. a) Explain the concept of multithreading.
b) Differentiate process and thread.
c) Explain Thread life cycle. [4+4+8] (JNTU FEB 2010)

2. In JAVA, is exception handling implicit or explicit or both. Explain with the help of example java programs.
(a) What is a thread? Explain the concept of a Multithread Programming?
(b) Describe the life cycle of a thread? (JNTU AUG/SEP 2009)

3.(a) With the help of an example, explain multithreading by extending thread class.
(b) Implementing Runnable interface and extending thread, which method you prefer for multithreading and why? (JNTU AUG/SEP 2008/ APR 2007)

4. How do you set and get priority values for threads in Java. (JNTU SEP 2007)

5. What is multithreading? What are its advantages? How does it improve the performance of java. Give an example of an application that needs multithreading? (JNTU NOV 2006)

6. Compare process based and thread based multitasking. (JNTU NOV 2006)
7. How multithreading in single processor system is different from multithreading in multiprocessor system. Explain. (JNTU NOV 2006)
8. How multithreading is achieved using Runnable interface. Write an example program (JNTU NOV 2006/NOV/DEC 2005)
9. How is priority level is incorporated in threads and what does it achieve? (JNTU AUG/SEP 2006)
10. Give a note about how multitasking is implemented in Java? (JNTU MAR 2006)
11. How is synchronization enforced? (JNTU MAR 2006)
12. How many ways are possible in java to create multiple threaded programs? (JNTU NOV/DEC 2005)
13. What is the difference between multiprocessing and multithreading? What is to be done to implement these in program? (JNTU NOV/DEC 2005)
14. How do we set priorities for threads? Describe the complete life cycle of a thread. (JNTU NOV/DEC 2005)
15. What is thread based preemptive multitasking. (JNTU APR/MAY 2005)

UNIT-VII

1. Explain in detail about the following event classes:
 - a) Component Event
 - b) Container Event
 - c) Focus Event [6+5+5] (JNTU FEB 2010)
2. (a) Explain the reason of creating a subclass of Frame is preferred over creating an instance of Frame when creating a window.
 - (b) Explain the steps in creating a subclass of frame with the help of examples. (JNTU AUG/SEP 2009)
3. Describe the ActionEvent, the AdjustmentEvent, the ComponentEvent and the ContainerEvent classes along with their methods and constructors (JNTU AUG/SEP 2009)
4. Briefly explain the following terms:
 - (a) Component.
 - (b) Container.
 - (c) Panel.
 - (d) Window. (JNTU AUG/SEP 2009)

5. What is event source? Give examples of event sources. How events are generated. Are all events generated by user actions? Comment on it. (JNTU AUG/SEP 2008)

6. Explain in detail about the following event classes:

(a) ComponentEvent

(b) ContainerEvent

(c) FocusEvent

(JNTU AUG/SEP 2008, APR 2007)

7. Write a Java Program which creates human face. (JNTU AUG/SEP 2008)

8. Write a Java Program which displays a clock. The edges are static. Three edges are required indicating hour, minute, and second. The length and width of edges should be approximately chosen. Display "Timex" at the centre of the clock. (JNTU AUG/SEP 2008)

9. Explain the steps in creating a subclass of frame with the help of examples.

(JNTU SEP 2007)

10. Briefly explain the components of AWT.

(JNTU SEP 2007)

11.(a) What is Delegation Event model? Explain it. What are its benefits?

(b) Define Event. Give examples of events. Define event handler. How it handles events.

(JNTU SEP 2007/ NOV 2006)

12. (a) Define Graphics context. How do you obtain graphics context.

(b) Explain in brief different drawing functions of Java.

(JNTU SEP 2007)

13. Explain the steps in creating a subclass of frame with the help of examples.

(JNTU APR 2007)

14. What are the methods supported by the following interfaces. Explain each of them

(a) ActionListener interface

(b) MouseMotionListener interface

(c) TextListener interface. (JNTU APR 2007)

15. What are the methods supported by KeyListener interface and MouseListener interface. Explain each of them with examples. (JNTU APR 2007)

16. What are the methods supported by the following interfaces. Explain each of them

(a) ActionListener interface

(b) MouseMotionListener interface

(c) TextListener interface. (JNTU APR 2007)

17. What are the methods supported by KeyListener interface and MouseListener interface. Explain each of them with examples. (JNTU APR 2007)

18. How using different fonts improves the user interface.

(JNTU APR 2007)

19. What is the functionality supported by java related to Fonts. Give overview of Java's event handling mechanism.

(JNTU NOV 2006/ NOV/DEC 2005)

20. What is meant by AWT? How will you create User Interfaces for applets?

(JNTU NOV 2006)

21. How event driven programming is different from Procedure oriented programming.
(JNTU NOV 2006)

22. How many types of nested classes are there? Explain each with an example.
(JNTU AUG/SEP 2006)

23. Write program to create Border layout with a component in each layout area.
(JNTU AUG/SEP 2006)

24. Write a program to illustrate Grid bag layout. (JNTU AUG/SEP 2006/NOV/DEC 2005)

25. Write a program, which illustrates all the mouse and window events. (JNTU MAR 2006)

26. Write down any seven event classes. (JNTU NOV/DEC 2005)

27. Discuss the Event Listeners. (JNTU NOV/DEC 2005)

28. What is an event driven programming and how is it structured?
(JNTU NOV/DEC 2005)

29. How events are categorized in Java? (JNTU NOV/DEC 2005)

30. Differentiate event sources and event listeners in an applet with an example.
(JNTU NOV/DEC 2005)

31. Write down any event generators. (JNTU NOV/DEC 2005)

32. Write down any five AWT components. (JNTU NOV/DEC 2005)

33. What is an adapter class and how can adapter classes be effective?
(JNTU NOV/DEC 2005)

34. Write a program that draws a series of eight concentric circles. The circles should be separated by 10 pixels.
(JNTU NOV/DEC 2005)

35. Write a program that randomly draws characters in different font sizes and colors.
(JNTU NOV/DEC 2005)

36. Describe about Event handling model? Explain about any three components in Swing.
(JNTU NOV/DEC 2005)

37. Write a program for creating and manipulating the radio button. (JNTU NOV/DEC 2005)

38. Explain the class hierarchy of various window types. (JNTU NOV/DEC 2005)
39. Write a program to draw several shapes in the created window using applets. (JNTU NOV 2005)
40. How to handle the various mouse events within the applet? (JNTU APR/MAY 2005)

UNIT-VIII

1. a) Define Applet. Explain life cycle of an Applet.
b) Create applet by passing parameters to applets? Example. [8+8] (JNTU FEB 2010)
2. Write a stand-alone AWT based application which creates a frame window that responds to mouse clicks and key strokes. (JNTU AUG/SEP 2009)
3. (a) What are the advantages of Layout managers? Why Java prefers Layout managers instead of fixing the component by x & y Coordinates? What are the different Layout managers AWT supports?
(b) What is preferred size of a component and how it is related to the Layout Managers. (JNTU AUG/SEP 2009)
4. (a) In what way JList differ from JComboBox?
(b) JList does not support scrolling. Why? How this can be remedied? Explain with an example. (JNTU AUG/SEP 2009)
5. What are various JFC containers? List them according to their functionality. Explain each of them with examples. (JNTU AUG/SEP 2009)
6. (a) Explain various components of User Interface.
(b) How will you arrange components on User Interface. (JNTU AUG/SEP 2008)
7. (a) In what way JList differ from JComboBox?
(b) JList does not support scrolling. Why? How this can be remedied? Explain. (JNTU AUG/SEP 2008)
8. (a) How will you create check boxes and choice boxes? Explain the steps in detail.
(b) Explain the differences between exclusive checkbox and non-exclusive check box. (JNTU AUG/SEP 2008)
9. What are containers? List various containers. Explain the usage of JPanel with example. (JNTU SEP 2007)
10. What is JFC? Explain the differences between JTextArea, JTextComponent, JTextField with examples. (JNTU SEP 2007)
11. Create an applet with two toolbars. One toolbar should be created using JButtons and a separator and another toolbar should be created using 3 custom Action classes. Add one to the "north" and another to the "south" sides of

border layout. When the user clicks one of the buttons in the toolbar, it will print a message to the console stating that which button is being pressed from which toolbar. Add functionalities to the buttons such as New, Open, Close, Save, Cut, Copy, Paste. (JNTU SEP 2007)

12. Differentiate following with suitable examples:

- (a) Frame, JFrame
- (b) Applet, JApplet
- (c) Menu, Jmenu. (JNTU APR 2007)

13. Explain the following:

- (a) Creating an applet
- (b) Passing parameters to applets
- (c) Adding graphics and colors to applets. (JNTU APR 2007)

14. Explain various methods of Applet class with necessary examples. (JNTU APR 2007)

15. What is the use of JFrame? Create a JFrame containing a JDesktoppane, which has a single JInternalFrame. (JNTU NOV 2006)

16. Explain the functionality of JComponent with example. Differentiate Jcomponent and JPanel. (JNTU NOV 2006)

17. What is the use of JPasswordField? Explain with an aid of an application program. (JNTU NOV 2006, APR 2007)

18. What are the differences between JPopupMenu and JMenu? (JNTU NOV 2006, APR 2007)

19. Write a java program to create an applet with six buttons representing your favorite six colors. When button is clicked, the background must change to the corresponding color. (JNTU AUG/SEP 2006)

20. Write a program that draws lines of random lengths in random colors. (JNTU MAR 2006)