OBJECT ORIENTED PROGRAMMING THROUGH JAVA

Subject Code: CS305ES
Regulations : R16 - JNTUH
Class : II Year B.Tech CSE I Semester

Department of Computer Science and Engineering

BHARAT INSTITUTE OF ENGINEERING AND TECHNOLOGY

Ibrahimpatnam - 501 510, Hyderabad

OBJECT ORIENTED PROGRAMMING THROUGH JAVA (CS305ES)

COURSE PLANNER

I. COURSE OVERVIEW:

Java is the most popular platform, which is used to develop several applications for the systems as well as embedded devices like mobile, laptops, tablets and many more. It is an object oriented programming language and has a simple object model, as it has derived from C and C++. It provides a virtual machine, which is accumulated with byte-code and can run on any system. With time the importance and popularity of Java is on rise as it has the magic in its remarkable abilities to innovate and morph as the technology landscape changes. It is the language of choice for developing applications for the BlackBerry Smartphone. It is important for information technology industry to develop and create multiple web-based or server based applications to enhance the industrial competency. There is huge scope for this programming language.

II. PREREQUISITE(S):

The knowledge of following subjects is essential to understand the subject:

1. Computer Programming
2. Data Structures

III. COURSE OBJECTIVES:

S. No.	Course Objectives
At the end of the course, the students will be able to:	
I.	To **introduce** the object oriented programming concepts.
II.	To **understand** object oriented programming concepts, and apply them in solving Problems.
III.	To **introduce** the principles of inheritance and polymorphism; and demonstrate how they relate to the design of abstract classes
IV.	To **introduce** the implementation of packages and interfaces
V.	To **introduce** the concepts of exception handling and multithreading.
VI.	To **introduce** the concepts of Collection Framework.
VII.	To **introduce** the design of Graphical User Interface using applets and swing controls.

IV. COURSE OUTCOMES:

S. No.	Course Outcomes (CO)
After completing this course the student must demonstrate the knowledge and ability to:	
CO1	Able to **understand** the use of OOPs concepts.
CO2	Able to **solve** real world problems using OOP techniques.
CO3	Able to **understand** the use of abstraction.
CO4	Able to **understand** the use of Packages and Interface in java.
CO5	Able to **develop** and **understand** exception handling, multithreaded applications with synchronization.
CO6	Able to **understand** the use of Collection Framework.
CO7	Able to **design** GUI based applications and **develop** applets for web applications.

V. HOW PROGRAM OUTCOMES ARE ASSESSED:

Program Outcomes (PO)		Level	Proficiency assessed by
PO1	**Engineering knowledge**: Apply the knowledge of mathematics, science, engineering fundamentals, and an engineering specialization to the solution of complex engineering problems.	3	Assignments, Tutorials, Mock Tests
PO2	**Problem analysis**: Identify, formulate, review research literature, and analyze complex engineering problems reaching substantiated conclusions using first principles of mathematics, natural sciences, and engineering sciences.	3	Assignments, Tutorials, Mock Tests
PO3	**Design/development of solutions**: Design solutions for complex engineering problems and design system components or processes that meet the specified needs with appropriate consideration for the public health and safety, and the cultural, societal, and environmental considerations.	3	Assignments, Tutorials, Mock Tests
PO4	**Conduct investigations of complex problems**: Use research-based knowledge and research methods including design of experiments, analysis and interpretation of data, and synthesis of the information to provide valid conclusions.	-	
PO5	**Modern tool usage**: Create, select, and apply appropriate techniques, resources, and modern engineering and IT tools including prediction and modeling to complex engineering activities with an understanding of the limitations.	-	
PO6	**The engineer and society**: Apply reasoning informed by the contextual knowledge to assess societal, health, safety, legal and cultural issues and the consequent responsibilities relevant to the professional engineering practice.	3	Assignments, Tutorials, Mock Tests --
PO7	**Environment and sustainability**: Understand the impact of the professional engineering solutions in societal and environmental contexts, and demonstrate the knowledge of, and need for sustainable development.	2	Assignments, Tutorials,
PO8	**Ethics**: Apply ethical principles and commit to professional ethics and responsibilities and norms of the engineering practice.	1	Assignments
PO9	**Individual and team work**: Function effectively as an individual, and as a member or leader in diverse teams, and in multidisciplinary settings.	-	--
PO10	**Communication**: Communicate effectively on complex engineering activities with the engineering community and with society at large, such as, being able to comprehend and write effective reports and design documentation, make effective presentations, and give	-	--

Program Outcomes (PO)		Level	Proficiency assessed by
	and receive clear instructions.		
PO11	**Project management and finance**: Demonstrate knowledge and understanding of the engineering and management principles and apply these to one's own work, as a member and leader in a team, to manage projects and in multidisciplinary environments.	-	--
PO12	**Life-long learning**: Recognize the need for, and have the preparation and ability to engage in independent and life-long learning in the broadest context of technological change.	2	Assignments, Tutorials

1: Slight (Low) **2: Moderate (Medium)** **3: Substantial (High)** **- : None**

VI. HOW PROGRAM SPECIFIC OUTCOMES ARE ASSESSED:

Program Specific Outcomes (PSO)		Level	Proficiency assessed by
PSO1	**Software Development and Research Ability:** Ability to understand the structure and development methodologies of software systems. Possess professional skills and knowledge of software design process. Familiarity and practical competence with a broad range of programming language and open source platforms. Use knowledge in various domains to identify research gaps and hence to provide solution to new ideas and innovations.	3	Lectures, Assignments, Tutorials, Mock Tests
PSO2	**Foundation of mathematical concepts:** Ability to apply the acquired knowledge of basic skills, principles of computing, mathematical foundations, algorithmic principles, modeling and design of computer- based systems in solving real world engineering Problems.	3	Lectures, Assignments, Tutorials, Mock Tests
PSO3	**Successful Career:** Ability to update knowledge continuously in the tools like Rational Rose, MATLAB, Argo UML, R Language and technologies like Storage, Computing, Communication to meet the industry requirements in creating innovative career paths for immediate employment and for higher studies.	2	Lectures, Assignments

1: Slight (Low)	**2: Moderate (Medium)**	**3: Substantial (High)**	**- : None**

VII. SYLLABUS:

UNIT – I

Object-oriented thinking- A way of viewing world – Agents and Communities, messages and methods, Responsibilities, Classes and Instances, Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts. Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions, control Statements, Introducing classes, Methods and Classes, String handling.

Inheritance– Inheritance concept, Inheritance basics, Member access, Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance, Polymorphism-ad hoc Polymorphism, pure polymorphism, method overriding, abstract classes, Object class, forms of inheritance- specialization, specification, construction, extension, limitation, combination, benefits of inheritance, costs of inheritance.

UNIT II

Packages- Defining a Package, CLASSPATH, Access protection, importing packages.

Interfaces- defining an interface, implementing interfaces, Nested interfaces, applying interfaces, variables in interfaces and extending interfaces.

Stream based I/O(java.io) – The Stream classes-Byte streams and Character streams, Reading console Input and Writing Console Output, File class, Reading and writing Files, Random access file operations, The Console class, Serialization, Enumerations, auto boxing, generics.

UNIT – III

Exception handling - Fundamentals of exception handling, Exception types, Termination or resumptive models, Uncaught exceptions, using try and catch, multiple catch clauses, nested try statements, throw, throws and finally, built- in exceptions, creating own exception sub classes.

Multithreading- Differences between thread-based multitasking and process-based multitasking, Java thread model, creating threads, thread priorities, synchronizing threads, inter thread communication.

UNIT -IV

The Collections Framework (java.util)- Collections overview, Collection Interfaces, The Collection classes- Array List, Linked List, Hash Set, Tree Set, Priority Queue, Array Deque. Accessing a Collection via an Iterator, Using an Iterator, The For-Each alternative, Map Interfaces and Classes, Comparators, Collection algorithms, Arrays, The Legacy Classes and Interfaces- Dictionary, Hashtable ,Properties, Stack, Vector

More Utility classes, String Tokenizer, Bit Set, Date, Calendar, Random, Formatter, Scanner

UNIT – V

GUI Programming with Swing – Introduction, limitations of AWT, MVC architecture, components, containers. Understanding Layout Managers, Flow Layout, Border Layout, Grid Layout, Card Layout, Grid Bag Layout.

Event Handling- The Delegation event model- Events, Event sources, Event Listeners,Event classes, Handling mouse and keyboard events, Adapter classes, Inner classes,Anonymous Inner classes.

A Simple Swing Application, **Applets** – Applets and HTML, Security Issues, Applets and Applications, passing parameters to applets. Creating a Swing Applet, Painting in Swing, A Paint example, Exploring Swing Controls- JLabel and Image Icon, JText Field, The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button, JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs.

SUGGESTED BOOKS:

TEXT BOOKS:

1. Java The complete reference, 9th edition, Herbert Schildt, McGraw Hill Education (India) Pvt. Ltd
2. Understanding Object-Oriented Programming with Java, updated edition, T. Budd, Pearson Education

REFERENCE BOOKS:

1. An Introduction to programming and OO design using Java, J. Nino and F.A. Hosch, John Wiley & sons.
2. Introduction to Java programming, Y. Daniel Liang, Pearson Education.
3. Object Oriented Programming through Java, P. Radha Krishna, Universities Press.
4. Programming in Java, S. Malhotra, S. Chudhary, 2nd edition, Oxford Univ. Press.
5. Java Programming and Object oriented Application Development, R. A. Johnson, Cengage Learning.

NPTEL Web Course: http://nptel.ac.in/courses/106106147/3

NPTEL Video Course: http://www.nptelvideos.com/java/java_video_lectures_tutorials.php

GATE SYLLABUS: NOT APPLICABLE

IES SYLLABUS: NOT APPLICABLE

VIII. COURSE PLAN:

Lecture No.	Week No.	TOPIC	Course Learning Outcome	Reference
	UNIT - 1			
1	1	A way of viewing world – Agents and Communities,	**Know** about Agents and Communities	Book No. 1, 2
2		Messages and methods, Responsibilities, Classes and Instances	**Gathering** Knowledge About Messages methods Responsibilities Classes and Instances	
3		Class Hierarchies- Inheritance, Method binding, Overriding and Exceptions, Summary of Object-Oriented concepts	**Understanding the** Class Hierarchies and Object-Oriented concepts	
4		Java buzzwords, An Overview of Java, Data types, Variables and Arrays, operators, expressions	**Know** about Java buzzwords, Overview of Java, Variables and Arrays, operators, expressions	

5	2	Control Statements, Introducing classes, Methods and Classes, String handling	**Know** about Control Statements, Methods, Classes, String handling	
6		**Inheritance**– Inheritance concept, Inheritance basics, Member access	**Compose** the Knowledge of the Followings: (a) Inheritance concept (b) Member access (c) Constructors (d) Creating Multilevel hierarchy, super uses, using final with inheritance (e) Polymorphism Concepts f) abstract classes	
7		Constructors, Creating Multilevel hierarchy, super uses, using final with inheritance		
8		Polymorphism-ad hoc Polymorphism, pure polymorphism, method overriding		
9	3	abstract classes, Object class, forms of inheritance- specialization, specification		
10		construction, extension, limitation, combination, Benefits of inheritance, costs of inheritance.		
1.		Review of Unit-I		
2.		***Mock Test – I***		
	UNIT – 2			
3.	4	Packages- Defining a Package, CLASSPATH,	**Gathering** Knowledge about Packages and Interfaces.	Book No. 1, 2
4.		Access protection, importing packages		
5.		**Interfaces-** defining an interface, implementing interfaces,		
6.		Nested interfaces		
		Tutorial / Bridge Class # 1		
7.	5	Applying interfaces, variables in interfaces and extending interfaces	**Know** about applying interfaces,variables in interfaces & extending interfaces	
8.		**Stream based I/O(java.io)** – The Stream classes-Byte streams and Character streams	**Compose** the Knowledge of the Followings: a) Stream classes b) Reading console Input and Writing Console	
9.		Reading console Input and Writing Console Output,		
10.		File class, Reading and writing Files		
		Tutorial / Bridge Class # 2		
11.	6	Random access file operations		
12.		The Console class, Serialization		

13.		Enumerations, auto boxing, generics.	Output c) File class d) Serialization , Enumerations	
14.		Revision		
		Tutorial / Bridge Class # 3		
	UNIT – 3			
15.	7	**Exception handling -** Fundamentals of exception handling, Exception types	**Compose** the Knowledge of the Followings: a) **Exception handling** b) built- in exceptions c) Creating own exception sub classes	Book No. 1, 2
16.		Termination or presumptive models		
17.		Uncaught exceptions, using try and catch, multiple catch clauses		
18.		Nested try statements, throw, throws and finally		
		Tutorial / Bridge Class # 4		
19.	8	built- in exceptions		
20.		Creating own exception sub classes.		
21.		**Multithreading-** Differences between thread-based multitasking and process-based multitasking	**Gathering** Knowledge about **Multithreading**	
22.		Revision		
		Tutorial / Bridge Class # 5		
	I Mid Examinations (Week 9)			
	UNIT – 3 Contd.			
23.	10	Java thread model, creating threads,	**Know** about Java thread model, thread priorities, synchronizing threads, inter thread communication	Book No. 1, 2
24.		thread priorities		
25.		synchronizing threads,		
26.		inter thread communication		
		Tutorial / Bridge Class # 6		
	UNIT – 4			
27.	11	**The Collections Framework (java.util)-** Collections overview, Collection Interfaces	**Gathering** Knowledge about **Collections Framework (java.util)-**	Book No. 1, 2
28.		The Collection classes- Array List, Linked List, Hash Set		
29.		Tree Set, Priority Queue, Array Deque.		
30.		Accessing a Collection via an Iterator, Using an Iterator		
		Tutorial / Bridge Class # 7		
31.	12	The For-Each alternative, MapInterfaces and Classes	**Compose** the	

32.		Comparators, Collection algorithms, Arrays,	Knowledge of the Followings: a) For-Each alternative, MapInterfaces and Classes b) Comparators c) Legacy Classes and Interfaces d) String Tokenizer, Bit Set e) Date, Calendar, Random Formatter, Scanner	
33.		The Legacy Classes and Interfaces- Dictionary, Hashtable		
34.		Properties, Stack, Vector More Utility classes		
		Tutorial / Bridge Class # 8		
35.	13	String Tokenizer, Bit Set		
36.		Date, Calendar,		
37.		Random Formatter, Scanner		
38.		Revision		
		Mock Test - II		
	UNIT – 5			
39.	14	**GUI Programming with Swing** – Introduction, limitations of AWT	**Gathering** Knowledge about **GUI Programming with Swing**	Book No. 1, 2, 4
40.		MVC architecture, components, containers		
41.		Understanding Layout Managers, Flow Layout, Border Layout,		
42.		Grid Layout, Card Layout, Grid Bag Layout		
		Tutorial / Bridge Class # 9		
43.	15	**Event Handling-** The Delegation event model- Events, Event sources, Event Listeners	**Know** about **Event Handling**	
44.		Event classes, Handling mouse and keyboard events, Adapter classes	**Know** about Event classes, Handling mouse and keyboard events, Adapter classes	
45.		Inner classes, Anonymous Inner classes. A Simple Swing Application	**Know** about Inner classes	
46.		**Applets** – Applets and HTML, Security Issues, Applets and Applications	**Gathering** Knowledge about **Applets**	
		Tutorial / Bridge Class # 10		
47.	16	passing parameters to applets Creating a Swing Applet, Painting in Swing, A Paint example	**Compose** the Knowledge of the	

48.		Exploring Swing Controls- JLabel and Image Icon, JText Field,	Followings: a) passing parameters to applets b) Swing Controls c) Swing Buttons	
49.		The Swing Buttons- JButton, JToggle Button, JCheck Box, JRadio Button,		
50.		JTabbed Pane, JScroll Pane, JList, JCombo Box, Swing Menus, Dialogs		
		Tutorial / Bridge Class # 11		
51.	17	Revision		
52.		Revision		
53.		Revision		
54.		Revision		
		Tutorial / Bridge Class # 12		

IX. MAPPING COURSE OUTCOMES LEADING TO THE ACHIEVEMENT OF PROGRAM OUTCOMES AND PROGRAM SPECIFIC OUTCOMES:

Course Outcomes	Program Outcomes (PO)												Prog Outc
	PO1	**PO2**	**PO3**	**PO4**	**PO5**	**PO6**	**PO7**	**PO8**	**PO9**	**PO10**	**PO11**	**PO12**	**PSO1**
CO1	3	3	3	-	-	2	3	3	-	-	-	3	2
CO2	3	3	3	-	-	2	3	3	-	-	-	3	2
CO3	1	1	1	-	-	2	2	2	-	-	-	2	1
CO4	3	3	3	-	-	2	3	3	-	-	-	3	2
CO5	3	3	3	-	-	2	3	3	-	-	-	3	2
CO6	1	1	1	-	-	2	2	2	-	-	-	2	1
C07	2	1	2			1	2	1				2	2
C08	1	2	2			1	2	1				1	2

X. QUESTION BANK: (JNTUH)

DESCRIPTIVE QUESTIONS

UNIT-I

Short Answer Questions-

SNO	Question	Blooms Taxonomy Level	Course Outcome
1	**What** is a constructor?	Knowledge	5
2	**What** are objects? **How** are they created from a class?	Knowledge	5
3	**What** is the difference between a public member and a private member of a class?	Analyze	2
4	**List** some java keywords?	Knowledge	5
5	**What** is an array?	Knowledge	5

Long Answer Questions-

S.No	Question	Blooms Taxonomy Level	Course Outcome

1	**What** is Java? **Explain** the features of Java.	**Analyze**	**2**
2	**Describe** the Java environment.	Knowledge	**5**
3	**Explain** the structure of Java program.	Understand	**3**
4	**Explain** the data types available in Java.	Knowledge	**5**
5	**Explain** type casting with example.	Knowledge	**5**

UNIT-2

Short Answer Questions-

S.No	Question	Blooms Taxonomy Level	Course Outcome
1	**What** are the types of inheritances in java?	Knowledge	5
2	**Explain** about Object class in detail.	Understand	3
3	**Define** the abstract class?	Knowledge	5
4	**Define** package?	Knowledge	5
5	**What** is Enumeration?	Knowledge	5

Long Answer Questions-

S.	Question	Blooms Taxonomy Level	Course Outcome
1	**What** are the types of inheritances in java? **Explain** each of them in detail	**Analyze**	**2**
2	Is there any alternative solution for Inheritance? If so **explain** the advantages and disadvantages of it.	**Apply**	**3**
3	**What** is interface? **Write** a program to demonstrate how interfaces can be extended.	Knowledge	**5**
4	**What** is package? **How** do you create a package? **Explain** about the access protection in packages?	Knowledge	**5**
5	**Explain** about final classes, final methods and final variables?	Understand	**3**

UNIT-3

Short Answer Questions-

S. No	Question	Blooms Taxonomy Level	Course Outcome
1	**Explain** creation of thread using run able interface with an example	Understand	3
2	**Explain** about inter –thread Communication with an example	Knowledge	4
3	**Explain** about thread interrupts with an example.	Understand	3
4	**Explain** producer consumer problem with an example	Understand	3
5	**Explain** the life cycle of a thread and multithreading	Knowledge	5

Long Answer Questions-

S.No	Question	Blooms Taxonomy Level	Course Outcome

1	With the help of an example, **explain** multithreading by extending Thread class ?	**Apply**	**3**
2	Implementing Run able interface and extending Thread, which method you prefer for multithreading and why?	**Analyze**	**2**
3	**What** is the role of stack in exception handling?	**Analyze**	**2**
4	Give the classification of exceptions	Knowledge	**5**
5	In Java, is exception handling implicit or explicit or both? **Explain** with the help of example Java programs	**Analyze**	**2**

UNIT-4

Short Answer Questions-

.No	**Question**	**Blooms Taxonomy Level**	**Course Outcome**
1	**Explain** StringTokenizer with an example.	Understand	3
2	**Explain** Random and scanner with examples.	Knowledge	4
3	**Explain** hash table and stack with examples.	Understand	3
4	**Explain** Array List and Vector with examples.	Knowledge	5
5	**Explain** enumeration and iterator with an example	Knowledge	5

Long Answer Questions-

Sn. N	**Question**	**Blooms Taxonomy Level**	**Course Outcome**
1	**Write** about Java Collection frame with example?	Knowledge	**5**
2	**Write** about a)Enumeration b)String Tokenizer c) Calendar and Properties	Knowledge	**5**
3	**What** is inter thread communication. What methods are employed?	Knowledge	**5**
4	**How** will you create strings and StringBuffers? How will you modify them?	**Analyze**	**2**
5	**Why** do you use frames?	Knowledge	**5**

UNIT-5

Short Answer Questions-

S.No	**Question**	**Blooms Taxonomy Level**	**Course Outcome**
1	**Explain** applet security issues?	Understand	2
2	**Write** a program for passing parameters to applet?	Apply	3
3	**Define** AWT?	Knowledge	5
4	**What** are various JFC containers?	Analyze	2
5	**Define** JFrame, JApplet, JDialog and Jpanel	Knowledge	5

Long Answer Questions-

S.	**Question**	**Blooms**	**Course**

		Taxonomy Level	Outcome
1	**What** are the methods supported by Key Listener interface and Mouse Listener interface. Explain each of them with examples.	Knowledge	**5**
2	**Explain** the functionality of JComponent with example. Differentiate JComponent and JPanel.	Knowledge	**5**
3	**Write** a java program which creates human face.	Apply	**3**
4	**What** are various JFC containers? **List** them according to their functionality. **Explain** each of them with examples.	Knowledge	**5**
5	**Write** a java program which draws a dashed line and dotted line using applet.	Apply	**3**

OBJECTIVE QUESTIONS:

JNTUH:

UNIT-1

1. Java is developed by _____

ans)Sun Microsystems of USA

2. _____ is one of the java features that enables java program to run anywhere anytime.

ans)Platform-Independent

3. Java compiler translates source code into ----------

ans) Bytecode (Virtual Machine Code)

4. Java interpreter translates _____ into machine code.

ans) Bytecode(Virtual Machine Code)

5. Java compiler produces an intermediate code known as ----------

ans)Bytecodes

6. _____ tool helps us to find errors in our programs.

Ans) jdb

7. The _____ includes hundred of classes and methods grouped into several function packages.

ans)API

8. The java interpreter uses _____ method before any objects are created. (b)

a) Main

9. _____ is use for naming classes, methods, variables etc in a program. (c)

ans)Identifier

10. _____ operator is used to construct conditional expression. (c)

ans)Ternary(?:)

UNIT-2

1._____ is default access specifier in JAVA. (a)

a)friendly b) private c) protected d) public

1. _ 2. _____ Variables and methods can be called without using the objects. (a)

a)static b) final c)abstract d) none of the above

2. 3

3.3. Which of the following statements is true? (b)

a)Java supports operator overloading b) Java supports interfaces
c) Java supports pointers d) Java supports multiple inheritance

4. _____ keyword is used to inherit a class. (b)
a)extend b) extends c)implement d) implements

5. We cannot create a subclass of _____ class. (d)
a)Abstract b) public c)static d) final

6. ____ methods must be override in the subclass. (c)
a) public b)final c) abstract d) static

7. What keyword is used in Java to define a constant?
a) static b) final c) abstract d) private (b)

8. _____ constructor is created when object of particular is created. (a)
a) default b) parameterized c) copy d) none

9. _____inheritance is not supported by java. (a)
a) Multiple b) multilevel c) hierarchical d) hybrid

10. We cannot create a subclass of ____class. (d)
a) abstract b) public c) static d)final

UNIT-3

1. Which is a checked Runtime Exception? [ ]
A. Null Pointer Exception B. Interrupted Exception
C. Arithmetic Exception D. ArrayIndexOutOfBoundsException

2. Which of the following method is not defined by MouseListener Interface? [ ]
A. Mouse Clicked B. mouse Dragged C. mouse Released D. mouse Exited

3. Which of the following layout is used as default layout manager? [ ]
A. Border Layout B. Card Layout C. Flow Layout D. Grid Layout

4. The following method is called when we leave a web page that contains an applet [ ]
A. Pause ( ) B. stop( ) C. destroy( ) D. hide( )

5. Which of the following method doesn't belong to Thread? [ ]
A.s Alive B. join C. sleep D. wake

6. Which event is generated when a scrollbar is manipulated? [ ]
A. Item Event B. Adjustment Event C. Check Event D. Text Event

7. Which block following will execute whether or not an exception is thrown? [ ]
A. Try B. Catch C. Throw D. Finally

8. _____method is used to find the nth no.of character of given string s1.
A. s1.index(n) B. S1.substr(n) C. 1.length() D. s1.charAt(n) [D]

9. Character streams can be used to read and write _____ Unicode characters.
Ans) 16-bit

10 "Converting invalid string to a number" is the _____type of error.
Ans) Runtime

UNIT -4

1. Which of the following is the valid priority we can use for thread? [ ]
A.MIN_PRIORITY B. MINIMUM_PRIORITY
C. LOW_PRIORITY D. ZERO_PRIORITY

2. Which listener interface is needed in handling Text Field? [ ]
A.ActionListener B. Item Listener C. Text Listener D. Input Listener

3. The following method is called when we leave a web page that contains an applet [ ]
A. pause ( ) B. Stop ( ) C. Destroy ( ) D. Hide ( )

4. What is the argument type of program's main () method? [ ]
A. Character array B. String array C. String D. Character

5. Which one does not have a value Of (String) method [ ]
A. Long B. Boolean C. Character D. Integer

6. Which operator is used to create and concatenate string? [ ]

A. ++ B. && C. & D. +

7. Which of the following are Java keywords (asopposedtoreservedwords)? [ ]

A. go to B. FALSE C. Extends D. malloc

8. What is the minimum value of char type? Select the one correct answer. [ ]

A. 0 B. -215 C. 215-1 D. -28

9. _____is a passive control. [ ]

a. TextField b. Label c. Button d. TextArea

10. _____is a single line edit control. [ ]

a. TextField b. Label c. Button d. TextAreaControl

UNIT -5

1. 1. _____ is used to connect Java's I/O system to other programs.
2. The _____ allows us to pass parameters to the Applet through HTML page.
3. TCP/IP is used to implement _____ connection.
4. The class Event Object is defined in _____ package.
5. The _____ function can be used to find IP address of the host machine.
6. The fundamental class of Java swing JApplet extends _____ class.
7. Border Layout manager divides window in to _____ areas.
8. A try block may have _____ number of catch block(s).
9. To select or to change the font we have to use _____ method.
10. At the top of the AWT hierarchy is the _____ class.

XII. GATE QUESTIONS: NOT APPLICABLE

XIII. WEBSITES:

1 http://www.javatpoint.com/

2 **java**.sun.com/docs/books/tutorial/**java**/TOC.html

3 http://www.learnjavaonline.org/

4 http://www.tutorialspoint.com/java/

5 www.**java**.com/en/download/faq/develop.xml

6 www.oracle.com › Java › Java SE

7 www.w3schools.com

XIV. EXPERT DETAILS:

1. Prof. Sriram K Rajamani,Ph.D (IITH)
2. Dr. R.B.V. Subramanian (NITW)

XV. JOURNALS:

1. iosrjournals.org/
2. java.sys-con.com/
3. cisjournal.org/journal of computing/
4. www.bizjournals.com
5. www.ijcaonline.org

XVI. LIST OF TOPICS FOR STUDENT SEMINARS:

1. Classes & Instances
2. Inheritance
3. Method Binding
4. Java Buzzwords & Data Types
5. Classes & Objects
6. Access Control
7. Method Overloading
8. Method Overriding
9. Packages & Interfaces

10. Exception Handling
11. Multithreading
12. Event Handling
13. Applets
14. Swings
15. Black Berry technology
16. Facebook thrift
17. Blue jacking
18. Mobile number portability
19. Cyber crime
20. Joomla &CMS

XVII. CASE STUDIES / SMALL PROJECTS:

PROJECT 1:

Private Banking Network java project explains about developing a software application forbanks for working, communication and sharing information inside organization. This project report covers purpose of intranet banking, literature survey, system analysis, software and hardware requirement, system architecture and system testing. This project is developed in java platform using socket programming.

This application works like a distributed systems using OOPS and networking concepts. Private banking network is a intranet application which works inside organization. Systems inside organization are interlinked with LAN and other networks are connected using leased lines. Intranet connections are connected to external networks using different gateways. In banking sector communication between employees is compulsory so this application will help to communicate, share documents and allow teleconference features for employees inside organization. This software provides scope for further development by integrating software with appropriate hardware equipment and uses it as ATM software. This application can be useful only for banking sector and present application is restricted to offline banking, accounting, money management. With few modifications this software can be interlinked with online banking.

PROJECT 2:-

Control System in Network Environment is implemented using java. This projects is useful for computer science final year students who are interested in developing network related project.

In cellular communication networks, the geographical area is divided into smaller regions, called cells. In each cell, there is one Mobile Service Station (MSS) as well as a number of Mobile Hosts (MH). The communication between MSSs is, in general, through wired links, while the links between an MH and MSS is wireless. A Mobile Host can communicate with other Mobile Hosts in the system only through the Mobile Service Station in its cell.

There are two kinds of channels available to an MH: communication channel and control channel. The former is used to support communication between an MH and the MSS in its cell, while the latter is set aside to be used exclusively to send control messages that are generated by the channel allocation algorithm. In this paper, henceforth, unless specified otherwise, the term channel or wireless channel refers to a communication channel. When an MH wants to communicate with another MH, it sends a request message to the MSS in its own cell. The request can be granted only if a wireless channel can be allocated for communication between MSS and MH. Since channels are limited, they should be reused as much as possible.