SAMSKRUTI COLLEGE OF ENGINEERING & TECHNOLOGY
(Approved by AICTE, New Delhi & Affiliated to JNTUH.)
Kondapur(V), Ghatkesar(M), Medchal(Dist)

Subject Name: JAVA PROGRAMMING LANGUAGE
Prepared by: Dr. K. Ramasubramanian, Assistant Professor, CSE
Year, Semester, Regulation: II Year- II Sem (R18)

UNIT - I	
PART – A (Short Answer Questions)	
1	**Define** OOP?
2	**Distinguish** between procedural language and OOPs?
3	**Define** Encapsulation?
4	**Define** Inheritance?
5	**Define** Polymorphism?
6	**List** advantages of OOP?
7	**List** disadvantages of OOP?
8	**Explain** briefly out history of java?
9	**Explain** briefly about different types of data types in java?
10	**What** is constant? Explain different types of constants?
11	**What** is Variable? **Describe** scope and life time of variables?

S. No	Question
12	**List** and describe different types of operators?
13	**Define** type conversion?
14	**Define** type casting?
15	**Define** enumerated types?
16	**Define** an array?
17	**Define** this reference?
18	**Define** constructor?
19	**Define** recursion?
20	**Define** garbage collection?
PART – B (Long Answer Questions)	
1	**Discuss** the various characteristics of object oriented programming concepts?
2	**Explain** briefly about the features (buzzwords) of Java.
3	**Compare** various differences between Java and C++.
4	**Why** Java is a pure object oriented programming language
5	**Distinguish** between procedure oriented and object oriented programming.
6	**Explain** the importance of **this** keyword in java with an example program
7	**What** is method overloading? Explain with an example program
8	**Discuss** about the method overloading and constructor overloading with an example.
9	**What** is Array? Explain different types of array and also discuss its advantages.
10	**Explain** briefly about String class and discuss various methods in String class with an example.
11	**Explain** about the console input and output with an example.
12	**What** is a conditional statements? Discuss various types with examples.
13	**Explain** the different types of constructors in java.
14	**What** is the use of break and continue statements in java? Explain with an example.
15	**What** is comment? Explain the different types of comments that are used in java.
16	**What** is the use of the operators in java? Explain with an example.
17	**Define** static field and method? Explain with an example.
18	**What** are the different types of operators present in java? Explain
19	**What** is type conversion and casting? Explain with an example.
20	**Explain** each of the primitive datatypes present in java? Discuss in detail.
21	**Discuss** about various techniques of parameter passing with an example program.

PART – C (Problem Solving and Critical Thinking Questions)

1

```
Student john12 = new Student(1001, "John",
12); Student john13 = new Student(1002,
"John", 13);
System.out.println("comparing John, 12 and John, 13 with
compareTo
:" + john12.compareTo(john13));
```

then **predict** the output of the code?

2

What is the output of the program? class Lifetime

```
{
public static void main(String args[])
{
int x;
for (x=0; x<3; x++)
{
int y=-1;
System.out.pirnltn(“ y is :” +
y); y=100;
System.out.println(“ y is now : “ + y);
}}}
```

3

What will be the output of the program?

```
public class If2
{
    static boolean b1, b2;
    public static void main(String [] args)
    {
        int x =
        0; if (
        !b1 )
 {
            if ( !b2 )
            {
                b1 =
                true;
                x++;
                if ( 5 > 6 )
                {
                   x++;
                }
                if ( !b1 )
                    x = x + 10;
                else if ( b2 = true
                    ) x = x + 100;
                else if ( b1 | b2
                    ) x = x +
                    1000;
            }
        }
        System.out.println(x);
```

4 **Explain** the following code legal in Java? is it example of method overloading or overriding?

```
public String getDescription(Object
   obj){ return obj.toString;
}
public String getDescription(String
   obj){ return obj;
}
and
public void getDescription(String obj)
{ return obj;
}
```

5 **Analyze** the following program and find the output of the program?

```
public class CounterAtomic {
        private AtomicLong counter = new AtomicLong();

        public void increment() {
                counter.incrementAndGet();
        }

        public long get() {
                return counter.get();
        }
}
```

6 What will be the output of the program?

```
public class Test
{
    public int aMethod()
    {
        static int i =
        0; i++;
        return i;
    }
    public static void main(String args[])
    {
        Test test = new
        Test();
        test.aMethod();
        int j = test.aMethod();
        System.out.println(j);
    }
}
```

7 What will be the output of the program?

```
public class Test
{
    public static void main(String args[])
    {
        int i = 1, j =
        0; switch(i)
        {
            case 2: j += 6;
            case 4: j  +=
            1; default: j
            += 2; case 0:
            j += 4;
        }
        System.out.println("j = " + j);
    }
}
```

8 Analyze the following program and find the output of the program?

```
Class Test
{
public static void main(String args[])
{
int x, y;
y=20;

for(x=0; x<10: x++)
{
System.out.println("this is x:" +x);
System.out.println("this is y:" +y);
y= y-2;
}
}
}
```

9 What will be the output of the program?

```
class BitShift
{
    public static void main(String [] args)
    {
        int x = 0x80000000;
        System.out.print(x + " and
        "); x = x >>> 31;
        System.out.println(x);
    }
}
```

10 Analyze and find out What will be the problem in the program?

```
class Equals
{
   public static void main(String [] args)
   {
      int x = 100;
      double y =
      100.1;
      boolean b = (x = y);
      System.out.println(b);
   }
}
```

UNIT - II

PART – A (Short Answer Questions)

1	**Define** Inheritance?
2	**List** types of inheritances in java?
3	**What** are Member access rules in java?
4	**Explain** the Uses of “Super” keyword
5	**Explain** the Uses „final‟ keyword with inheritance
6	**What** is Object class?
7	**What** is abstract class?
8	**Define** polymorphism?
9	**Define** dynamic binding?
10	**Define** method overriding?
11	**Compare and Contrast** differences between interfaces vs Abstract classes.
12	**Define** interface?
13	**What** is an inner class?
14	**Define** static inner classes.
15	**What** is package?
16	**Define** various steps for creating and importing packages.
17	**What** are abstract methods?
18	**What** is CLASSPATH?
19	**List** advantages of inheritance?
20	**How** we implement an interface. Explain.

1	**What** is Inheritance? Discuss its uses and Hierarchical abstractions?
2	**List** different types of inheritances in java? Explain each of them in detail with an example programs.
3	**Discuss** about Object class and its methods.
4	**Explain** the Usage of “Super” keyword, with an example program.
5	**What** is package? **Explain** different types of Packages.
6	Compare various differences between interfaces and abstract class.
7	**What** is dynamic binding? Explain with an example?
8	**Define** method overriding? Write with an example?
9	**Explain** the usage of abstract classes and methods? With an example programs.

10	**Discuss** how interfaces are defined and implemented.
11	**Explain** the uses of inner classes and local inner classes. With an example?
12	**Discuss** in detail about creating and importing packages with an examples.
13	**Discuss** in detail how packages can be defined, creating and accessed.
14	**How** interfaces can be extended? Explain.
15	**What** is interface? Write Differences between classes and interfaces?
16	**What** is final keyword? Explain its importance in java with an example program.
17	**What** is inheritance? Explain the benefits of inheritance with an example
18	**What** are various Member access rules explain with an example.
19	**Discuss** the various levels of Access protection available for packages and their implications.
20	**Compare** and **Contrast** overloading and overriding methods.

PART – C (Problem Solving and Critical Thinking Questions)

1 Analyze the program and give output

```
public class Foo
{
   Foo()
   {
      System.out.print("foo");
   }

class Bar
{
   Bar()
   {
      System.out.print("bar");

   }
   public void go()
   {
      System.out.print("hi");
   }
} /* class Bar ends
          */
   public static void main (String [] args)
   {
      Foo f = new
      Foo();
      f.makeBar();
   }
   void makeBar()
   {
      (new Bar() {}).go();
   }
      }/* class Foo ends */
```

2	**Explain** the following code legal in Java?
	
	class OuterClass {
	private int privInt = 10;
	public void createInnerClass() {
	InnerClass inClass = new
	InnerClass(); inClass.accessOuter();
	}
	class InnerClass {
	public void accessOuter() {
	System.out.println("The outer
	class's privInt is " + privInt);
	}
	}
	}
	public static void main(String[] args)
	{
	OuterClass outClass = new OuterClass();
	OuterClass.InnerClass inner = outClass.new InnerClass();
	inner.accessOuter();
	}		

3	**Analyze** the following program and find the output of the program?
	class A {
	//Members and methods declarations.
	}
	class B extends A {
	//Members and methods from A are inherited.
	//Members and methods declarations of B.
	}
	class A {
	public A()
	{
	System.out.println("New A");
	}
	}
	class B extends A {
	public B()
	{
	super(); System.out.println("New B");
	}
	}		

<table>
<tr><td>4</td><td>

Illustrate the output of the following program?

```
interface MyInterface
{
   public void method1();
   public void method2();
}
class XYZ implements MyInterface
{
  public void method1()
  {
      System.out.println("implementation of method1");
  }
  public void method2()
  {
      System.out.println("implementation of method2");
  }
  public static void main(String arg[])
  {
      MyInterface obj = new XYZ();
      obj. method1();
  }
}
```

</td><td></td><td></td></tr>
<tr><td>5</td><td>

What is the output of the program?

```
Class A
{
    final public int GetResult(int a, int b)
    { return 0; }
}
class B extends A
{
    public int GetResult(int a, int b)
    {return 1; }
}
public class Test
{
    public static void main(String args[])
    {
        B b = new B();
        System.out.println("x = " + b.GetResult(0, 1));
    }
}
```

</td><td></td><td></td></tr>
</table>

6 What will be the output of the program?

```
class Super
{

    public int i = 0;

    public Super(String text)
    {
        i = 1;
    }
}

class Sub extends Super
{
    public Sub(String text)
    {
        i = 2;
    }
    public static void main(String args[])
    {
        Sub sub = new Sub("Hello");
        System.out.println(sub.i);
    }
}
```

7 What will be the output of the program?

```
interface Count
{
    short counter = 0;
    void countUp();
}
public class TestCount implements Count
{
    public static void main(String [] args)
    {
        TestCount t = new
        TestCount(); t.countUp();
    }
    public void countUp()
    {
        for (int x = 6; x>counter; x--, ++counter)
        {
            System.out.print(" " + counter);
        }
    }
}
```

8 Analyze and find out the output of the program?

```
public class Test
{
    public int aMethod()
    {
        static int i =
        0; i++;
        return i;
    }
    public static void main(String args[])
    {
        Test test = new
        Test();
        test.aMethod();
        int j =
        test.aMethod();
        System.out.println(j);
    }
    }
```

9 Develop a Java Program to create an abstract class named Shape that contains two integers and an empty method named printArea(). Provide three classes named Rectangle, Triangle and Circle such that each one of the classes extends the class Shape. Each one of the classes contains only the method print Area () that prints the area of the given shape.

10 Analyze and find out the output of the program?

```
package mypack;

class Book
{
 String bookname;
 String author;
 Book(String b, String c)
 {
  this.bookname = b;
  this.author = c;
 }
 public void show()
 {
  System.out.println(bookname+" "+ author);
 }
}

class test
{
 public static void main(String[] args)
 {
  Book bk = new Book("java","Herbert");
  bk.show();
 }
 }
```

	UNIT - III
PART – A (Short Answer Questions)	
1	**Define** Exception?
2	**Distinguish** between exception and error?
3	**What** are the benefits of exception handling
4	**Explain** the classification of exceptions.
5	**Define** checked exceptions?
6	**Define** unchecked exceptions?
7	**Define** built in exceptions?
8	**Explain** the usage of try and catch
9	**Explain** the usage of throw, throws and finally.
10	**Distinguish** between throw and throws?
11	**Distinguish** between process and thread?
12	**What** are thread states? Explain.
13	**What** are the different ways to create a thread?
14	**Define** producer consumer problem?
15	**Define** inter-thread communication?
16	**How** threads are synchronized?
17	**What** are different thread priorities
18	**How** many ways can thread be Created?
19	**Explain** about the alive() and join() method
20	**Explain** about “thread class implements Runnable interface”.

1	**What is an** exception? Explain how an exception can be handled in Java? And also list the benefits of Exception Handling.
2	**What** are try, catch, and finally keywords in with an example
3	**Explain** the usage of throw keyword? Write with an example
4	**Define** the usage of throws keyword in exception handling? Explain with an example program.
5	**Define** a exception called “NotEqualException” that is thrown when a float value is not equal to 3.14. Write a java program that uses the above user defined exception.
6	**Differentiate** between Checked and Unchecked exceptions?
7	**Explain** the process of creating an own exception subclass with an example.
8	**Develop** a java program to implement built in exceptions?
9	**Explain** the various ways of creation of thread with an examples.
10	**What** is multithreading? Explain.
11	**What** is a thread? Explain the states of a thread with an example.
13	**Distinguish** between multi-tasking and multi-threading
15	**Explain** in detail the process of creating thread with an example
16	**How** do we set priorities for threads? Explain.

17	**What** is synchronization? Explain with an example program.
18	**Discuss** about producer consumer pattern with an example.
19	**How** is interthread communication achieved? Give an example.
20	**Explain** the advantages of multithreading.

PART – C (Problem Solving and Critical Thinking Questions)

1 What will be the output of the program?

```
public class X
{
    public static void main(String [] args)
    {
        try
        {
            badMethod();
            System.out.print("A")
            ;
        }
        catch (Exception ex)
        {
           System.out.print("B");
        }
        finally
        {
           System.out.print("C");
        }
        System.out.print("D");
    }
    public static void badMethod()
    {
        throw new Error();
    }
        }
```

2	Analyze the program and find the output ?
	
	public class Test
	{
	public static void aMethod() throws Exception
	{
	try {
	throw new Exception();
	}
	finally {
	System.out.print("finally
	");
	}
	}
	public static void main(String args[])
	{
	try
	{
	aMethod();
	}
	catch (Exception e) {
	System.out.print("exception
	");
	}
	System.out.print("finished");
	}
	}		

3	What will be the output of the program?
	
	class s1 implements Runnable
	{
	int x = 0, y = 0;
	int addX()
	{
	x++;
	return x;}
	int addY()
	{y++;
	return y;}
	public void run()
	{
	for(int i = 0; i < 10; i++)
	System.out.println(addX() + " " + addY());
	}
	public static void main(String args[])
	{
	s1 run2 = new s1();
	Thread t1 = new Thread(run1);
	Thread t2 = new Thread(run2);
	t1.start();
	t2.start();
	}
	}		

4	**Explain** the output of the following program?
	class Exceptions {
	  public static void main(String[] args) {
	
	  String languages[] = { "C", "C++", "Java", "Perl", "Python" };
	
	  try {
	    for (int c = 1; c <= 5; c++) {
	      System.out.println(languages[c]);
	    }
	  }
	  catch (Exception e) {
	    System.out.println(e);
	  }
	  }
	}		

5	**What** is the output of the below program?
	class Allocate {
	  public static void main(String[] args) {
	
	    try {
	      long data[] = new long[1000000000];
	    }
	    catch (Exception e) {
	      System.out.println(e);
	    }
	
	    finally {
	      System.out.println("finally block will execute always.");
	    }
	  }
	}
	then **predict** the output?		

6	What will be the output of the program?
	
	class MyThread extends Thread
	{
	    public static void main(String [] args)
	    {
	        MyThread t = new MyThread();
	        Thread x = new Thread(t);
	        x.start();
	    }
	    public void run()
	    {
	        for(int i = 0; i < 3; ++i)
	        {
	            System.out.print(i + "..");
	        }
	    }
	        }		

7 What will be the output of the program?

```
public class RTExcept
{
    public static void throwit ()
    {
        System.out.print("throwit ");
        throw new RuntimeException();
    }
    public static void main(String [] args)
    {
        try
        {
            System.out.print("hello ");
            throwit();
        }
        catch (Exception re )
        {
            System.out.print("caught ");
        }
        finally
        {
            System.out.print("finally ");
        }
        System.out.println("after ");
    }
}
```

8 Analyze the program and find the output

```
public class NFE
{
    public static void main(String [] args)
    {
    String s = "42";
        try
        {
            s = s.concat(".5");
            double d = Double.parseDouble(s);
            s = Double.toString(d);
            int x = (int) Math.ceil(Double.valueOf(s).doubleValue());
            System.out.println(x);
        }
        catch (NumberFormatException e)
        {
            System.out.println("bad number");
        }
    }}
```

9 What will be the output of the program?

```
class MyThread extends Thread
{
    MyThread()
    {
        System.out.print(" MyThread");
    }
    public void run()
    {
        System.out.print(" bar");
    }
    public void run(String s)
    {
        System.out.println(" baz");
    }
}
public class TestThreads
{
    public static void main (String [] args)
    {
        Thread t = new MyThread()
        {
            public void run()
             {
                System.out.println(" foo");
             }
        }
        t.start();
    }
}
```

10 What will be the output of the program?

```
class implements Runnable
{
    int x, y;
    public void run()
    {
        for(int i = 0; i < 1000; i++)
            synchronized(this)
            {
                x = 12;
                y = 12;
            }
        System.out.print(x + " " + y + " ");
    }
    public static void main(String args[])
    {
        s run = new s();
        Thread t1 = new Thread(run);
        Thread t2 = new Thread(run);
        t1.start();
        t2.start();
    } }
```

UNIT - IV

PART – A (Short Answer Questions)

1	**Define** collections?
2	**Define** Java collection Frame work.
3	**Define** Array **List** with syntax
4	**Define** Vector with syntax.
5	**Define** hash table with syntax.
6	**Define** stack with syntax.
7	**Define** enumeration with syntax.
8	**What** is Iterator?
9	**Explain** the function of StringTokenizer.
10	**Define** random class.
11	**Define** Scanner class.
12	**Define** Calendar class.
13	**Define** Properties class
14	**Define** Stream?
15	**Define** byte stream?
16	**Define** character stream?
17	**Define** text input/output file?
18	**Define** JDBC?
19	**List** types of JDBC drivers.
20	**Define** random access file?

PART – B (Long Answer Questions)

1	**Explain** the concept of Java collection frame. Write a brief overview on it.
2	**Explain** the concept of generics and generic collections in java.
3	**Explain** briefly about Vector class with an example.

4	**What** is hash table? Explain with an example?		
5	**Demonstrate** about stack class with an example.		
6	**What** is enumeration? Explain about with an example.		
7	**Explain** briefly about iterator with an example.		
8	**Discuss** about StringTokenizer class. Explain with an example.		
9	**Explain** in detail about Random class with an example.		
10	**What** is Scanner class? Explain with an example.		
11	**Explain** in detail Calendar class with an example.		
12	**What** are text input/output file operations? Explain		
13	**Explain** in details about binary input/output file operations with examples.		
14	**Discuss** the concept of Random access file. Also explain its operations.		
15	**How** a file can be managed using file class? Explain.		
16	**Distinguish** between a)InputStream and Reader classes b)OutputStream and Writer Classes		
17	**Explain** in detail about the types of drivers in JDBC.		
18	**Discuss** the process of connecting to database using JDBC.		
19	**Develop** a JDBC application for updating data.		
20	**Explain** the process of getting/accessing metadata for a resultset.		

1	What will be the output of the program?
	
	package foo;
	import java.util.Vector;
	private class MyVector extends Vector
	{
	    int i = 1;
	    public MyVector()
	    {
	        i = 2;
	    }
	}
	public class MyNewVector extends MyVector
	{
	    public MyNewVector ()
	    {
	        i = 4;
	    }
	    public static void main (String args [])
	    {
	        MyVector v = new MyNewVector();
	    }
	}		

2	**Analyze** the following program and find the output of theprogram?
	public class Test {
	    public static void main(String[] args) {
	        System.out.println(Math.min(Double.MIN_VALUE, 0.0d));
	    }
	}		

3	Find the output of the program.
	import java.util.*;
	class I
	{
	public static void main (String[] args)
	{
	Object i = new ArrayList().iterator();
	System.out.print((i instanceof List)+",");
	System.out.print((i instanceof Iterator)+",");
	System.out.print(i instanceof ListIterator);
	}
	}		

4	**Compare** the output before adding and after adding?
	import java.util.*;
	
	public class ArrayListExample {
	public static void main(String args[]) {
	ArrayList<String> obj = new ArrayList<String>();
	obj.add("Ajeet");
	obj.add("Harry");
	System.out.println("Currently the array list has following elements:"+obj);
	obj.add(0, "Rahul");
	obj.add(1, "Justin");
	obj.remove("Chaitanya");
	obj.remove("Harry");
	System.out.println("Current array list is:"+obj);
	obj.remove(1);
	System.out.println("Current array list is:"+obj);
	}
	}		

5	What is the output of the following program
	import java.util.*;
	
	public class ArrayListExample
	{
	public static void main(String args[])
	{
	ArrayList<String> a1 = new ArrayList<String>();
	a1.add(1);
	a1.add(2);
	a1.add(3);
	a1.add(4);
	System.out.println("Current array list is:"+a1);
	Integer ia[]= new Integer [a1.size()];
	ia= a1.toArray(ia);
	int sum=0;
	for (int i : ia) sum += i;
	System.out.println("Sum is:"+sum)
	}
	}		

6	Analyze the program and explain the importance of **line 1** and give the output
	import java.util.StringTokenizer;
	class Stdemo
	{
	Static String in= “title= java : the complete reference” +
	“author= schildt” +
	“publisher= Osborne/mcgraw-hill “ +
	“copyright= 2005”;
	public static void main(String args[])
	{
	String key= st.nextToken(); -----1
	String val= st.nextToken();
	System.out.println(key + “\t” + val);
	}
	}		

7	What is the output of the below program?
	import java.io.*;
	class BRRead
	{
	public static void main(String args[]) throws IOException
	{
	char c;
	{
	BufferedReader br= new BufferedReader(new InputStreamReader (System.in));
	System.out.println(“enter characters, „q‟ to quit.”);
	Do
	{
	c= (char) br.read();
	System.out.println(c);
	While(c != “q‟);
	}
	}
	}		

8	Analyze the program and give the output and also explain the importance of **hasNext() and next() methods.**
	import java.util.*;
	class
	Findlinedemo
	{
	public static void main(String args[])
	{
	String s=”Name: Tom Age: 28 Id:
	77”; Scanner sc= new Scanner(s);
	sc.findInLine (“Age”);
	if(sc.hasNext())
	Syste.out.println(sc.next())
	; else
	Syste.out.println(“error”);
	}
	}		

9	What is the output of the below
	program? import java.io.*;
	class HTdemo
	{
	public static void main(String args[])
	{
	Hashtable< String, Double> balance= Hashtable< String, Double>();
	Enumeration<String> names;
	String str;
	double
	bal;
	balance.put(“abc” 3435.35);
	balance.put(“abc” 1254.35);
	names= balance.keys();
	while(names.hasMoreElements()
	)
	{
	Str= names.nextElement();
	System.out.println(str + “ : “ + balance.get(str);
	}
	System.out.println();
	}		

10	**Analyze** the below program and give the output of the following program?
	import
	java.util.Scanner; class
	Division {
	public static void main(String[] args)
	{ int a, b, result;
	Scanner input = new
	Scanner(System.in);
	System.out.println("Input two integers");
	a = input.nextInt();
	b =
	input.nextInt();
	result = a / b;
	System.out.println("Result = " + result);
	}
	}		

UNIT - V

PART – A (Short Answer Questions)

1	**Define** AWT class hierarchy?
2	**Distinguish** between swings Vs AWT?
3	**Explain** the hierarchy for swing?
4	**List** the components of swing.
5	**What** is an event?
6	**Define** Applet.
7	**What** are the event sources?
8	**Define** Jbutton, JLabel, JTextField and JtextArea?
9	**Define** Layout management?
10	**List** Layout manager types – border and grid flow?
11	**Explain** about Events, Event sources, Event classes
12	**Explain** about Event Listeners.

13	**Compare** and contrast the relationship between Event sources and Listeners?		
14	**Define** Delegation event model?		
15	**Explain** various events for handling a button click?		
16	**Explain** various events for handling mouse events?		
17	**Define** adapter class?		
18	**Distinguish** between applet and application?		
19	**List** the methods of the applet		
20	**Describe** applet security issues?		
PART – B (Long Answer Questions)			
1	**Explain** in detail about the classification of swing components.		
2	**Explain** in detail about AWT class hierarchy.		
3	**Discuss** in detail about types of Layout management?		
4	**Develop** a java program for handling a button clicks?		
5	**Develop** a java program to illustrate mouse related events?		
6	**Explain** in brief about Events, Event sources and Event classes.		
7	**What** is event listener? Explain.		
8	**Develop** a java program for simple applet?		
9	**Develop** a java program for passing parameters to applet?		
10	**Develop** an applet that receives an integer in one text field and computes its factorial value and returns it in another text field, when the button named “compute” is clicked		
11	**Explain** briefly about Adapter classes.		
12	**What** is the importance of Delegation Event Model on Event Handling?		
13	**Distinguish** various differences between Swing and AWT.		
14	**Compare** and **Contrast** differences between applets and applications		
15	**What** is Applet? Write in detail about life cycle of an applet.		
16	**What** are various security issues of applet.		

1	**Predict** the output using below code and what happens if below code is not used in GUI programs?
	public void windowClosing(WindowEvent e)
	{ dispose();
	System.exit(0);
	}		

2	Explain the below code and mention its important
	Public void actionPerformed(ActionEvent e)
	{
	it(e.getSource()== b1)---Assume b1 is variable of button
	{
	int x= Integer.parseInt(t1.getText()) ---- Assume t1 is variable of textfield
	int y= Integer.parseInt(t2.getText()) ---- Assume t2 is variable of textfield
	int sum= X+Y;
	t3.setText(“ ”+sum); ---- Assume t3 is variable of textfield
	}
	}		

3	What is the output the program
	import java.awt.*;
	import java.swing.*;
	/* < applet code= “Jlabeldemo” width= 200 height =200>
	</applet>
	*/
	public class JLabeldemo extends JApplet
	{
	ImageIcon i= new ImageIcon(“ india.gif ”);
	JLabel ji=new JLabel(“ INDIA” , I , JLabel.CENTER);
	add(ji);
	}
	}		

4	**Explain** the output of the following program?
	import java.applet.*;
	import java.awt.*;
	
	public class Main extends Applet{
	public void paint(Graphics g){
	g.drawString("Welcome in Java Applet.",40,20);
	}
	}
	<HTML>
	<HEAD>
	</HEAD>
	<BODY>
	<div >
	<APPLET CODE="Main.class" WIDTH="800" HEIGHT="500">
	</APPLET>
	</div>
	</BODY>
	</HTML>		

5	**Explain** the usage of the following code?
	public void actionPerformed(ActionEvent ae){
	try{
	num = Integer.parseInt(input.getText());
	sum = sum+num;
	input.setText("");
	output.setText(Integer.toString(sum));
	lbl.setForeground(Color.blue);
	lbl.setText("Output of the second Text Box : "
	+ output.getText());
	}
	catch(NumberFormatException e){
	lbl.setForeground(Color.red);
	lbl.setText("Invalid Entry!");
	}		

6	**Analyze** the program and explain the importance 1 to 4 line and give the output.
	
	import java.awt.*;
	class Frame1 extends Frame
	{
	Frame1()
	{
	setTitle(“demo”); -----------1
	setSize(200,200);
	setVisible(true); ------------2
	setLayout(new FlowLayout());
	Label l1= new Label(“java”);
	Label l2= new Label(“j2ee”);
	add(l1); ---------------3
	add(l2); ---------------4
	}
	}
	Class Labeldemo
	{
	Public static void main(String args();
	{
	Frame1 f= new Frame();
	}
	}		

7	What is the output the program
	import java.awt.*;
	import java.applet.*;
	/* < applet code= “statusdemo.class” width= 200 height =200>
	</applet>
	*/
	public class satusdemo extends Applet
	{
	Public void init()
	{
	setBackground(Color.red);
	}
	Public void paint(Graphics g)
	{
	g.drawString(“this is in the applet window” 10,20)‟
	showStatus(“this is the status window message”);
	}
	}		

8	Explain the usage of following line of code line by line.
	Public void mouseClicked(MouseEvent me)
	{
	Mousex-=0;
	Mousey=10;
	Msg= “mouse clicked”
	Repaint();
	}
	Public void mouseEntered(MouseEvent me)
	{
	Mousex-=0;
	Mousey=10;
	Msg= “mouse entered”
	Repaint();
	}		

9	What is the output of the program
	import java.applet.Applet;
	import java.awt.*;
	public class Sms extends Applet {
	public void init() {
	}
	public void paint(Graphics g) {
	g.setColor(Color.blue);
	Font font = new Font("verdana", Font.BOLD, 15);
	g.setFont(font);
	g.drawString("Welcome To Aeronautical Eng College", 50, 50);
	}
	}		

10	What is the output the program and explain line 1 and 2.
	import java.awt.*;
	import java.applet.*;
	/* < applet code= “GridLayoutDemo” width= 300 height =300>
	</applet>
	*/
	public class GridLayoutDemo extends Applet
	{
	static final int n=5;
	public void init()
	{
	setLayout(new GridLayout(n , n)); ------- 1
	setFont (new Font (“SamsSerof”, Font.BOLD, 24)); ---- 2
	for (int j=0l j<n; j++)
	{
	int k= I * n + j;
	if(k>00)
	Add(new button (“” + k0;
	}}}}		