1.CLASS AND OBJECTS

```
import java.util.*;
class SavingsAccount
{
byte bankcityid;
short bankbranchid;
long accid;
String accname;
char acctype;
double balance;
float interest;
boolean active;
public void openAccount()
{
Scanner in = new Scanner(System.in);
System.out.println("Enter Bank City Id");
bankcityid = in.nextByte();
System.out.println("Enter Bank Branch Id");
bankbranchid = in.nextShort();
System.out.println("Enter Account Id");
accid = in.nextLong();
System.out.println("Enter Account Name");
accname = in.next();
System.out.println("Enter Account Type 'c' or 's'");
acctype = in.next().charAt(0);
System.out.println("Enter Initial Amount Deposited to account");
balance = in.nextDouble();
System.out.println("Enter Interest rate");
interest = in.nextFloat();
System.out.println("Enter true to make the account active");
active = in.nextBoolean();
}
public void deposit(double amount)
{
balance = balance+amount;
}
public void withdraw(double amount)
{
double checkamt;
checkamt = balance-amount;
if(checkamt<=1000)
{
System.out.println("As per Bank Rule you should maintain minimum balance of Rs 1000
\n\n\n");
```

```
}
else
{
System.out.println("Processing withdrawal.....\n\n\n");
balance = balance-amount;
}
}
public void addInterest()
{
balance = balance+balance*interest;
}
public double getBalance()
{
return balance;
}
public void printAccount()
{
System.out.println("Bank City id:"+bankcityid);
System.out.println("Bank branch id:"+bankbranchid);
System.out.println("Account id:"+accid);
System.out.println("Account name:"+accname);
System.out.println("Account type 'c' current or 's' savings:"+acctype);
System.out.println("Balance amount:"+String.format("%,2f",balance));
System.out.println("Interest rate:"+interest);
}
}
public class SavingsAccountTester
{
public static void main(String[]args)
{
SavingsAccount a1 = new SavingsAccount();
a1.openAccount();
a1.deposit(40000);
a1.withdraw(7000);
a1.printAccount();
a1.addInterest();
System.out.println("current balance after adding
interest:"+String.format("%,2f",a1.getBalance()));
}
}
```

OUTPUT:

```
C:\Java\bin>javac SavingsAccountTester.java

C:\Java\bin>java SavingsAccountTester
Enter Bank City Id
123
Enter Bank Branch Id
4677
Enter Account Id
4906547432
Enter Account Name
keethu
Enter Account Type 'c' or 's'
c
Enter Initial Amount Deposited to account
25000
Enter Interest rate
5
Enter true to make the account active
true
Processing withdrawal.....


Bank City id:123
Bank branch id:4677
Account id:4906547432
Account name:keethu
Account type 'c' current or 's' savings:c
Balance amount:58,000.000000
Interest rate:5.0
current balance after adding interest:348,000.000000
```

2. CONTROL STATEMENTS

```
import java.util.*;
class Stu
{
   int reg,i;
   String name;
   int m[]=new int[5];
   int total;
   int avg;
   String grade;
void getdata()
{
   Scanner in=new Scanner(System.in);
   System.out.println("enter register number:");
    reg=in.nextInt();
   System.out.println("enter name:");
    name=in.next();
   System.out.println(" enter 5 subjects marks obtained");
for(i=0;i<5;i++)
{
  m[i]=in.nextInt();
}
}
void calculate()
{
   for(i=0;i<5;i++)
 {
  total= total+m[i];
 }
  avg=total/5;
if(avg>=75)
grade="first class with distinction";
else if((avg>=65) && (avg<75))
grade="first class";
else if((avg>=50) &&(avg<65))
grade="second class";
else if((avg>=40) && (avg<50))
grade=" third class";
else
grade=" fail";
}
void printdata()
{
System.out.println(" Name of the student:" +name);
```

```
System.out.println(" Register number:" +reg);
System.out.println("   marks obtained:  ");
 for(i=0;i<5;i++)
 {
   System.out.println( "  marks "  +i+  ":"  +m[i]);
 }
System.out.println(" total: " +total);
System.out.println(" average: " +avg);
System.out.println(" grade: " +grade);
}
}
class Stu2
{
public static void main(String args[])
{
Stu a1=new Stu();
a1.getdata();
a1.calculate();
a1.printdata();
}
}
```

OUTPUT:

```
Z:\>javac Stu2.java
Z:\>java Stu2
enter register number:
23
enter name:
shangavi
enter 5 subjects marks obtained
34
66
77
88
99
Name of the student:shangavi
Register number:23
enter marks obtained:
marks0:34
marks1:66
marks2:77
marks3:88
marks4:99
total:364
average:72
grade:first class
```

```
import java.util.Scanner;
```

3.inheritance

```
import java.util.Scanner;
import java.lang.Math;

class Calc

{

static int a,b;

static Scanner sc =new Scanner(System.in);
static void input()

{ System.out.println("Enter a no:");



a=sc.nextInt();
System.out.println("Enter a no:");

b= sc.nextInt();
}
static void input2()
{
System.out.println("Enter a no:");
a=sc.nextInt();}
static int add(int a,int b)

{ return a+b;}

static int sub(int a,int b)

{ return a-b;}

static int mul(int a,int b)

{ return a *b;
}


static int div(int a,int b)

{ return  a/b;}

static int sqr(int a)

{ return a*a;
}
}




class Calculator extends Calc
{
```

```
static double c;

static void input3() {


System.out.println("Enter a no: ");

c=sc.nextDouble();
}
static double sine(double d)
{
return Math.sin(Math.toRadians(d));
}
static double cosine(double d)

{

return Math.cos(Math. toRadians(d));
}
static double tann(double d)
{
return Math.tan(Math.toRadians(d));
}
static double asine(double d)

{

return Math.asin(d);
}
static double acosine(double d)
{
return Math. acos(d);
}
static double atann(double d)
{
return Math .atan(d);
}
public static void main (String args)
{
        System.out.println("----Menu-");

                        int m=sc.nextInt();

                        switch(m)

                        {

                        case 1:
                        input();

                        System.out.println("\n"+
                        add(a,b));
        break;
                        case 2:
```

```
input();
System.out.println("\n"+a+"-"+b+"="+sub(a,b));

break;

case 3:
input();

System.out.println("\n" +a+""+b+"="+mul(a,b));


input();

System.out.println("\n"+a+""+b+""+div(a,b));

break;

case 5:
input2();

System.out.println("\nSquare of "+a+""+sqr(a));

break;


        case 6:

        input3(); System.out.println("\nSine("+c+"="+sine(c));

        break;

        case 7:

        input3(); System.out.println("\nCos("+c+"="+cosine(c));

        break;
        case 8:

        input3(); System.out.println("\nTan("+c+")="+tann(c));

        break;

        case 9:

        input3(); System.out.println("\nAsin("+c+")="+asine(c));

        break;
        case 10:

        input3();

        System.out.println("\nACos("+c+")="+acosine(c));
        break;

        case 12:
```

```
                    input3();
                    System.out.println("\nAtan("+c+")"+atann(c));

                    break;

                    default:

                    System.out.println("Wrong Entry");
                }
}
}
```

4. USING MOUSE EVENTS

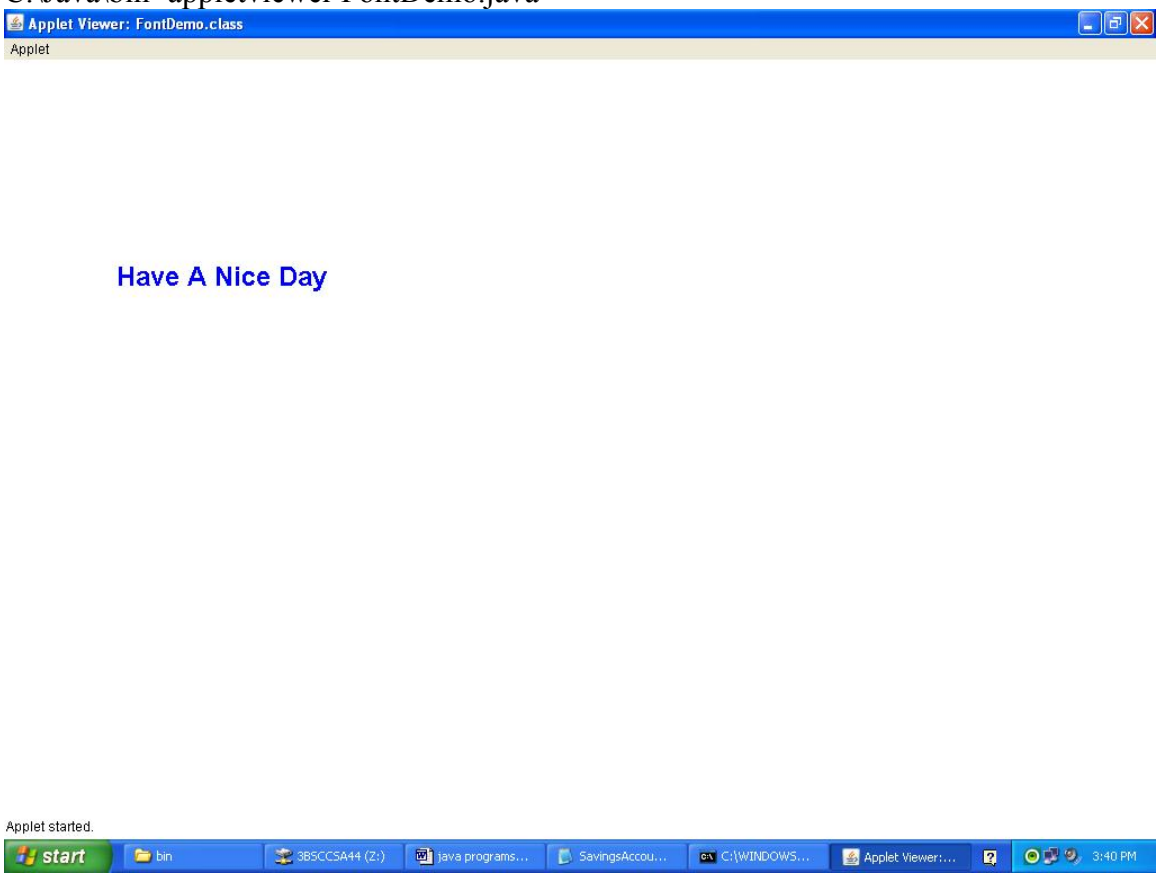
```
import java.awt.*;
import java.awt.event.*;
import java.awt.Color;
 public class FontDemo extends java.applet.Applet
{
Font f;
int cnt=0;
int size=8;
public void init()
{
addMouseListener(new MouseAdapter()
{
public void mouseClicked(MouseEvent me)
{
cnt++;
repaint();
}
});
}

public void paint(Graphics g)
{
f=new Font("Arial",Font.BOLD,size);
if(cnt<25)
{
size+=2;
f=new Font("Arial",Font.BOLD,size);
g.setFont(f);
g.setColor(Color.blue);
g.drawString("Have A Nice Day",100,200);
}
else
{
```

```
g.drawString("Mouse Clicked More Than 25 times",150,250);
}
}
}
/*<applet code="FontDemo.class" width=400 height=400>
</applet>*/
```

OUTPUT:

C:\Java\bin>javac FontDemo.java

C:\Java\bin>appletviewer FontDemo.java



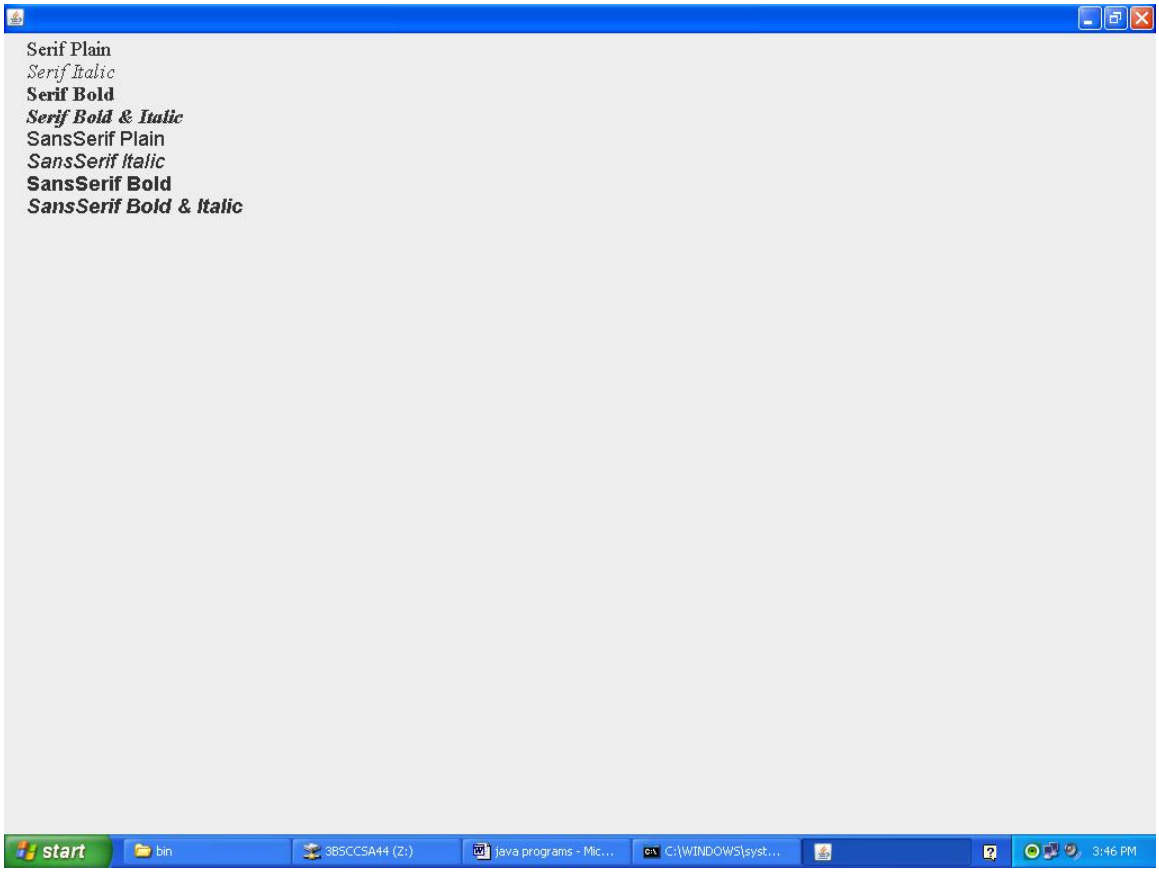
5. IMPLEMENTING FONT CLASS METHOD

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class Myfont extends JPanel
{
String[] type= { "Serif", "SansSerif" } ;
int[] styles={Font.PLAIN, Font.ITALIC, Font.BOLD,Font.ITALIC+ Font.BOLD};
String[] stylenames= {"Plain", "Italic", "Bold", "Bold & Italic"};
public void paint(Graphics g) {
for(int f=0;f<type.length;f++) {
for (int s=0;s<styles.length;s++) {
Font font=new Font(type[f],styles[s],18);
g.setFont(font);
String name=type[f]+" "+stylenames[s];
g.drawString(name,20,(f*4+s+1)*20);
}
}
}
public static void main(String[] a) {
JFrame f=new JFrame();
f.addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent e) {
System.exit(0);
}

}
);
f.setContentPane(new Myfont());
f.setSize(400,400);
f.setVisible(true);
}
}
```

OUTPUT:

C:\Java\bin>javac Myfont.java

C:\Java\bin>java Myfont



6. EXCEPTION HANDLING

```
import java.io.*;
class InsufficientFundsException extends Exception
{
private double amount;
public InsufficientFundsException(double amount)
{
this.amount=amount;
}
public double getAmount()
{
return amount;
}
}

class CheckingAccount
{
private double balance;
private int number;
public CheckingAccount(int number)
{
this.number=number;
}

public void deposit(double amount)
{
balance+=amount;
}
public void withdraw(double amount)throws InsufficientFundsException
{
if(amount<=balance)
{
balance-=amount;
```

```
}
else
{
double needs=amount-balance;
throw new InsufficientFundsException(needs);
}
}
public double getBalance()
{
return balance;
}
public int getNumber()
{
return number;
}
}

public class BankDemo
{
public static void main(String []args)
{
CheckingAccount c=new CheckingAccount(101);
System.out.println("depositing $500....");
c.deposit(500.00);
try
{
System.out.println("\n Withdrawing $100...");
c.withdraw(100.00);
System.out.println("\n Withdrawing $600...");
c.withdraw(600.00);
}
catch(InsufficientFundsException e)
{
System.out.println("Sorry,but you are short $"+e.getAmount());
e.printStackTrace();
}
}
}
```

OUTPUT:

```
C:\Java\bin>javac BankDemo.java

C:\Java\bin>java BankDemo
depositing $500....

 Withdrawing $100...

 Withdrawing $600...
Sorry,but you are short $200.0
InsufficientFundsException
        at CheckingAccount.withdraw(BankDemo.java:37)
        at BankDemo.main(BankDemo.java:62)
```

7.USING INTERFACE

```
public interface Electricitybill {
void calculate(int units);
}
public class Electricitybill1 implements Electricitybill
{
        public void calculate(int units)
        {
                double billpay=0;
                if(units<100) {
                        billpay = units*1.20;
                }
                else if(units<300) {
                        billpay=100*1.20+(units-100)*2;
                }
                else if(units>300) {
                        billpay=100*1.20+200*2+(units-300)*3;
                }
                else {
                        billpay=0;
                }
                System.out.println("units:"+units);
                System.out.println("bill to pay:"+billpay);
        }
}
public class Testinterface{
        public static void main(String args[]) {
                Electricitybill1 a =new Electricitybill1();
                a.calculate(280);
        }
}
```

OUTPUT:

units:280
bill to pay:480.0

8. ILLUSTRATING THREAD PRIORITY

```
import java.lang.*;
class A extends Thread
```

```
{
public void run()
{
System.out.println("threadA started");
for(int i=1; i<=4; i++)
{
System.out.println("\tFrom Thread A : i = "+i);
}
System.out.println("Exit from A");
}
}
class B extends Thread
{
public void run()
{
System.out.println("threadB started");
for(int j=1; j<+4; j++)
{
System.out.println("\tFrom Thread B : j = "+j);
}
System.out.println("Exit from B");
}
}
class C extends Thread
{
public void run()
{
System.out.println("threadC started");
for(int k=1; k<=4; k++)
{
System.out.println("\tFrom Thread C : k = " +k);
}
System.out.println("Exit from c");
}
}
class ThreadPriority
{
public static void main(String args[])
{
 A threadA = new A();
 B threadB = new B();
 C threadC = new C();

 threadC.setPriority(Thread.MAX_PRIORITY);
 threadB.setPriority(threadA.getPriority()+1);
 threadA.setPriority(Thread.MIN_PRIORITY);
```

```
 System.out.println("start thred A");
 threadA.start();
 System.out.println("start thread B");
 threadB.start();
 System.out.println("start thread C ");
 threadC.start();
 System.out.println("end of main thread");
}
}
```

OUTPUT:

```
C:\Java\bin>javac ThreadPriority.java

C:\Java\bin>java ThreadPriority
start thred A
start thread B
threadA started
threadB started
        From Thread B : j = 1
        From Thread B : j = 2
        From Thread B : j = 3
Exit from B
start thread C
end of main thread
threadC started
        From Thread C : k = 1
        From Thread C : k = 2
        From Thread C : k = 3
        From Thread C : k = 4
Exit from c
        From Thread A : i = 1
        From Thread A : i = 2
        From Thread A : i = 3
        From Thread A : i = 4
Exit from A
```