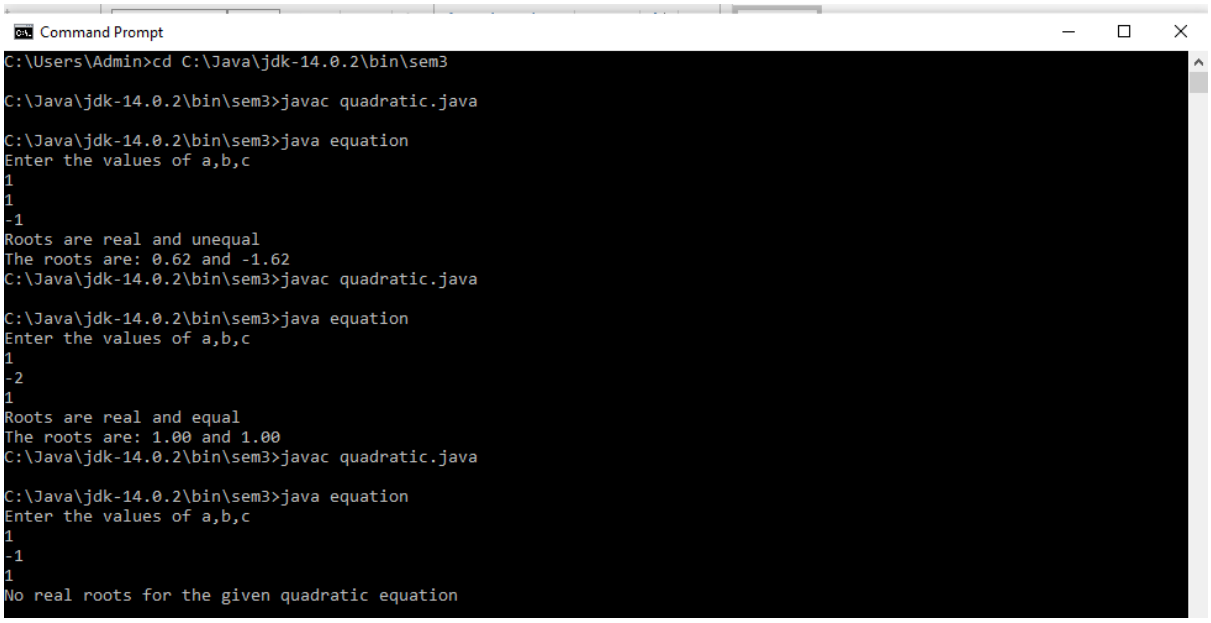
Program 1

Develop a Java program that prints all real solutions to the quadratic equation ax2 +bx+c = 0. Read in a, b, c and use the quadratic formula. If the discriminate b2-4ac is negative, display a message stating that there are no real solutions.

```
import java.util.Scanner;
class equation
{
   public static void main(String args[])
   {
      Scanner sc=new Scanner(System.in);
      double a,b,c,r1,r2,d;
      System.out.println("Enter the values of a,b,c");
      a=sc.nextDouble();
      b=sc.nextDouble();
      c=sc.nextDouble();
      d=(b*b)-(4*a*c);
      if(d<0)
         System.out.println("No real roots for the given quadratic equation");
      else if(d>=0)
      {
         r1=(-b+(Math.sqrt(d)))/(2*a);
         r2=(-b-(Math.sqrt(d)))/(2*a);
         if(d==0)
         {
            System.out.println("Roots are real and equal");
            System.out.printf("The roots are: %.2f and %.2f",r1,r2);
         }
         else
         {
```

```
            System.out.println("Roots are real and unequal");
            System.out.printf("The roots are: %.2f and %.2f",r1,r2);
        }
    }
  }
}
```



```
import java.util.*;
import java.lang.Math;
class quadratic
{
  public static void main (String args[])
  {
    Scanner sc = new Scanner(System.in);
    int a, b, c;
    float r, r1, r2, x, d;
    System.out.println("Enter the coefficients of
    the equation");
    a = sc.nextInt();
    b = sc.nextInt();
    c = sc.nextInt();
    d = ((b^2) -(4*a*c));
    if (d<0)
    {
      system.out.println ("The roots are negative");
      d=d*(-1);
      r = (Math.sqrt(d)/(2*a));
      x = -b/(2*a);
      system.out.println("The roots are"+x+
      "+"+r+"and"+x+"-"+r);
    }
  else if (d==0)
    {
      system.out.println("The roots are real
      and equal");
      r = -b/(2*a);
      system.out.println("The roots are"
      +r+"and"+r);
    }
```

```
else
{
System.out.println("The roots are real
and equal");
r1 = (-b + (Math.sqrt(d)))/2*a);
r2 =(-b-(Math.sqrt(d)))/2*a);
System.out.println("The roots are"
+r1+"and"+r2);
}
}
}
```

Program 2

Develop a Java program to create a class Student with members usn, name, an array credits and an array marks. Include methods to accept and display details and a method to calculate SGPA of a student.

```
import java.util.Scanner;
class Student
{

  String usn,name;
  int credits[];
  float marks[];
  int n;
  float tot=0;
  Student()
  {
            Scanner sc=new Scanner(System.in);
            System.out.println("Enter the number of subjects");
            n=sc.nextInt();
            credits=new int[n];
            marks=new float[n];
            usn="";
            name="";
  }
   void Accept()
   {
             Scanner sc=new Scanner(System.in);
          System.out.println("Enter your USN and Name");
      usn=sc.nextLine();
      name=sc.nextLine();
      System.out.println("Enter credits and marks for each subject");
      for(int i=0;i<n;i++)
                      {
        credits[i]=sc.nextInt();
        marks[i]=sc.nextFloat();
                      }
          }
   void Calculate()
          {
            int s=0;
            int m=0;
```

```
            float a=0;


    for(int i=0;i<n;i++)
            {
     if(marks[i]>=90)
                                m=10;
                    else if( marks[i]>=80)
                                m=9;
                    else if( marks[i]>=70)
                                m=8;
                    else if( marks[i]>=60)
                                m=7;

              else if( marks[i]>=50)
                                m=6;
                     else if( marks[i]>=40)
                                m=4;


                    else if(marks[i]<40)
                                m=0;

      s=s+credits[i];
                    a=a+(credits[i]*m);

             }
             tot=a/s;
           }
           void Display()
           {
                    System.out.println("The details of the student");
              System.out.println("USN:"+usn+"   Name:"+name);
                    System.out.println("Credits and Marks");
                     for(int i=0;i<n;i++)
                    {
       System.out.println(credits[i]+"  "+marks[i]);

                    }
                    System.out.printf("SGPA  %.2f",tot);
           }
}
class StudentMain
{
           public static void main(String args[])
```

```
        {
                Student s1=new Student();
                s1.Accept();
                s1.Calculate();
                s1.Display();
        }
}
```

```
SGPA  9.15
C:\Java\jdk-14.0.2\bin\sem3>javac sgpa.java

C:\Java\jdk-14.0.2\bin\sem3>java StudentMain
Enter the number of subjects
4
Enter your USN and Name
1bm19cs011
akshatha
Enter credits and marks for each subject
3
89
4
88
3
87
4
98
The details of the student
USN:1bm19cs011   Name:akshatha
Credits and Marks
3  89.0
4  88.0
3  87.0
4  98.0
SGPA  9.29
C:\Java\jdk-14.0.2\bin\sem3>
```

```
import java.util.Scanner;
class Student
 {
   String usn, name;
   int credits[];
   float marks[];
   int n;
   float tot=0;
   Student
    {
      Scanner usc = new Scanner (System.in);
      System.out.println ("Enter the number
      of subjects");
      n = usc.nextInt()
      credits = new int[n];
      marks = new float[n];
      usn = " ";
      name = " ";
    }
   void Accept()
    {
      Scanner usc = new Scanner (System.in);
      System.out.println("Enter your USN and
      Name");
      usn = usc.nextInt();
      name = usc.nextLine();
      System.out.println("Enter credits and
      marks for each subject");
      for (int i=0; i<n; i++)
        {
          credits[i] = usc.nextInt();
          marks[i] = usc.nextFloat();
```

```
    }
  }
void Calculate()
  {
    int s=0;
    int m=0;
    float a=0;
    for (int i=0; i<n; i++)
    {
        if (marks[i]>=90)
            m=10;
        else if (marks[i]>=80)
            m=9;
        else if (marks[i]>=70)
            m=8
        else if (marks[i]>=60)
            m=7
        else if (marks[i]>=50)
            m=6.
        else if (marks[i]>=40)
            m=4;
        else if (marks[i]<40)
            m=0;
        s = s+ credits[i];
        a = a+ (credits[i]*m);
    }
    tot = a/s;
  }
void Display()
  {
    System.out.println("The details of the student");
```

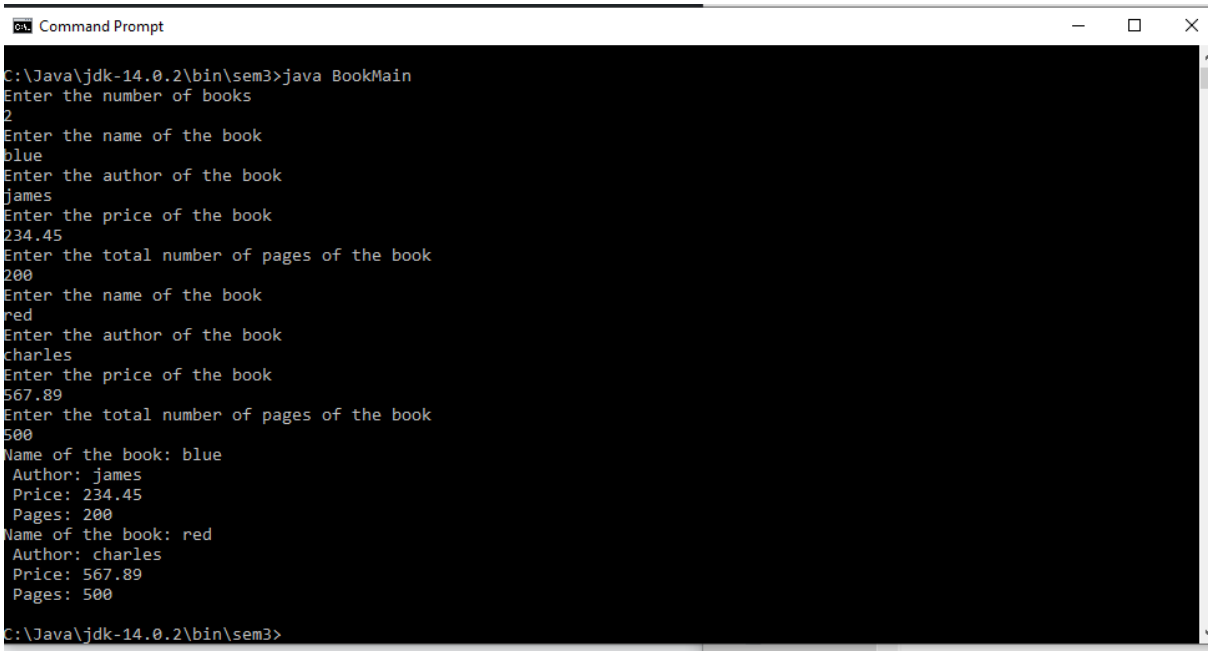
```
        System.out.println("USN."+usn+"  Name"+name);
        System.out.println("Credits and Marks");
        for(i=0;i<n;i++)
        {
            System.out.println(credits[i]+"  "+marks[i]);
        }
        System.out.printf("SGPA %.2f",tot);
    }
}
class StudentMain
{
    public static void main(String args[])
    {
        Student s1 = new Student();
        s1.Accept();
        s1.Calculate();
        s1.Display();
    }
}
```

Program 3

Create a class Book which contains four members: name, author, price, num_pages. Include a constructor to set the values for the members. Include methods to set and get the details of the objects. Include a toString( ) method that could display the complete details of the book. Develop a Java program to create n book objects.

```
import java.util.Scanner;
class Book
{
	String name,author;
	int pages;
	float price;
	Book()
	{
		name="";
		author="";
		pages=0;
		price=0.0f;
	}
	void Accept()

	{
		Scanner sc=new Scanner(System.in);
		System.out.println("Enter the name of the book");
		name=sc.nextLine();
		System.out.println("Enter the author of the book");
		author=sc.nextLine();
		System.out.println("Enter the price of the book");
		price=sc.nextFloat();
		System.out.println("Enter the total number of pages of the book");
		pages=sc.nextInt();
	}
	public String toString()
	{
		return("Name of the book: "+name+"\n Author: "+author+"\n Price:
"+price+"\n Pages: "+pages);
	}
}
class BookMain
{
	public static void main(String args[])
	{
		int n;
```

```
        Scanner sc=new Scanner(System.in);
        System.out.println("Enter the number of books");
        n=sc.nextInt();
        Book b[]=new Book[n];
        for(int i=0;i<n;i++)
        {
                b[i]=new Book();
                b[i].Accept();
        }
        for(int i=0;i<n;i++)
                System.out.println(b[i]);
        }
}
```



```
import java.util.Scanner;
class Book
 {
   String name, author;
   int pages;
   float price;
   Book()
    {
      name = "";
      author = "";
      pages = 0;
      price = 0.0f;
    }
   void Accept()
    {
      Scanner sc = new Scanner (System.in);
      System.out.println("Enter the name of the book");
      name = sc.nextLine();
      System.out.println("Enter the author of the book");
      author = sc.nextLine();
      System.out.println("Enter the price of the book");
      price = sc.nextFloat();
      System.out.println("Enter the total number of
      pages of the book");
      pages = sc.nextInt();
    }
   public String toString()
    {
      return ("Name of the book:"+name+" Author:"
      +author+"Price:"+price+"Pages:"+pages);
    }
```

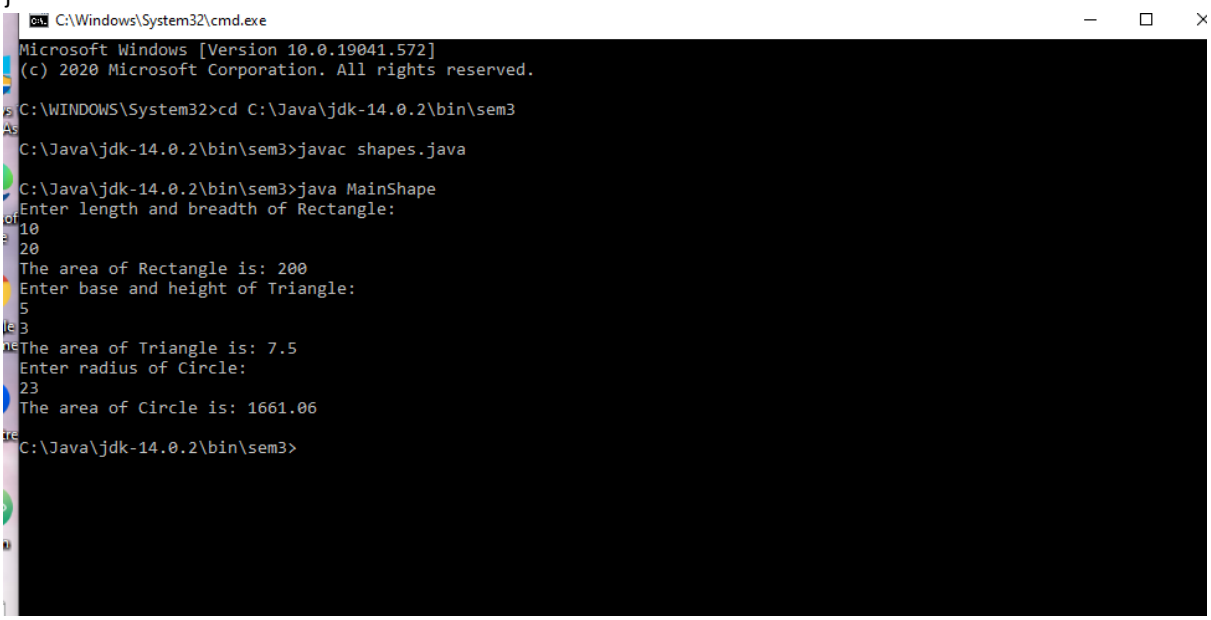
```
class Book.Main
{
    public static void main (String args [])
    {
        int n;
        Scanner sc = new Scanner (System.in);
        System.out.println ("Enter the number of
            books");
        n = sc.nextInt();
        Book b[] = new Book[n];
        for (int i=0; i<n; i++)
        {
            b[i] = new Book ();
            b[i]. Accept();
            System.out.println (b[i]);
        }
    }
}
```

Program 4

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 contain only the method printArea( ) that prints the area of the given shape.

```
import java.util.Scanner;
abstract class Shape
{
   int a1,a2;
   Scanner sc = new Scanner(System.in);
        abstract void printArea();

}

class Rectangle extends Shape
{
   void printArea()
        {
                System.out.println("Enter length and breadth of Rectangle: ");
                a1 = sc.nextInt();
                a2 = sc.nextInt();
                System.out.println("The area of Rectangle is: "+a1*a2);
        }
}

class Triangle extends Shape
{
        void printArea()
        {
                System.out.println("Enter base and height of Triangle: ");
                a1 = sc.nextInt();
                a2= sc.nextInt();
                System.out.println("The area of Triangle is: "+(a1*a2)/2f);
        }
}

class Circle extends Shape
{
        void printArea()
        {
```

```
                System.out.println("Enter radius of Circle: ");
                a1 = sc.nextInt();
                System.out.println("The area of Circle is: " +a1*a1*3.14f);
        }
}

class MainShape
{
        public static void main(String args[])
        {
                Rectangle r = new Rectangle();
                r.printArea();
                Triangle t = new Triangle();
                t.printArea();
                Circle c = new Circle();
                c.printArea();
        }
}
```



```
import java.util.Scanner;
abstract class Shape
{
  int n1, n2;
  Scanner sc = new Scanner(System.in);
  abstract void printArea()
}
class Rectangle extends Shape
{
  void printArea()
  {
    System.out.println("Enter length and breadth
    of Rectangle:");
    a1 = sc.nextInt();
    a2 = sc.nextInt();
    System.out.println("The area of rectangle is"
      +a1*a2);
  }
}
class Triangle extends Shape
{
  void printArea()
  {
    System.out.println("Enter base and height of
    Triangle");
    a1 = sc.nextInt();
    a2 = sc.nextInt();
  }
}
class Circle extends Shape
{
  void printArea()
```

```
    {
        System.out.println("Enter the radius of
            circle:");
        a1 = sc.nextInt();
        System.out.println("Area of circle is:"+a1*
                                                3.14);
    }
}
class MainShape
{
    public static void main(String args[])
    {
        Rectangle r = new Rectangle();
        r.printArea();
        Triangle t = new Triangle();
        t.printArea();
        Circle c = new Circle();
        c.printArea();
    }
}
```

Program 5

Develop a Java program to create a class Bank that maintains two kinds of account for its customers, one called savings account and the other current account. The savings account provides compound interest and withdrawal facilities but no cheque book facility. The current account provides cheque book facility but no interest. Current account holders should also maintain a minimum balance and if the balance falls below this level, a service charge is imposed. Create a class Account that stores customer name, account number and type of account. From this derive the classes Curr-acct and Sav-acct to make them more specific to their requirements. Include the necessary methods in order to achieve the following tasks: • Accept deposit from customer and update the balance. • Display the balance. • Compute and deposit interest • Permit withdrawal and update the balance • Check for the minimum balance, impose penalty if necessary and update the balance

```
import java.util.Scanner;
class Account
{
	String name,type;
	long acno;
	double bal;
	double minbal=1000.0;
	double w=0;
	Account(String name,String type,long acno,double bal)
	{
		this.name=name;
		this.type=type;
		this.acno=acno;
		this.bal=bal;
	}
		Scanner sc=new Scanner(System.in);

}


class Current extends Account
{
	Current (String name,long acno,double bal)
	{
		super(name,"Current",acno,bal);
	}
	void Withdraw()
	{
```

```
                System.out.println("Enter the amount you want to withdraw");
                w=sc.nextDouble();
                bal=bal-w;
                Balance();

        }
        void Deposit()
        {

                System.out.println("Enter the amount you want to deposit");
                w=sc.nextDouble();
                bal=bal+w;

        }
        void Balance()
        {
                if (bal<minbal)
                {
                        System.out.println("Insufficient balance ,penalty will be imposed");
                   bal=bal*0.3;
                }



        }
        void Display()
        {

                System.out.println("Name"+name+"\n Account number"+acno+"\n Type of
account"+type+"\nBalance"+bal);
        }
}
class Savings extends Account
{
        Savings (String name,long acno,double bal)
        {
                super(name,"Savings",acno,bal);
        }

        void Withdraw()
        {
                System.out.println("Enter the amount you want to withdraw");
                w=sc.nextDouble();
                bal=bal-w;
```

```
	}
	void Deposit()
	{
		System.out.println("Enter the amount you want to deposit");
		w=sc.nextDouble();
		bal=bal+w;
		Calculate();
	}
	void Calculate()
	{
		int t=2, R=55;
		bal=bal+bal*(Math.pow((1+(R/100)), t));
	}
	void Display()
	{

		System.out.println("Name"+name+"\n Account number"+acno+"\n Type of
account"+type+"\nBalance"+bal);
	}
}
class MainAccount
{
	public static void main(String args[])
	{
		Scanner sc=new Scanner(System.in);

	   System.out.println("Enter your name");
		String name=sc.nextLine();
		System.out.println("Enter your account number");
		long acno=sc.nextLong();
		System.out.println("Enter your account balance");
		float bal=sc.nextFloat();

		System.out.println("Type of account:\n 1.Current account\n 2.Savings
account\n 3.Exit");
			int o=sc.nextInt();
			if(o==1)
			{
				Current c = new Current(name,acno,bal);
				while(true)
				{
					System.out.println("1.Deposit\n2.Withdraw
Amount\n3.Display\n4.Exit");
					int ch = sc.nextInt();
```

```
                                        switch (ch)
                                        {
                                                case 1:
                                                c.Deposit();
                break;
                                                case 2:
                                                c.Withdraw();
                                                break;
                                                case 3:
                                                c.Display();
                                                case 4:
                                                System.exit(0);
                                                default:
                                                System.out.println("Invalid choice");
                                        }
                        }
                }
                else if(o==2)
                {
                        Savings s = new Savings(name,acno,bal);
                        while(true)
                        {
                                System.out.println("1.Deposit\n2.Withdraw
Amount\n3.Display\n4.Exit");
                                int ch = sc.nextInt();
                                switch (ch)
                                        {
                                                case 1:
                                                s.Deposit();
                break;
                                                case 2:
                                                s.Withdraw();
                                                break;
                                                case 3:
                                                s.Display();
                                                case 4:
                                                System.exit(0);
                                                default:
                                                System.out.println("Invalid choice");
                                        }
                        }
                }
                else if(o==3)
                        System.exit(0);
                else
```

```
                System.out.println("Invalid choice");


            }
        }
```

C:\Windows\System32\cmd.exe

```
Microsoft Windows [Version 10.0.19041.572]
(c) 2020 Microsoft Corporation. All rights reserved.

C:\WINDOWS\System32>cd C:\Java\jdk-14.0.2\bin\sem3

C:\Java\jdk-14.0.2\bin\sem3>javac accounts.java

C:\Java\jdk-14.0.2\bin\sem3>java MainAccount
Enter your name
asdf
Enter your account number
123456789
Enter your account balance
1000
Type of account:
 1.Current account
 2.Savings account
 3.Exit
1
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
1
Enter the amount you want to deposit
200
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
2
Enter the amount you want to withdraw
300
Insufficient balance ,penalty will be imposed
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
3
Nameasdf
 Account number123456789
 Type of accountCurrent
Balance270.0
1.Deposit
```

C:\Windows\System32\cmd.exe

```
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
4

C:\Java\jdk-14.0.2\bin\sem3>java MainAccount
Enter your name
asdf
Enter your account number
123456789
Enter your account balance
2000
Type of account:
 1.Current account
 2.Savings account
 3.Exit
2
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
1
Enter the amount you want to deposit
200
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
2
Enter the amount you want to withdraw
200
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
3
Nameasdf
 Account number123456789
 Type of accountSavings
Balance4200.0
1.Deposit
2.Withdraw Amount
3.Display
```

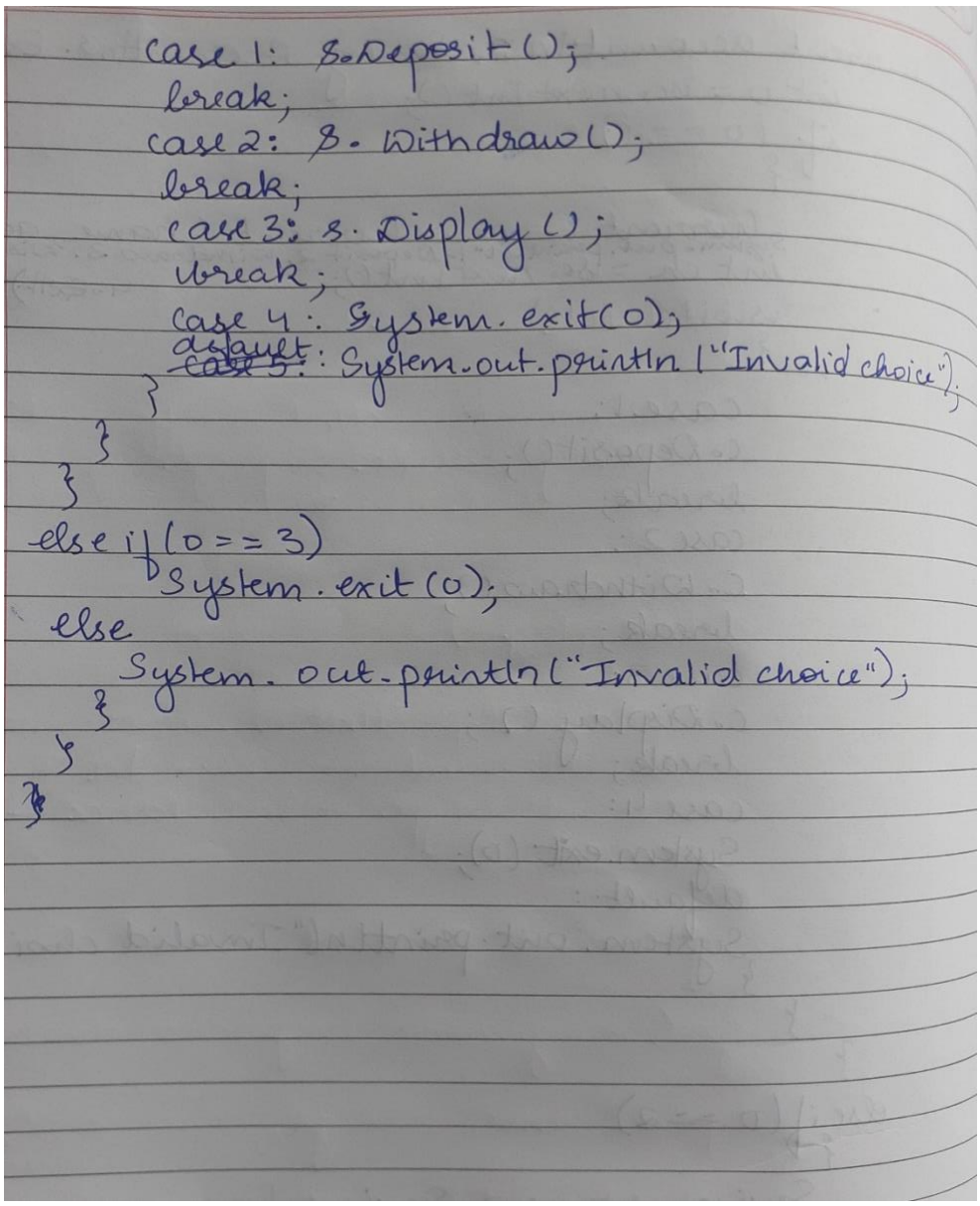
```
Enter the amount you want to withdraw
200
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
3
Nameasdf
 Account number123456789
 Type of accountSavings
Balance4200.0
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
3
Nameasdf
 Account number123456789
 Type of accountSavings
Balance4200.0
1.Deposit
2.Withdraw Amount
3.Display
4.Exit
4

C:\Java\jdk-14.0.2\bin\sem3>
```

```
import java.util.Scanner;
class Account
 {
   String name,type;
   long acno;
   double bal;
   double minbal =1000.0; w=0;
   Account (String name, String type, long acno,
   double bal)
    {
     this.name = name;
     this.type = type;
     this.long = acno;
     this.bal = bal;
    }
    Scanner isc = new Scanner (System.in);
  }
class Current extends Account
{
  Current (String name, long acno, double bal)
   {
     super (name, "current", acno, bal);
   }
   void Withdraw ()
    {
     System.out.println("Enter the amount");
     w = isc.nextDouble();
     bal = bal - w;
     Balance ();
    }
   void Deposit ()
     {
```

```
        System.out.println("Enter the amount");
        wo = sc.nextDouble();
        bal = bal + wo;
    }
    void Balance()
    {
        if (bal < minbal)
        {
            System.out.println("Insufficient balance, penalty will be imposed");
            bal = bal * 0.3;
        }
    }
    void Display()
    {
        System.out.println("Name" + name + "\n Account number" + acno + "\n Type of account" + type + "\n Balance" + bal);
    }
}
class Savings extends Account
{
    Savings(String name, long acno, double bal)
    {
        super(name, "Savings", acno, bal);
    }
    void Withdraw()
    {
        System.out.println("Enter the amount");
        wo = sc.nextDouble();
        bal = bal - wo;
    }
```

```
void ~~Display~~ Deposit()
{
System.out.println("Enter the amount");
w = sc.nextDouble();
bal = bal+w;
calculate();
}
void calculate()
{
int t = 2, r = 55;
bal = bal + bal * (Math.pow((1+(R/100)),
}
void Display()
{
System.out.println("Name"+name+"Ac
number"+acno+"Type of account"+type
"In Balance"+bal);
}
}
class MainAccount
{
public static void main(String args[])
{
Scanner sc = new Scanner(System.in);
System.out.println("Enter your name");
String name = sc.nextLine();
System.out.println("Enter your account
number");
long acno = sc.nextLong();
System.out.println("Enter your balance");
float bal = sc.nextFloat();
System.out.println("Type of account : In
```

```
Current account\n 2. Savings Account\n3. Exit:");
int o = sc.nextInt();
if (o == 1)
{
Current c = new Current (name, acno, bal);
System.out.println("1.Deposit 2. Withdraw 3. Display
int ch = sc.nextInt();                    4.Exit");
switch (ch)
{
case 1:
c.Deposit();
break;
case 2:
c.Withdraw();
break;
case 3:
c.Display();
break;
case 4:
System.exit(0);
default:
System.out.println("Invalid choice");
}
}
}
else if(o == 2)
{
Savings s = new Savings (name, acno, bal);
int ch =
System.out.println("1.Deposit 2. Withdraw 3.
    Display 4. Exit");
ich = sc.nextInt();
switch(ch)
```



Program 6

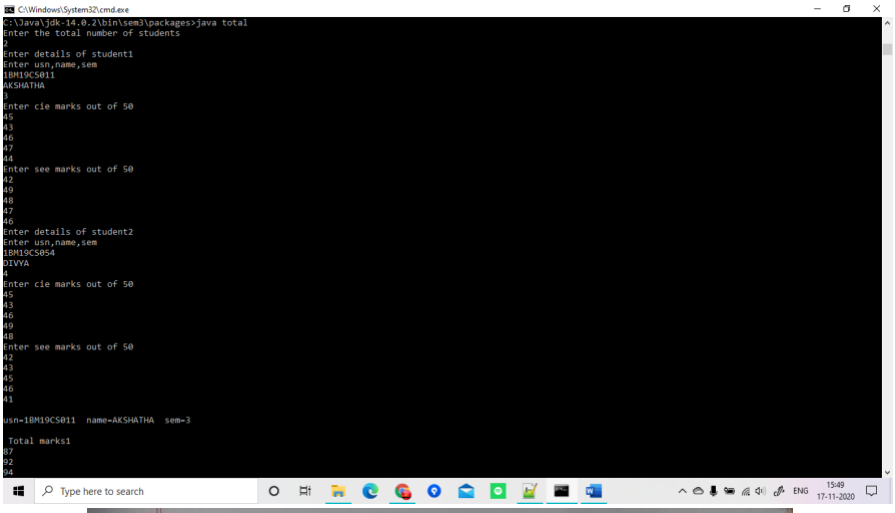
Create a package CIE which has two classes- Student and Internals. The class Personal has members like usn, name, sem. The class internals has an array that stores the internal marks scored in five courses of the current semester of the student. Create another package SEE which has the class External which is a derived class of Student. This class has an array that stores the SEE marks scored in five courses of the current semester of the student. Import the two packages in a file that declares the final marks of n students in all five courses.

```
package cie;

import java.util.*;
public class Student
{
```

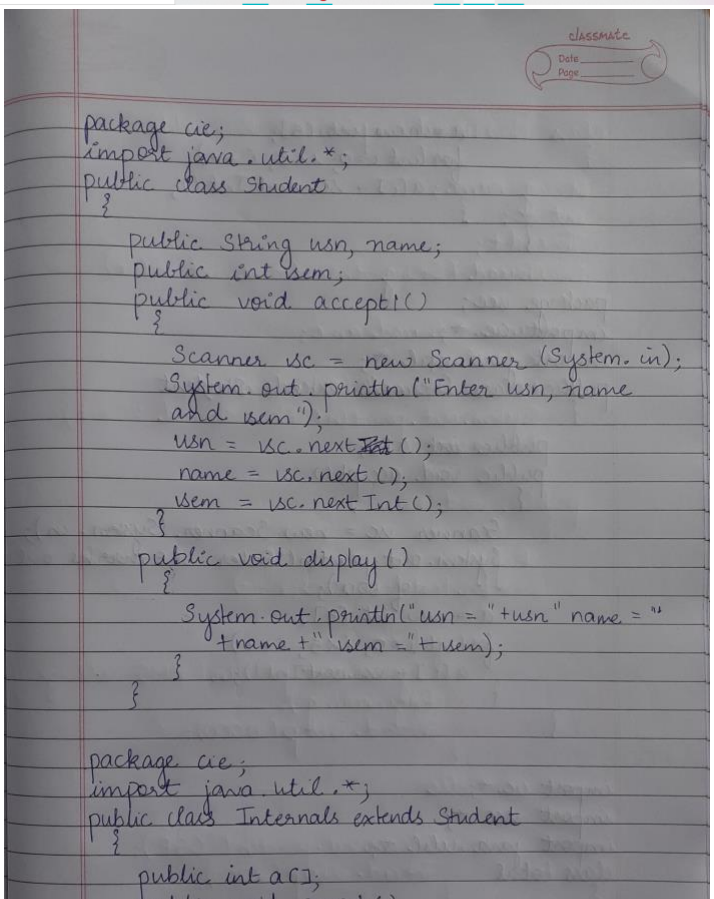
```
        public String usn,name;
        public int sem;

        public void accept1()
        {
    Scanner sc=new Scanner(System.in);
                System.out.println("Enter usn,name,sem");
                usn=sc.next();
                name=sc.next();
                sem=sc.nextInt();

   }

         public void display()
        {

                System.out.println("usn="+usn+"  name="+name+"  sem="+sem);

   }
}


package cie;

import java.util.*;

public class Internals extends Student
{

        public int a[];
public void accept()
        {
    Scanner sc=new Scanner(System.in);
                System.out.println("Enter cie marks out of 50");
    a=new int[5];
                for(int i=0;i<5;i++)
                 a[i]=sc.nextInt();
    }


}


package see;
import cie.*;
```

```
import java.util.*;

public class  Externals extends Student
{
          public int a[];
   public void accept()
          {
       Scanner sc=new Scanner(System.in);
                    System.out.println("Enter see marks out of 50");
       a=new int[5];
                    for(int i=0;i<5;i++)
                     a[i]=sc.nextInt();
    }
}



import cie.*;
import see.*;
import java.util.*;

class  total
{
          public static void main(String[] args)
          {



                   Scanner sc=new Scanner(System.in);
                    System.out.println("Enter the total number of students ");
       int n=sc.nextInt();
                    cie.Internals in[]=new cie.Internals[n];
                   see.Externals ex[]=new see.Externals[n];
       int total;
                    for(int j=0;j<n;j++)
                   {
                    System.out.println("Enter details of student"+(j+1));
                    in[j]=new cie.Internals();
                    ex[j]=new see.Externals();
                     in[j].accept1();

                     in[j].accept();
                     ex[j].accept();
                   }
```

```
                System.out.println(" ");
                for(int j=0;j<n;j++)
               {
 in[j].display();
 System.out.println("\n Total marks"+(j+1));

for(int k=0;k<5;k++)
                System.out.println(in[j].a[k]+(ex[j].a[k]));

               }}}
```



—



```
package cie;
import java.util.*;
public class Student
 {
    public String usn, name;
    public int sem;
    public void accept1()
     {
       Scanner sc = new Scanner (System.in);
       System.out.println("Enter usn, name
       and sem");
       usn = sc.nextLine();
       name = sc.next();
       sem = sc.nextInt();
     }
     public void display()
      {
        System.out.println("usn = "+usn" name = "
          +name+" sem ="+sem);
      }
 }

package cie;
import java.util.*;
public class Internals extends Student
 {
    public int a[];
```

```
        a = new int [5];
        for(int i=0; i<5; i++)
            a[i] = sc.nextInt();
        }
    }

package see;
import cie.*;
import java.util.*;
public class Externals extends Student
  {
    public int a[];
    public void accept()
      {
        Scanner sc = new Scanner (System.in);
        System.out.println("Enter SEE marks out
                of 100");
        a = new int [5];
        for(int i=0; i<5; i++)
            a[i] = sc.nextInt();
        }
    }

import cie.*;
import see.*;
import java.util.*;
class total
 {
   public static void main (String args[])
     {
       Scanner sc = new Scanner (System.in);
       System.out.println("Enter the total
```

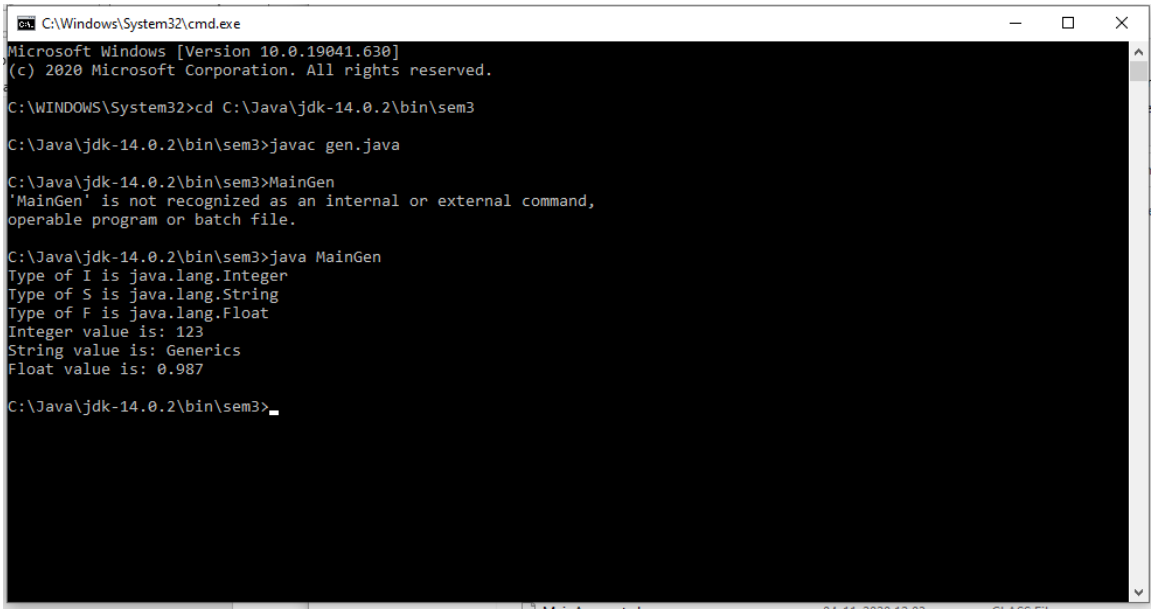
```
number of students ");
int n = isc.nextInt();
cie.Internals in[] = new cie.Internals[n];
isee.Externals ex[] = new isee.Externals[n];
int total;
for(int j=0; j<n; j++)
{
    System.out.println("Enter details
    of student" + (j+1));
    in[j] = new cie.Internals();
    ex[j] = new cie.Externals();
    in[j].accept1();
    in[j].accept();
    ex[j].accept();
}
System.out.println(" ");
for(int j=0; j<n; j++)
{
    in[j].display();
    System.out.println("\nTotal
    marks" + (j+1));
    for(int k=0; k<5; k++)
    System.out.println(in[j].a[k] +
        (ex[j].a[k]/2));
}
}
}
```

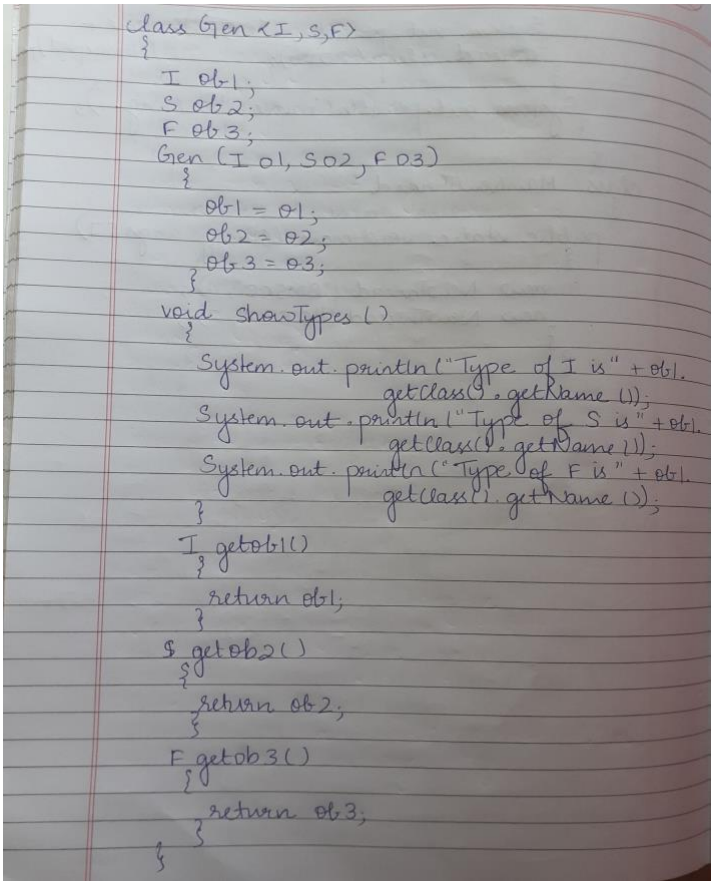
Program 7

Write a program to demonstrate generics with multiple object parameters.

```
class Gen<I,S,F>
{
  I ob1;
  S ob2;
  F ob3;
  Gen(I o1, S o2,F o3)
  {
   ob1=o1;
          ob2=o2;
          ob3=o3;
  }
  void showTypes()
  {
   System.out.println("Type of I is " +ob1.getClass().getName());
          System.out.println("Type of S is " +ob2.getClass().getName());
          System.out.println("Type of F is " +ob3.getClass().getName());
  }
          I getob1()
          {
           return ob1;
          }
   S getob2()
   {
    return ob2;
   }
          F getob3()
          {
                  return ob3;
          }

}
class MainGen
 {
 public static void main(String args[])
 {
 Gen<Integer, String,Float> tgObj = new Gen<Integer, String,Float>(123, "Generics",0.987f);
 tgObj.showTypes();

 int a=tgObj.getob1();
 System.out.println("Integer value is: " +a);
 String b=tgObj.getob2();
 System.out.println("String value is: " +b);
```

```
Float c=tgObj.getob3();
System.out.println("Float value is: "+c);
 }
}
```





```
class Gen<I,S,F>
{
 I ob1;
 S ob2;
 F ob3;
 Gen(I o1, S o2, F o3)
  {
   ob1 = o1;
   ob2 = o2;
   ob3 = o3;
  }
 void showTypes()
  {
   System.out.println("Type of I is" + ob1.
                      getClass().getName());
   System.out.println("Type of S is" + ob1.
                      getClass().getName());
   System.out.println("Type of F is" + ob1.
                      getClass().getName());
  }
 I getob1()
  {
   return ob1;
  }
 S getob2()
  {
   return ob2;
  }
 F getob3()
  {
   return ob3;
  }
}
```

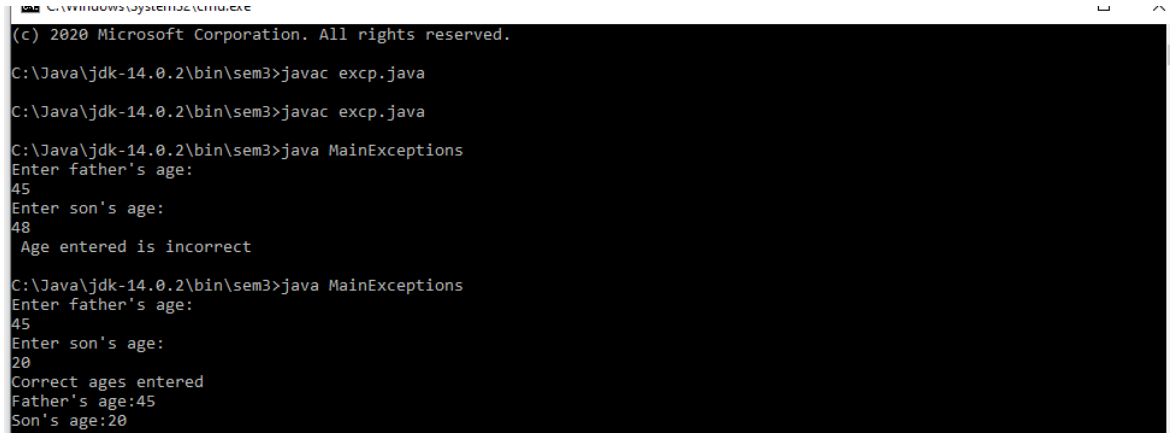
```
class MainGen
{
  public static void main (String args [])
  {
    Gen <Integer, String, Float> tgObj =
    new Gen <Integer, String, Float> (123, "Generics", 98.6);
    tgObj. showTypes();
    int a = tgObj. getob1();
    String b = tgObj. getob2();
    Float c = tgObj. getob3();
    System. out. println ("Integer value is :" +a
    "\n String value is:" +b "\n Float value is :" +c);
  }
}
```

Program 8

Write a program that demonstrates handling of exceptions in inheritance tree. Create a base class called “Father” and derived class called “Son” which extends the base class. In Father class, implement a constructor which takes the age and throws the exception Wrong Age( ) when the input age=father’s age.

```
import java.util.Scanner;

class WrongAge extends Exception {
int age;

WrongAge(int x) {
age = x;
}

public String toString() {
return "Age entered is incorrect";
}
}

class father {
int a;
father(int x) {
a = x;
}

void check() throws WrongAge {
if(a<0) {
throw new WrongAge(a);
}
}
}

class son extends father {
int age;
son(int fage, int sage) {
super(fage);
age = sage;
}

void check() throws WrongAge {
if (age >= a || age<0) {
throw new WrongAge(age);
}
else {
System.out.println("Correct ages entered");
System.out.println("Father's age:" + a + "\n" + "Son's age:" + age);
}
}
}

class MainExceptions{
public static void main(String args[]) {
```

```
Scanner sc = new Scanner(System.in);
System.out.println("Enter father's age:");
int f = sc.nextInt();
father ff = new father(f);
try {
  ff.check();
  }
catch (WrongAge e) {
System.out.println(" "+e);
}
System.out.println("Enter son's age:");
int s = sc.nextInt();
son ss = new son(f, s);
try {
  ss.check();
  }
catch (WrongAge e) {
System.out.println(" "+e);
}
}
}
```



```
import java.util.*;
class WrongAge extends Exception {
int age;
WrongAge (int x) {
age = x;
}
public String toString (){
return "Age entered is incorrect";
}
}
class father {
int a;
father (int x) {
a = x;
}
}
class son extends father {
int age;
son (int fage, int sage) {
super (fage);
age = fage;
}
void check () throws WrongAge {
if ( age >= a || age < 0 || a < 0) {
throw new WrongAge (age);
}
else {
System.out.println ("Correct ages entered \n Fathers
age :" + a + "\n" + "Sons age" + age);
}
}
}
```

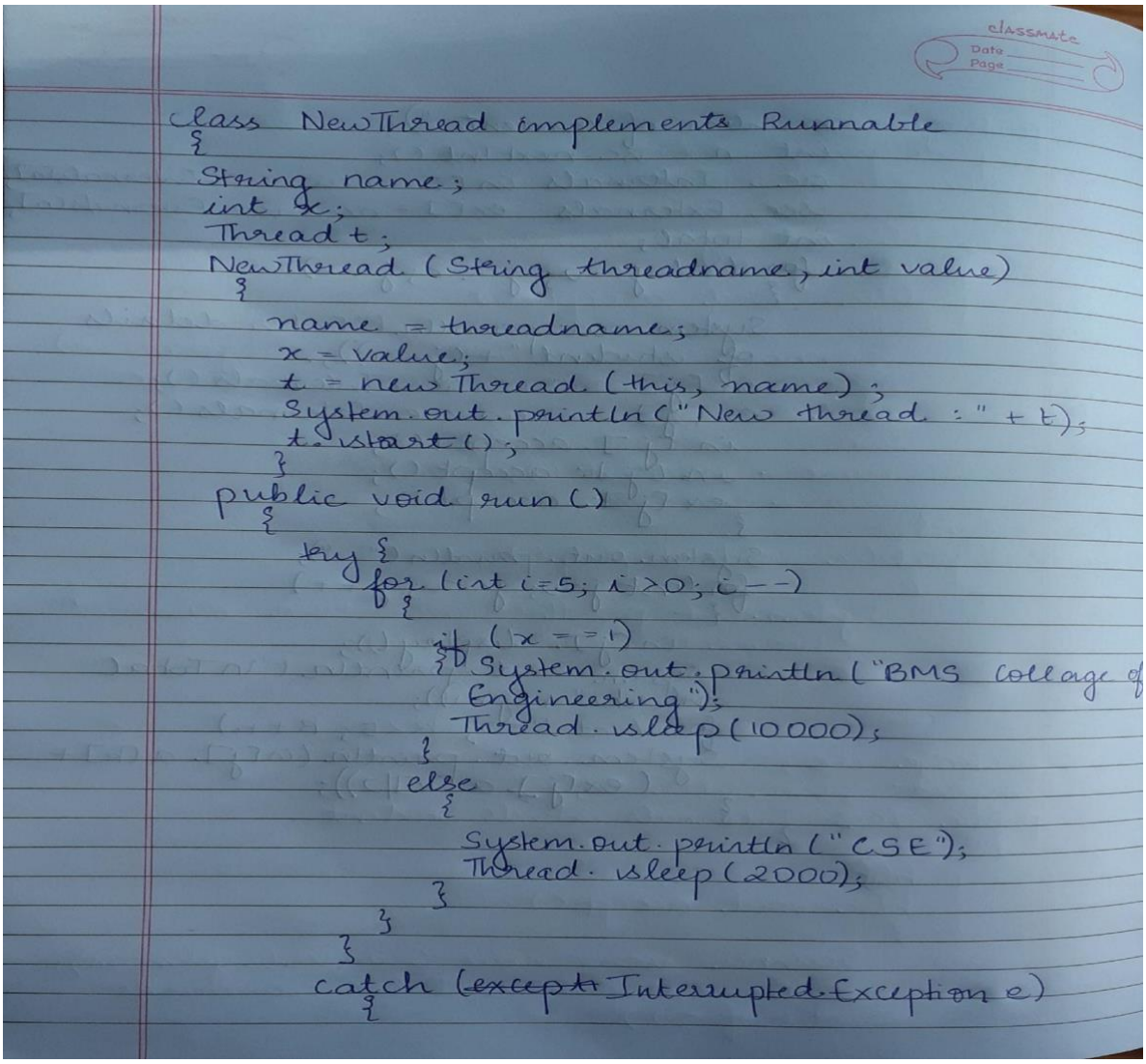
```
class ExceptionsMain {
public static void main (String args[]) {
Scanner sc = new Scanner (System.in);
System.out.println(" Enter fathers age and sons age");
int f = sc.nextInt();
int s = sc.nextInt();
son ss = new son (f,s);
try {
    ss.check();
}
catch (WrongAge e) {
System.out.println(" "+e);
}
}
}
```

Program 9

Write a program which creates two threads, one thread displaying “BMS College of Engineering” once every ten seconds and another displaying “CSE” once every two seconds.

```
class NewThread implements Runnable
{
String name;
int x;
Thread t;
NewThread(String threadname,int value)
{
name = threadname;
x=value;
t = new Thread(this, name);
System.out.println("New thread: " + t);
t.start();
}

public void run()
{
try {
for(int i = 5; i > 0; i--)
{
            if(x==1)
            {
           System.out.println("BMS Collage of Engineering");
        Thread.sleep(10000);
            }
            else
            {
                        System.out.println("CSE");
        Thread.sleep(2000);
            }
}
} catch (InterruptedException e) {
System.out.println(name + "Interrupted");
}
System.out.println(name + " exiting.");
}
}

class MainNewThread
{
public static void main(String args[])
{
new NewThread("BMSCE",1);
new NewThread("CSE",2);


}
}
```

```
C:\Java\jdk-14.0.2\bin\sem3>javac threads.java

C:\Java\jdk-14.0.2\bin\sem3>java MainNewThread
New thread: Thread[BMSCE,5,main]
New thread: Thread[CSE,5,main]
BMS Collage of Engineering
CSE
CSE
CSE
CSE
CSE
BMS Collage of Engineering
CSE exiting.
BMS Collage of Engineering
BMS Collage of Engineering
BMS Collage of Engineering
BMSCE exiting.

C:\Java\jdk-14.0.2\bin\sem3>
```



```
class NewThread implements Runnable
{
String name;
int x;
Thread t;
NewThread (String threadname, int value)
{
    name = threadname;
    x = value;
    t = new Thread (this, name);
    System.out.println("New thread : " + t);
    t.start();
}
public void run()
{
    try {
        for (int i=5; i>0; i--)
        {
            if (x==1)
            {
                System.out.println("BMS collage of
                Engineering");
                Thread.sleep(10000);
            }
            else
            {
                System.out.println("CSE");
                Thread.sleep(2000);
            }
        }
    }
    catch (except InterruptedException e)
    {
```

```
            System.out.println ("CSE"+ name +
            Thread.sleep(2000);          "Interrupted");
        }
    System.out.println(name+" exiting.");
    }
}
class MainNewThread
{
    public static void main (String args[])
    {
        new NewThread ("BMSCE", 1);
        new NewThread ("CSE", 2);
    }
}
```

Program 10

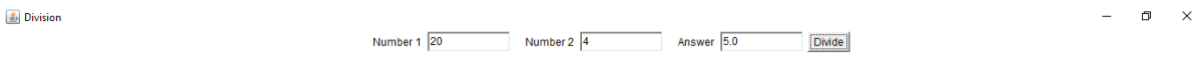
Write a program that creates a user interface to perform integer divisions. The user enters two numbers in the text fields, Num1 and Num2. The division of Num1 and Num2 is displayed in the Result field when the Divide button is clicked. If Num1 or Num2 were not an integer, the program would throw a NumberFormatException. If Num2 were Zero, the program would throw an Arithmetic Exception Display the exception in a message dialog box.

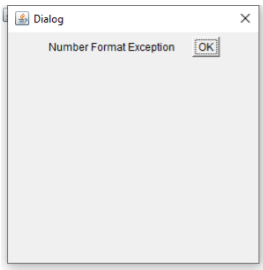
```
import java.awt.*;

import java.awt.event.*;


public class Division extends Frame implements ActionListener {

public class DialogD extends Frame{
String msg;
DDialog myDialog;


public DialogD() {
myDialog = new DDialog(this, "New Dialog Box", msg);
}
}
String msg;

TextField num1, num2, res;

Label l1, l2, l3;

Button div;



public Division() {

setLayout(new FlowLayout());

l1 = new Label("Number 1",Label.RIGHT);

l2 = new Label("Number 2", Label.RIGHT);

l3 = new Label("Answer", Label.RIGHT);

num1 = new TextField(10);

num2 = new TextField(10);

res = new TextField(10);

div = new Button("Divide");
```

```
add(l1);

add(num1);

add(l2);

add(num2);

add(l3);

add(res);

add(div);


div.addActionListener(this);

addWindowListener(new MyWindowAdapter());

}



public void actionPerformed(ActionEvent ae) {

int num1 = 0, num2 = 0;


try {

num1 = Integer.parseInt(this.num1.getText());

num2 = Integer.parseInt(this.num2.getText());

double num3 =(double) num1 / num2;


res.setText(String.valueOf(num3));

msg = "The Division was Successful";


} catch (NumberFormatException e) {

System.out.println(e);

res.setText("");

msg = "Number Format Exception ";

DDialog dd = new DDialog(this, "Dialog", msg);
```

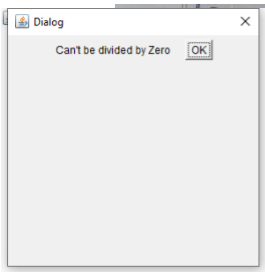
```
dd.setVisible(true);
return;
}

try {

if(num2==0)

throw new ArithmeticException();


msg = "Can't be divided by Zero";


} catch (ArithmeticException e) {

System.out.println("Can't be divided by Zero" + e);

res.setText("");

msg = "Can't be divided by Zero";

DDialog dd = new DDialog(this, "Dialog", msg);
dd.setVisible(true);
return;

}

}



public void paint(Graphics g) {

g.drawString(msg, 80, 100);

}



public static void main(String[] args) {

Division appwin = new Division();


appwin.setSize(new Dimension(480,280));

appwin.setTitle("Division");

appwin.setVisible(true);

}
```

```
}


class MyWindowAdapter extends WindowAdapter {

public void windowClosing(WindowEvent we)

{

System.exit(0);

}

}

class DDialog extends Dialog {
DDialog(Frame parent, String title, String msg)
{
super(parent,title,false);
setLayout(new FlowLayout());
setSize(300,300);
add(new Label(msg));
Button b;
add(b = new Button("OK"));
b.addActionListener((ae)->dispose());
addWindowListener(new WindowAdapter() {
public void windowClosing(WindowEvent we)
{
dispose();
}
});
}
}
```




Dialog
Number Format Exception
OK


Number 1
20
Number 2
aaa
Answer
Divide


Dialog
Can't be divided by Zero
OK


Number 1
20
Number 2
0
Answer
Divide

```
import java.awt.*;
import java.awt.event.*;
public class Division extends Frame implements
ActionListener {
public class DialogD extends Frame {
String msg;
DDialog myDialog;
public DialogD() {
myDialog = new DDialog (this, "New Dialog Box",
}
}
String msg;
TextField msg; num1, num2, res;
Lable l1, l2, l3;
Button div;
public Division () {
setLayout (new FlowLayout());
l1 = new Label ("Number 1", Label.RIGHT);
l2 = new Label ("Number 2", Label.RIGHT);
l3 = new Label ("Answer", Label.RIGHT);
num1 = new TextField (10);
num2 = new TextField (10);
res  = new TextField (10);
div = new Button ("Divide");
add (l1);
add (num1);
add (l2);
add (num2);
add (l3);
add (res);
add (div);
div. addActionListner (this);
```

```
addWindowListner(new MyWindowAdapter());
}
public void actionPerformed(ActionEvent ae) {
int num1=0, num2=0;
try {
num1 = Integer.parseInt(this.num1.getText());
num2 = Integer.parseInt(this.num2.getText());
double num3 = (double) num1/num2;
res.setText(String.valueOf(num3));
msg = "The Division was Successful";
} catch (NumberFormatException e) {
System.out.println(e);
res.setText(" ");
msg = "Number Format Exception";
DDialog dd = new DDialog(this, "Dialog", msg);
dd.setVisible(true);
return;
}
try {
if(num2==0)
throw new ArithmeticException();
msg = "Can't divide by zero";
} catch (ArithmeticException e) {
System.out.println("Can't divide by zero"+e);
res.setText(" ");
msg = "Can't be divided by zero";
DDialog dd = new DDialog(this, "Dialog", msg);
dd.setVisible(true);
return;
}
}
public void paint(Graphics g) {
```

```
g.drawString(msg, 80, 100);
}
public static void main(String[] args) {
Division appwin = new Division();
appwin.setSize(new Dimension(480, 280));
appwin.setTitle("Division");
appwin.setVisible(true);
}
}
class MyWindowAdapter extends WindowAdapter {
public void windowClosing(WindowEvent we)
{
System.exit(0);
}
}
class DDialog extends Dialog {
DDialog(Frame parent, String title, String msg)
{
super(parent, title, false);
setLayout(new FlowLayout());
setSize(300, 300);
add(new Label(msg));
Button b;
add(b = new Button("OK"));
b.addActionListner((ae) -> dispose());
addWindowListner(new WindowAdapter() {
public void windowClosing(WindowEvent we)
{
dispose();
}
});
}
}
```