JAVA PROGRAMS

1. **Write java program to print Biggest of 3 Numbers using Command line arguments.**

```
class Biggest
 {
  int biggestNumber(int a,int b,int c)
   {
    int G=a;
    if(b>G)
     {
      G=b;
     }
    if(c>G)
     {
      G=c;
     }
     return(G);
   }
  public static void main(String src[])
   {
    int x=Integer.parseInt(src[0]);
    int y=Integer.parseInt(src[1]);
    int z=Integer.parseInt(src[2]);
    Biggest obj=new Biggest();
    System.out.println("Biggest number="+obj.biggestNumber(x,y,z));
   }
 }
```

JAVA PROGRAMS

Output:

java Biggest 23 56 54

Biggest number=56

2. Write a java program to print Factorial of a given number.

```
import java.util.Scanner;
class Factorial
 {
  long findFactorial(int n)
   {
    long fact=1;
    while(n>0)
     {
      fact=fact*n;
      n-=1;
     }
     return(fact);
   }
  public static void main(String src[])
   {
    System.out.print("Enter a number for find factorial=");
    Scanner scan=new Scanner(System.in);
    int N=scan.nextInt();
    Factorial obj=new Factorial();
    System.out.print("Factorial of "+N+" = "+obj.findFactorial(N));
   }
 }
```

Output:

Enter a number for find factorial=7

Factorial of 7 = 5040

3. Write a java program to print sum of Sum of Digits and check for palindrome.

```
import java.util.Scanner;
class Palindrome
 {
  int sum_Of_Digits(int n)
   {
    int sum=0;
```

JAVA PROGRAMS

```
        while(n>0)
         {
           sum=sum+(n%10);
           n=n/10;
         }
         return (sum);
       }
     boolean palindromeCheck(int n)
       {
        int rev=0, N=n;
        while(n>0)
         {
           rev=rev*10+(n%10);
           n=n/10;
         }
        if(rev==N)
         {
           return(true);
         }
        else
         {
           return(false);
         }
       }
       public static void main(String src[])
        {
          System.out.print("Enter a number=");
          Scanner scan=new Scanner(System.in);
          int n=scan.nextInt();
          Palindrome obj=new Palindrome();
          System.out.println("Sum of all digits of "+n+" = "+obj.sum_Of_Digits(n));
          if(obj.palindromeCheck(n))
           {
             System.out.println(n+" is a palindrome number");
           }
          else
           {
             System.out.println(n+" is not a palindrome number");
           }
        }
  }
```

Output:

JAVA PROGRAMS

```
Enter a number=678
Sum of all digits of 678 = 21
678 not a palindrome number
```

4. Write a java program to print the names in sorted order using arrays.

```
import java.util.Scanner;
class Names
 {
  String[] namesSorting(String[] str)
   {
    int len=str.length;
    for(int i=0; i<len; i++)
     {
      for(int j=i+1; j<len; j++)
       {
        if(str[i].compareTo(str[j])>0)
         {
          String ch=str[i];
          str[i]=str[j];
          str[j]=ch;
         }
       }
     }
    return(str);
   }
  public static void main(String src[])
   {
    System.out.print("How many names do you want to enter=");
    Scanner scan=new Scanner(System.in);
    int n=scan.nextInt();
    String str[]=new String[n];
    Scanner sca=new Scanner(System.in);
    for(int i=0; i<n; i++)
     {
      System.out.print("Enter "+(i+1)+"th name=");
      str[i]=sca.nextLine();
     }
     System.out.println("Entered names are following:");
     for(int i=0; i<n; i++)
      {
        System.out.println((i+1)+". "+str[i]);
      }
    Names obj=new Names();
```

```
        String s[]=obj.namesSorting(str);
         System.out.println("Sorted names are following:");
          for(int i=0; i<n; i++)
           {
              System.out.println((i+1)+". "+s[i]);
           }
       }
    }
```

Output:

```
 How many names do you want to enter=4
Enter 1th name=Ranjan
Enter 2th name=Suresh
Enter 3th name=Aaditaya
Enter 4th name=Aman
Entered names are following:
1. Ranjan
2. Suresh
3. Aaditaya
4. Aman
Sorted names are following:
1. Aaditaya
2. Aman
3. Ranjan
4. Suresh
```

5. **Write a java program to compute matrix multiplication using arrays.**

```
import java.util.Scanner;
class Matrix
 {
   int[][] matInput(int r,int c)
     {
      int mat[][]=new int[r][c];
      Scanner s=new Scanner(System.in);
      for(int i=0; i<r; i++)
       {
         for(int j=0; j<c; j++)
           {
             System.out.print("Enter "+(i+1)+"*"+(j+1)+"th element=");
             mat[i][j]=s.nextInt();
           }
       }
```

JAVA PROGRAMS

```
      return(mat);
   }
int[][] matMultiplication(int mat1[][],int mat2[][])
 {
   int row1=mat1.length;
   int col1=mat1[0].length;
   int row2=mat2.length;
   int col2=mat2[0].length;
   int mat3[][]=new int[row1][col2];
   for(int i=0; i<row1; i++)
    {
      for(int j=0; j<col2; j++)
       {
         int sum=0;
         for(int k=0; k<col1; k++)
          {
            sum=sum+(mat1[i][k]*mat2[k][j]);
          }
         mat3[i][j]=sum;
       }
    }
   return(mat3);
 }
 void matOutput(int mat[][])
  {
   int row=mat.length;
   int col=mat[0].length;
   for(int i=0; i<row; i++)
    {
      for(int j=0; j<col; j++)
       {
         System.out.print(mat[i][j]+"\t");
       }
      System.out.println();
    }
  }
public static void main(String src[])
 {
   System.out.println("...Matrix Multiplication...");
   System.out.print("Enter no. of rows and columns of 1st matrix=");
   Scanner scan=new Scanner(System.in);
   int r1=scan.nextInt();
```

```
        int c1=scan.nextInt();
        System.out.print("Enter no. of rows and columns of 2nd matrix=");
        int r2=scan.nextInt();
        int c2=scan.nextInt();
        Matrix obj=new Matrix();
        if(c1==r2)
         {
          System.out.println("Enter values in 1st "+r1+"*"+c1+" matrix=");
          int mat1[][]=obj.matInput(r1,c1);
          System.out.println("Enter values in 2nd "+r2+"*"+c2+" matrix=");
          int mat2[][]=obj.matInput(r2,c2);
          int mat3[][]=obj.matMultiplication(mat1,mat2);
          System.out.println("\nEntered 1st matrix is following:");
          obj.matOutput(mat1);
          System.out.println("\nEntered 2nd matrix is following:");
          obj.matOutput(mat2);
          System.out.println("\nAfter multiplication resultant 3rd matrix is following:");
          obj.matOutput(mat3);
         }
        else
         {
           System.out.println("NOTE: Columns of 1st matrix should be equal to rows of 2nd
matrix");
           System.out.println("...Try again...");
         }
      }
 }
```

Output:

```
  ...Matrix Multiplication...
Enter no. of rows and columns of 1st matrix=2
3
Enter no. of rows and columns of 2nd matrix=3
2
Enter values in 1st 2*3 matrix=
Enter 1*1th element=3
Enter 1*2th element=4
Enter 1*3th element=5
Enter 2*1th element=4
Enter 2*2th element=3
Enter 2*3th element=5
Enter values in 2nd 3*2 matrix=
```

JAVA PROGRAMS

```
Enter 1*1th element=3
Enter 1*2th element=4
Enter 2*1th element=5
Enter 2*2th element=6
Enter 3*1th element=7
Enter 3*2th element=6

Entered 1st matrix is following:
3     4     5
4     3     5

Entered 2nd matrix is following:
3     4
5     6
7     6

After multiplication resultant 3rd matrix is following:
64     66
62     64
```

II. Method Overloading

6. **Write a java program to demonstrate method overloading to add two integers, add two strings.**

```
import java.util.Scanner;
class MethodsOverloading
 {
  void addition(int a,int b)
   {
    System.out.println("Addition of "+a+" and "+b+" is= "+(a+b));
   }
  void addition(String s1, String s2)
   {
    String str=s1+s2;
    System.out.println("Addition of entered strings= "+str);
   }
 }
 class MainClass
  {
     public static void main(String src[])
      {
```

```
        System.out.print("Enter two string values=");
        Scanner scan=new Scanner(System.in);
        String str1=scan.nextLine();
        String str2=scan.nextLine();
        System.out.print("Enter two integer values=");
        int x=scan.nextInt();
        int y=scan.nextInt();
        MethodsOverloading obj=new MethodsOverloading();
        obj.addition(x,y);
        obj.addition(str1,str2);
       }
  }
```

Output:

```
Enter two string values=Reva
University
Enter two integer values=56
78
Addition of 56 and 78 is= 134
Addition of entered strings= RevaUniversity
```

III. Constructor overloading

7. **Write a java program for Rectangle class using constructor overloading with different number of parameter list.**

```
import java.util.Scanner;
class Rectangle
 {
  Rectangle(int L,int B)
   {
    int pe=2*(L+B);
     System.out.println("Perimeter of Rectangle= "+pe);
   }
  Rectangle(int area)
   {
    System.out.println("Area of Rectangle= "+area);
   }
 }
class MainClass
 {
  public static void main(String src[])
```

```
        {
          System.out.print("Enter length and breadth of rectangle=");
          Scanner scan=new Scanner(System.in);
          int l=scan.nextInt();
          int b=scan.nextInt();
          int Ar=l*b;
          Rectangle obj1=new Rectangle(Ar);
          Rectangle obj2=new Rectangle(l,b);
        }
    }
```

Output:

```
Enter length and breadth of rectangle=78
88
Area of Rectangle= 6864
Perimeter of Rectangle= 332
```

IV.Inheritance & Abstract class

8.Write a java program to demonstrate i. Simple Inheritance ii. multilevel inheritance.

i. Simple Inheritance

```
import java.util.Scanner;

class Base
 {
    String name,SRN;
    long mob;
    void input()
     {
        System.out.print("Enter Name,SRN and Mobile no of student=");
        Scanner scan=new Scanner(System.in);
        name=scan.nextLine();
        SRN=scan.nextLine();
        mob=scan.nextLong();
```

JAVA PROGRAMS

```
        }
    }
class Derived extends Base
 {
   void output()
    {
      System.out.println("Entered values are following:");
      System.out.println("Name: "+name+"\nS R N: "+SRN+"\nMobile: "+mob);
    }
   public static void main(String src[])
    {
      Derived obj=new Derived();
      obj.input();
      obj.output();
    }
 }
```

Output:

```
Enter Name,SRN and Mobile no of student=Aaditaya
R21DE678
9856847345
Entered values are following:
Name: Aaditaya
S R N: R21DE678
Mobile: 9856847345
```

ii. multilevel inheritance.

JAVA PROGRAMS

```
import java.util.Scanner;
class Base
  {
   long bin;
   void input()
    {
      System.out.print("Enter a binary number=");
      Scanner scan=new Scanner(System.in);
      bin=scan.nextLong();
    }
  }
class Derived extends Base
 {
   long binToDec(long bin)
    {
      if( bin==0)
       {
         return(0);
       }
      else
       {
         return((bin%10)+2*binToDec(bin/10));
       }
    }
 }
```

JAVA PROGRAMS

```
class MainClass extends Derived
 {
   void output()
    {
      System.out.print("Decimal number= "+binToDec(bin));
    }
   public static void main(String src[])
    {
      MainClass obj=new MainClass();
      obj.input();
      obj.output();
    }
 }
```

Output:

Enter a binary number=1011

Decimal number= 11

9.Write a Java program to implement an abstract class.

```
abstract class Bank
 {
   abstract double rateOfInterest();
 }
```

```
class SBI extends Bank
 {
```

JAVA PROGRAMS

```
        double rateOfInterest()
          {
          return(4.5);
          }
        }

    class POSB extends Bank
     {
      double rateOfInterest()
        {
        return(7.8);
        }
      }

class BankingServices
  {
   public static void main(String abs[])
     {
      Bank obj1=new POSB();
      Bank obj2=new SBI();
      System.out.println("SBI Rate of interest= "+obj2.rateOfInterest());
      System.out.println("POSB Rate of interest= "+obj1.rateOfInterest());
     }
  }
```

Output:

JAVA PROGRAMS

SBI Rate of interest= 4.5

POSB Rate of interest= 7.8

V.Method Overriding

10 Write a java program to demonstrate Method overriding (use super keyword) .

```
import java.util.Scanner;
class Base
 {
   void arithmeticOperation(int a,int b,int c)
       {
         int add=a+b+c;
         System.out.println("Addition of entered numbers are: "+add);
       }
 }

class Derived extends Base
 { @Override
   void arithmeticOperation(int a,int b,int c)
     {
         super.arithmeticOperation(a,b,c);
         int mul=a*b*c;
         System.out.println("Multiplication of entered numbers are: "+mul);
     }

 }
class MainClass
```

JAVA PROGRAMS

```
 {
   public static void main(String src[])
    {
      System.out.print("Enter three numbers=");
      Scanner scan=new Scanner(System.in);
      int x=scan.nextInt();
      int y=scan.nextInt();
      int z=scan.nextInt();
      Derived obj=new Derived();
      obj.arithmeticOperation(x,y,z);
    }
 }
```

Output:

```
Enter three numbers=56
67
78
Addition of entered numbers are: 201
Multiplication of entered numbers are: 292656
```

VI. Packages

11. Write a Java program to demonstrate user defined packages.

```
package mypackage;
import java.util.Scanner;
public class ArrayClass
  {
   public int[] inputArray()
```

JAVA PROGRAMS

```
  {
    Scanner scan=new Scanner(System.in);
    System.out.print("How many numbers do you want to enter in array list=");
    int n=scan.nextInt();
    int Arr[]=new int[n];
    for(int i=0; i<n; i++)
     {
       System.out.print("Enter "+(i+1)+"th number=");
       Arr[i]=scan.nextInt();
     }
     return(Arr);
  }
 public int greatestNumber(int Arr[])
  {
    int G=Arr[0];
    for(int i:Arr)
     {
       if(G<i)
        {
          G=i;
        }
     }
     return(G);
  }
 public static void main(String src[])
```

JAVA PROGRAMS

```
    {
      ArrayClass obj=new ArrayClass();
      int A[]=obj.inputArray();
      int G=obj.greatestNumber(A);
      System.out.print("Greatest Number= "+G);
    }
  }
```

Package compilation: `javac -d . ArrayClass.java`

Package run: `java mypackage.ArrayClass`

Output:

```
 How many numbers do you want to enter in array list=4
Enter 1th number=45
Enter 2th number=67
Enter 3th number=98
Enter 4th number=78
Greatest Number= 98
```

#Now using created array functionons inside "mypackage" in our another programs:

```
import mypackage.ArrayClass;
class BiggestElement              // If in package methods are protected then we have to extends  class.
  {
    public static void main(String src[])
     {
       ArrayClass obj=new ArrayClass();
       int A[]=obj.inputArray();
       int G=obj.greatestNumber(A);
```

```
        System.out.print("Greatest Number= "+G);
    }
  }
```

Program compilation: javac BiggestElement.java

Program run: java BiggestElement

Output:

How many numbers do you want to enter in array list=5

Enter 1th number=23

Enter 2th number=65

Enter 3th number=678

Enter 4th number=343

Enter 5th number=567

Greatest Number= 678

VII. Multiple Inheritance: Interface

12.Write a Java program to illustrate the multiple inheritance by using i. single Interface ii. Multiple interfaces iii. Inherited interface.

Single Interface

```
interface Bank
 {
    String name="Post_Office_Saving_Bank";//By default, public,static and final
    double rateOfInterest();//By default public,abstract
 }
class BankingServices implements Bank
 {
     public double rateOfInterest()
```

```
        {
           return(7.8);
        }

      public static void main(String src[])
       {
         Bank obj=new BankingServices();
         System.out.println("Bank name: "+name);
         System.out.print("Rate of interest="+obj.rateOfInterest());
       }
  }
```

Output:

```
Bank name: Post_Office_Saving_Bank
Rate of interest=7.8
```

Multiple interfaces

```
interface A
 {
   public abstract void print();
 }
interface B
 {
   public abstract void print();
 }
class University implements A,B
 {
```

```
    public void print()
     {
       System.out.println("...University biodata...");
       System.out.println("Name: Reva University\nAddress: Bengaluru,Hindustan");
     }
    public static void main(String src[])
     {
       University obj=new University();
       obj.print();
     }
  }
```

Output:

```
...University biodata...
Name: Reva University
Address: Bengaluru,Hindustan
```

VII) final, super, static keywords

13. Write a java program to illustrate the keywords i)super ii)static iii)final

i)Super:

```
class POSB
 {
   String name="Post_Office_Saving_Bank";
   double rateOfInterest()
    {
      return(7.8);
    }
```

```
 }
class IPPB extends POSB
 {
   String name="India_Post_Payment_Bank";
   double rateOfInterest()
    {
      System.out.println(super.name+"\tRate of interest="+super.rateOfInterest());
      return(4.0);
    }
   public static void main(String src[])
    {
      IPPB obj=new IPPB();
      System.out.println(obj.name+"\tRate of interest="+obj.rateOfInterest());
    }
 }
```

Output:

```
Post_Office_Saving_Bank Rate of interest=7.8
India_Post_Payment_Bank Rate of interest=4.0
```

ii)static:

```
import java.util.Scanner;
class Table
 {
   static int n;//static variable
   static{ //static block
     Scanner scan=new Scanner(System.in);
```

JAVA PROGRAMS

```
    System.out.print("Enter a number=");
    n=scan.nextInt();
    System.out.println(n+" table is following...");
  }
  static void generateTable()//static method
   {
     for(int i=1; i<11; i++)
       {
         System.out.println(n+"*"+i+"="+(i*n));
       }
    }
  public static void main(String src[])
   {
     generateTable();
   }
 }
```

Output:

```
Enter a number=8
8 table is following...
8*1=8
8*2=16
8*3=24
8*4=32
8*5=40
8*6=48
```

JAVA PROGRAMS

8*7=56

8*8=64

8*9=72

8*10=80

Final:

```
final class A //if we make any class as final, we can not extends it.
 {
   final int var=78;
   final void run()
    {
      System.out.println("Final_Keyword_implementation");
    }
    var=67;//we can not change the value of final variable.
 }
class Final_Keyword extends A
 {
   void run() // we can not override the final method.
    {
      System.out.println("Final_keyword_inplementation");
    }
   public static void main(String str[])
    {
      Final_Keyword obj=new Final_Keyword();
      obj.run();
    }
```

```
 }
```

Output:

Complie time error.

IX) Exception handling

14. Write a java program to demonstrate exception handling with i. single catch block ii. multiple catch blocks.

```
class Exception_Handling
 {
   public static void main(String src[])
    {
     int a,b,Z;
     a=65;
     b=0;
     System.out.println("Start_part");
     try{
       Z=a/b;
       System.out.println(Z);
     }
     catch(Exception ex){
      System.out.println(ex);
     }
     System.out.println("End_part");
    }
 }
```

Output:

Start_part

JAVA PROGRAMS

java.lang.ArithmeticException: / by zero

End_part

Multiple_Catch_Blocks:

```
import java.util.Scanner;
class Exception_Handling
 {
   public static void main(String src[])
    {
      int a,b; double z;
      int Arr[]=new int[]{1,2,3,4,5};
      String str=null;
      Scanner scan=new Scanner(System.in);
      System.out.println("Start_part");
      try{
        System.out.print("Enter two numbers for division=");
        a=scan.nextInt();
        b=scan.nextInt();
        z=a/b;
        System.out.println(z);
        System.out.print("Enter array index no for print array element=");
        int n=scan.nextInt();
        System.out.println(Arr[n]);
        System.out.println(str.length());
      }
```

```
      catch(ArithmeticException ex){
        System.out.println("Arithmetic exception occurs");
      }
      catch(ArrayIndexOutOfBoundsException ex){
        System.out.println("ArrayIndexOutOfBounds exception occurs");
      }
      catch(NullPointerException ex){
        System.out.println("NullPointerException occurs");
      }
      System.out.println("End_part");
    }
 }
```

Output1:

```
Start_part
Enter two numbers for division=2
0
Arithmetic exception occurs
End_part
```

Output2:

```
Start_part
Enter two numbers for division=4
2
2.0
Enter array index no for print array element=7
ArrayIndexOutOfBounds exception occurs
```

End_part

Output3:

Start_part

Enter two numbers for division=4

2

2.0

Enter array index no for print array element=4

5

NullPointerException occurs

End_part

X) Multithreading

15) Write a Java program to demonstrate the concept of Inter thread communication by Suitable example

```
import java.util.Scanner;
class Customer
  {
    double amount=5000;
    synchronized void withdrawal(double amount)
      {
         System.out.println("Going to withdrowal...");
         if(this.amount< amount)
          {
            System.out.println("Less balance waiting for deposit...");
            try{
               wait();
```

```
            }
            catch (Exception ex)
             {
               System.out.println(ex);
             }
           }
        this.amount=this.amount-amount;
        System.out.println("Withdrowal is completed...");
     }
  synchronized void deposit(double amount)
    {
     System.out.println("Going to deposit...");
     this.amount=this.amount+amount;
     System.out.println("Deposit completed...");
     notify();
    }
   void viewBalance()
    {
     System.out.println("Your total balance is: "+amount);
    }
}

class Banking
 {
    public static void main(String src[])
```

JAVA PROGRAMS

```
{
 final Customer obj=new Customer();
 Scanner scan=new Scanner(System.in);

 while(true)
  {
   System.out.println("1. for deposit.\t 2. withdrawal \t 3. for view balance \t 4. for exit");
   System.out.println("Enter your choice= ");
   int choice=scan.nextInt();
     if(choice==1)
      {
        System.out.print("Enter ammount for deposit=");
        final double amount=scan.nextDouble();
        new Thread(){
          public void run()
           {
             obj.deposit(amount);
           }
         }.start();
      }
     else if(choice==2)
      {
        System.out.println("Enter amount for withdrawal=");
        final double amount=scan.nextDouble();
        new Thread(){
```

JAVA PROGRAMS

```
                public void run()
                 {
                   obj.withdrawal(amount);
                 }
                }.start();
            }
          else if(choice==3)
            {
              obj.viewBalance();
            }
          else if (choice==4)
            {
              System.out.println("Exit....");
              break;
            }
          else
            {
             System.out.println("Your choice is wrong try again...");
            }
        }
      }
  }
```

Output:

1. for deposit. 2. withdrawal 3. for view balance 4. for exit

Enter your choice=

JAVA PROGRAMS

1

Enter ammount for deposit=3456

1. for deposit. 2. withdrawal 3. for view balance 4. for exit

Enter your choice=

Going to deposit...

Deposit completed...

3

Your total balance is: 8456.0

1. for deposit. 2. withdrawal 3. for view balance 4. for exit

Enter your choice=

2

Enter amount for withdrawal=

455

1. for deposit. 2. withdrawal 3. for view balance 4. for exit

Enter your choice=

Going to withdrowal...

Withdrowal is completed...

3

Your total balance is: 8001.0

1. for deposit. 2. withdrawal 3. for view balance 4. for exit

Enter your choice=

4

Exit....

16.Write a Program on MultiThreads using Thread Class

JAVA PROGRAMS

```
import java.util.Scanner;
class Table extends Thread
 {
   @Override
   public void run( )
    {
      Scanner scan=new Scanner(System.in);
      System.out.print("Enter a number for generate table=");
      int n=scan.nextInt();
      try{
         for(int i=1; i<=10; i++)
          {
            System.out.println(n+"*"+i+"="+(i*n));
            Thread.sleep(1000);
          }
      }
     catch(InterruptedException ex)
      {
         System.out.print(ex);
      }
    }
 }
class A extends Thread
 {
   @Override
```

JAVA PROGRAMS

```
    public void run( )
     {
       try{
          for(int i=1; i<=11; i++)
            {
              System.out.println("\tJai  Shri  Raam");
              Thread.sleep(1000);
            }
       }
      catch(InterruptedException ex)
       {
          System.out.print(ex);
       }
   }
 }
class MainClass
 {
   public static void main(String src[]) throws InterruptedException
     {
       Table obj1=new Table();
       A obj2=new A();
       obj1.start();
       obj2.start();
     }
 }
```

JAVA PROGRAMS

Output:

Jai Shri Raam

Enter a number for generate table=9

9*1=9

Jai Shri Raam

9*2=18

Jai Shri Raam

9*3=27

Jai Shri Raam

9*4=36

Jai Shri Raam

9*5=45

Jai Shri Raam

9*6=54

Jai Shri Raam

9*7=63

Jai Shri Raam

9*8=72

Jai Shri Raam

9*9=81

Jai Shri Raam

9*10=90

Jai Shri Raam

Thanks

JAVA PROGRAMS