|  |  |
| --- | --- |
| **Program No:1** | **Date:17/12/18** |
| **Program Title:JAVA program to print twin primes upto 1000** | |

**Problem Statement:**

**Program to check print twin primes upto 1000.**

**Key Concept:**

**Twin primes :pair of primes with differences 2.**

**Program:**

**import java.util.\*;**

**class twinprime**

**{**

**public static void main(String[] args)**

**{**

**twinprime t=new twinprime();**

**for (int i = 2; i < 1000; i++)**

**{**

**if (t.Prime(i) && t.Prime(i + 2))**

**System.out.print("("+i+","+(i+2)+"),");**

**}**

**}**

**boolean Prime(int n)**

**{**

**if (n < 2) return false;**

**for (int i = 2; i <= n / 2; i++)**

**{**

**if (n % i == 0) return false;**

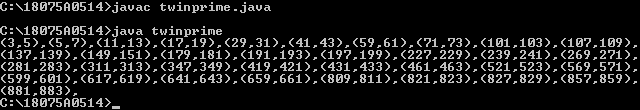
**}**

**return true;**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:2** | **Date:17/12/18** |
| **Program Title:JAVA program to check for Palindrome number.** | |

**Problem Statement:**

**Program to check for Palindrome number;**

**Key Concept:**

**A palindrome number is that remains the same when its digits arereversed.**

**Program:**

**import java.util.\*;**

**public class palindrome**

**{**

**public static void main(String args[]**

**{**

**int n, m, a = 0,x;**

**Scanner s = new Scanner(System.in);**

**System.out.print("Enter any number:");**

**n = s.nextInt();**

**m = n;**

**while(n > 0)**

**{**

**x = n % 10;**

**a = a \* 10 + x;**

**n = n / 10;**

**}**

**if(a == m)**

**System.out.println("Given number "+m+" is Palindrome");**

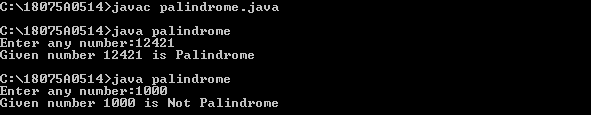
**else**

**System.out.println("Given number "+m+" is Not Palindrome");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:3** | **Date:17/12/18** |
| **Program Title:JAVA program to find the missing element in an array of n natural numbers.** | |

**Problem Statement:**

**Program to find the missing element in an array of n natural numbers.**

**Key Concept:**

**Missing element in given array of natural numbers.**

**Program:**

**import java.util.\*;**

**class missingno**

**{**

**public static void main(String args[])**

**{**

**int n,i;**

**Scanner sc=new Scanner(System.in);**

**System.out.println("enter n:");**

**n=sc.nextInt();**

**int[] ar=new int[n];**

**System.out.println("enter array elements");**

**for(i=0;i<n;i++)**

**{**

**ar[i]=sc.nextInt();**

**}**

**for(i=0;i<n;i++)**

**{**

**if(ar[i]!=i+1)**

**break;**

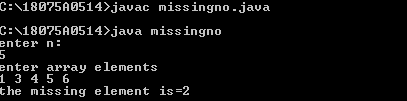
**}**

**System.out.println("the missing element is="+(i+1));**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:4** | **Date:17/12/18** |
| **Program Title:JAVA program to demonstrate all string operations.** | |

**Problem Statement:**

**Program to demonstrate all string operations.**

**Key Concept:**

**Checking == & equals()method for string comparison.**

**Program:**

**import java.util.\*;**

**class stringopr**

**{**

**public static void main(String args[])**

**{**

**String s1="hello world";**

**char c=s1.charAt(0);**

**System.out.println("0th char of s1="+c);**

**System.out.println("length of s1="+s1.length());**

**System.out.println("substring(0,3)="+s1.substring(0,3));**

**String s2="abc";**

**String s3="hello world";**

**String s5="HELLO WORLD";**

**String s4=s1.substring(0,5)+"P!";**

**System.out.println("concatination="+s4);**

**System.out.println("concatination of s1 and s2="+s1.concat(s2));**

**boolean b1=(s1==s2);**

**boolean b2=(s1==s3);**

**boolean b3=s1.equals(s2);**

**boolean b4=s1.equals(s3);**

**boolean b5=s1.equalsIgnoreCase(s5);**

**System.out.println("b1="+b1);**

**System.out.println("b2="+b2);**

**System.out.println("b3="+b3);**

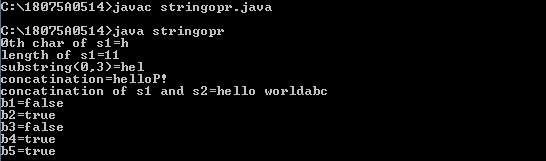
**System.out.println("b4="+b4);**

**System.out.println("b5="+b5);**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:5** | **Date:31/12/18** |
| **Program Title:JAVA program to implement Linear Search.** | |

**Problem Statement:**

**Program to implement Linear Search.**

**Key Concept:**

**For each loop.**

**Program:**

**import java.util.\*;**

**class lsearch**

**{**

**public static void main(String args[])**

**{**

**int ele;**

**Scanner sc=new Scanner(System.in);**

**int[] a={6,5,4,3,2,1};**

**boolean found=false;**

**System.out.println("enter element to be found");**

**ele=sc.nextInt();**

**for(int i:a)**

**{**

**if(i==ele)**

**{**

**found=true;**

**break;**

**}**

**}**

**if(found)**

**System.out.println(ele+" found");**

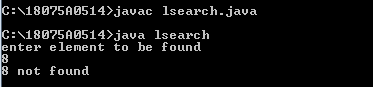
**else**

**System.out.println(ele+" not found");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:6** | **Date:31/12/18** |
| **Program Title:JAVA program to demonstrate break with lable.** | |

**Problem Statement:**

**Program to demonstrate break with label.**

**Key Concept:**

**Break with lable.**

**Program:**

**class breaklable**

**{**

**public static void main(String args[])**

**{**

**one:for(int i=0;i<3;i++)**

**{**

**System.out.println("pass"+i+":");**

**for(int j=0;j<=5;j++)**

**{**

**if(j==5)**

**break one;**

**System.out.print(j+" ");**

**}**

**System.out.println("inner loop");**

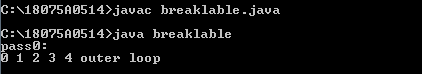
**}**

**System.out.println("outer loop");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:7** | **Date:31/12/18** |
| **Program Title:JAVA program to demonstrate continue with lable.** | |

**Problem Statement:**

**Program to demonstrate continue with lable.**

**Key Concept:**

**Continue with lable.**

**Program:**

**class Contilabel**

**{**

**public static void main(String[] args)**

**{**

**outer: for(int i=0;i<10;i++)**

**{**

**for(int j=0;j<10;j++)**

**{**

**if(j>0)**

**{**

**System.out.println("hai");**

**continue outer;**

**}**

**System.out.print(" "+(i\*j));**

**}**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:8** | **Date:31/12/18** |
| **Program Title:JAVA program to demonstrate wrapper classes.** | |

**Problem Statement:**

**Program to demonstrate wrapper classes.**

**Key Concept:**

**Wrapper classes.**

**Program:**

**import java.util.\*;**

**class wrapper**

**{**

**public static void main(String args[])**

**{**

**byte a = 1;**

**Byte byteobj = new Byte(a);**

**int b = 10;**

**Integer intobj = new Integer(b);**

**float c = 18.6f;**

**Float floatobj = new Float(c);**

**double d = 250.5;**

**Double doubleobj = new Double(d);**

**char e='a';**

**Character charobj=e;**

**System.out.println("Values of Wrapper objects(printing as objects)");**

**System.out.println("Byte object byteobj: " +byteobj);**

**System.out.println("Integer object intobj: " +intobj);**

**System.out.println("Float object floatobj: " +floatobj);**

**System.out.println("Double object doubleobj: " +doubleobj);**

**System.out.println("Character object charobj: " +charobj);**

**byte bv = byteobj;**

**int iv = intobj;**

**float fv = floatobj;**

**double dv = doubleobj;**

**char cv = charobj;**

**System.out.println("Unwrapped values (printing as data types)");**

**System.out.println("byte value, bv: " + bv);**

**System.out.println("int value, iv: " + iv);**

**System.out.println("float value, fv: " + fv);**

**System.out.println("double value, dv: " + dv);**

**System.out.println("char value, cv: " + cv);**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:9** | **Date:31/12/18** |
| **Program Title:JAVA program to implement telephone directory using command line arguments.** | |

**Problem Statement:**

**Program to implement telephone directory using command line arguments.**

**Key Concept:**

**Command line arguments.**

**Program:**

**class telephone**

**{**

**public static void main(String args[])**

**{**

**String s[][]={{"Rohit","9575764759"},{"Shetty",**

**"9054655643"}, {"Singam","9502057776"}};**

**int flag=-1;**

**for(int i=0;i<s.length;i++)**

**{**

**if(s[i][0].equals(args[0]))**

**{**

**System.out.println(s[i][1]);**

**flag=i;**

**break;**

**}**

**}**

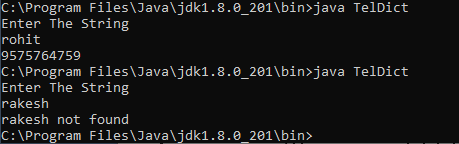
**if(flag==-1)**

**System.out.println("no record found");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:10** | **Date:31/12/18** |
| **Program Title:JAVA program to sort given set of strings.Supply strings at command prompt.** | |

**Problem Statement:**

**Program to sort given set of strings .Supply strings at command prompt.**

**Key Concept:**

**Command line arguments.**

**Program:**

**import java.util.\*;**

**class string**

**{**

**public static void main(String[] args)**

**{**

**int k=args.length;**

**String temp=new String();**

**String names[]=new String[k+1];**

**for(int i=0;i<k;i++)**

**{**

**names[i]=args[i];**

**}**

**for(int i=0;i<k;i++)**

**{**

**for(int j=i+1;j<k;j++)**

**{**

**if(names[i].compareTo(names[j])<0)**

**{**

**temp=names[i];**

**names[i]=names[j];**

**names[j]=temp;**

**}**

**}**

**}**

**System.out.println("sorted order is");**

**for(int i=0;i<k;i++)**

**{**

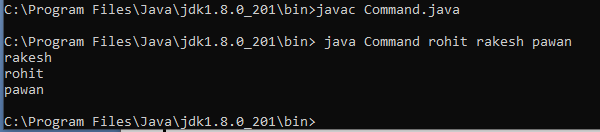
**System.out.println(names[i]);**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:11** | **Date:7/1/19** |
| **Program Title:JAVA program to find product of two matrices.** | |

**Problem Statement:**

**Program to find product of two matrices.**

**Key Concept:**

**Dynamic creation of 2D matrices.**

**Program:**

**import java.util.Scanner;**

**class matrix**

**{**

**public static void main(String args[])**

**{**

**int m, n, p, q, sum = 0, i, j, k;**

**Scanner sc = new Scanner(System.in);**

**System.out.println("Enter the number of rows and columnsof first matrix");**

**m = sc.nextInt();**

**n = sc.nextInt();**

**int first[][] = new int[m][n];**

**System.out.println("Enter elements of first matrix");**

**for (i = 0; i < m; i++)**

**{**

**for (j = 0; j < n; j++)**

**first[i][j] = sc.nextInt();**

**}**

**System.out.println("Enter the number of rows and columns of second matrix");**

**p = sc.nextInt();**

**q = sc.nextInt();**

**if (n != p)**

**System.out.println("The matrices can't be multiplied with each other.");**

**else**

**{**

**int second[][] = new int[p][q];**

**int multiply[][] = new int[m][q];**

**System.out.println("Enter elements of second matrix");**

**for (i = 0; i < p; i++)**

**{**

**for (j = 0; j < q; j++)**

**second[j][j] = sc.nextInt();**

**}**

**for (i = 0; i < m; i++)**

**{**

**for (j = 0; j < q; j++)**

**{**

**for (k = 0; k < p; k++)**

**{**

**sum = sum + first[i][k]\*second[k][j];**

**}**

**multiply[i][j] = sum;**

**sum = 0;**

**}**

**}**

**System.out.println("Product of the matrices:");**

**for (i = 0; i < m; i++)**

**{**

**for (j = 0; j < q; j++)**

**System.out.print(multiply[i][j]+" ");**

**System.out.println();**

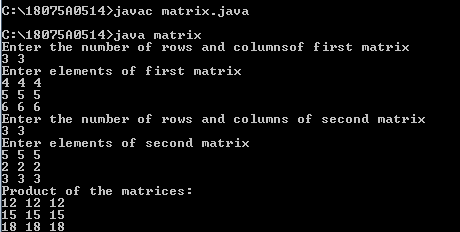
**}**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:12** | **Date:7/1/19** |
| **Program Title:JAVA program to print the pattern using jagged array.** | |

**Problem Statement:**

**Program to print thr pattern using jagged array.**

**Key Concept:**

**Jagged arrays.**

**Program:**

**import java.util.\*;**

**class pattern**

**{**

**public static void main(String[] args)**

**{**

**int n;**

**Scanner SC=new Scanner(System.in);**

**System.out.println("Enter no.of rows to print the pattern");**

**n=SC.nextInt();**

**int[][] a=new int[n][];**

**int t=0;**

**for(int i=0;i<n;i++)**

**{**

**a[i]=new int[i+1];**

**for(int j=0;j<=i;j++)**

**a[i][j]=t+j;**

**t=t+10;**

**}**

**for(int i=0;i<n;i++)**

**{**

**for(int j=0;j<a[i].length;j++)**

**{**

**if(i!=0)**

**System.out.print(a[i][j]+" ");**

**else**

**System.out.print(a[i][j]+"0");**

**}**

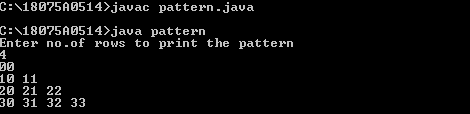
**System.out.println();**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:13** | **Date:7/1/19** |
| **Program Title:JAVA program to perform union,intersection, minus operations on a Arraylist.** | |

**Problem Statement:**

**Program to perform union,intersection,minus operations on a Arraylist.**

**Key Concept:**

**Arraylist provides dynamic arrays in java.**

**Program:**

**import java.util.\*;**

**class arraylist**

**{**

**public static void main(String args[])**

**{**

**int a,b;**

**Scanner sc=new Scanner(System.in);**

**System.out.println("enter sizes of one and two arraylist=");**

**a=sc.nextInt();**

**b=sc.nextInt();**

**ArrayList<Integer> one=new ArrayList<Integer>(a);**

**ArrayList<Integer> two=new ArrayList<Integer>(b);**

**ArrayList<Integer> union=newArrayList<Integer>(a+b);**

**ArrayList<Integer> intersect=new**

**ArrayList<Integer>(a+b);**

**ArrayList<Integer> minu=new ArrayList<Integer>(a);**

**System.out.println("enter elements of one arraylist");**

**for(int i=0;i<a;i++)**

**{**

**int x=sc.nextInt();**

**one.add(x);**

**}**

**System.out.println("enter elements of two arraylist");**

**for(int i=0;i<b;i++)**

**{**

**int x=sc.nextInt();**

**two.add(x);**

**}**

**System.out.println("one="+one);**

**System.out.println("two="+two);**

**union.addAll(one);**

**for(int x:two)**

**{**

**if(one.contains(x))**

**{ }**

**else**

**union.add(x);**

**}**

**System.out.println("arraylist after union="+union);**

**for(int x:one)**

**{**

**if(two.contains(x))**

**intersect.add(x);**

**}**

**System.out.println("arraylist after intersection="+intersect);**

**minu.addAll(one);**

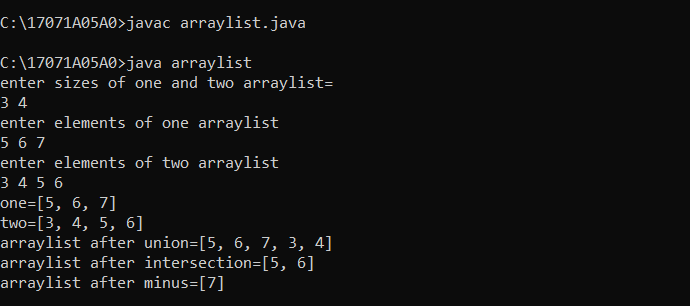
**minu.removeAll(two);**

**System.out.println("arraylist after minus="+minu);**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:14** | **Date:7/1/19** |
| **Program Title:JAVA program to convert array to arraylist.** | |

**Problem Statement:**

**Program to convert array to arraylist.**

**Key Concept:**

**Program:**

**import java.util.\*;**

**class array**

**{**

**public static void main(String args[])**

**{**

**int[] a={1,2,3,4,5,6,7};**

**ArrayList<Integer> A=new ArrayList<Integer>();**

**for(int i=0;i<a.length;i++)**

**A.add(a[i]);**

**System.out.println("after adding array elements to arraylist");**

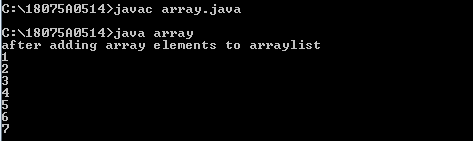
**for(int i=0;i<a.length;i++)**

**System.out.println(A.get(i));**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:15** | **Date:7/1/19** |
| **Program Title:JAVA program to find the pair of elements in an Arraylist whose sum is equals to a specified number.** | |

**Problem Statement:**

**Program to find the pair of elements in an Arraylist whose sum is equals to a specified number.**

**Key Concept:**

**Arraylist creation is dynamic.**

**Program:**

**class sumele**

**{**

**public static void main(String args[])**

**{**

**int i,j,ele,sum=0,n;**

**Scanner sc=new Scanner(System.in);**

**ArrayList<Integer> ar=new ArrayList<Integer>();**

**System.out.println("enter n:");**

**n=sc.nextInt();**

**for(i=0;i<n;i++)**

**{**

**int k=sc.nextInt();**

**ar.add(k);**

**}**

**System.out.println("enter sum element");**

**ele=sc.nextInt();**

**for(i=0;i<n;i++)**

**{**

**for(j=i+1;j<n;j++)**

**{**

**if((ar.get(i))+(ar.get(j))==ele)**

**System.out.println("("+ar.get(i)+","+ar.get(j)+")");**

**}**

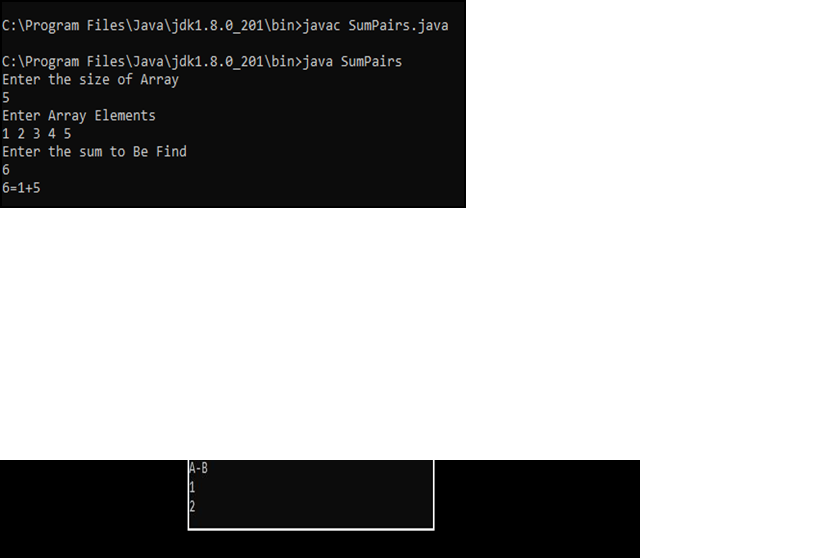
**}**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:16** | **Date:7/1/19** |
| **Program Title:JAVA program to find the Trace of a matrix.** | |

**Problem Statement:**

**Program to find Trace of a matrix.**

**Key Concept:**

**Dynamic creation of 2-D array i.e.,matrix and finding trace of matrix.Trace=sum of diagonal elements of a matrix.**

**Program:**

**import java.util.\*;**

**class trace**

**{**

**public static void main(String args[])**

**{**

**int m,n,i,j,sum=0;**

**Scanner s=new Scanner(System.in);**

**System.out.println("enter n:");**

**n=s.nextInt();**

**int arr[][]=new int[n][n];**

**System.out.println("enter array elements:");**

**for(i=0;i<n;i++)**

**{**

**for(j=0;j<n;j++)**

**{**

**arr[i][j]=s.nextInt();**

**}**

**}**

**for(i=0,j=0;i<n&&j<n;i++,j++)**

**{**

**sum+=arr[i][j];**

**}**

**System.out.println("trace of a matrix="+sum);**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:17** | **Date:7/1/19** |
| **Program Title:JAVA program to remove all duplicate elements in an ArrayList and also return new length**. | |

**Problem Statement:**

**Program to remove all duplicate elements in an ArrayList and also return new length.**

**Key Concept:**

**Arraylist and removing of duplicate elements.**

**Program:**

**import java.util.\*;**

**class duplicate**

**{**

**public static void main(String args[])**

**{**

**int i,j;**

**ArrayList<String> a = new ArrayList<String>();**

**a.add("1");a.add("1");a.add("3");**

**a.add("5");a.add("3");a.add("7");**

**System.out.println("arraylist="+a);**

**System.out.println("length="+a.size());**

**for ( i = 0; i < a.size(); i++)**

**{**

**for(j=i+1;j<a.size();j++)**

**{**

**if (a.get(i).equals(a.get(j)))**

**a.remove(i);**

**}**

**}**

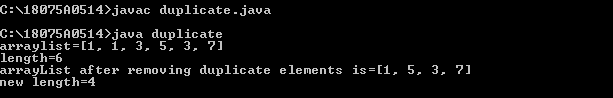
**System.out.println("arrayList after removing duplicate elements is="+a);**

**System.out.println("new length="+a.size());**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:18** | **Date:7/1/19** |
| **Program Title:JAVA program to find all LEADERS in an Arraylist.** | |

**Problem Statement:**

**Program to find all LEADERS in an ArrayList.**

**Key Concept:**

**LEADER is an element if it is greater than all elements of its right side.**

**Program:**

**import java.util.\*;**

**class leader**

**{**

**public static void main(String args[])**

**{**

**int i,j,n,f=0,k;**

**Scanner sc=new Scanner(System.in);**

**ArrayList<Integer> ar=new ArrayList<Integer>();**

**System.out.println("enter n:");**

**n=sc.nextInt();**

**for(i=0;i<n;i++)**

**{**

**k=sc.nextInt();**

**ar.add(k);**

**}**

**System.out.println("LEADERS are");**

**for(i=0;i<n;i++)**

**{**

**f=0;**

**for(j=i+1;j<n;j++)**

**{**

**if((ar.get(i))<(ar.get(j)))**

**{**

**f=-1;**

**break;**

**}**

**}**

**if(f==0)**

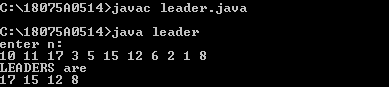
**System.out.print(ar.get(i)+" ");**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:19** | **Date:7/1/19** |
| **Program Title: JAVA program to separate even numbers & odd numbers and place first even numbers, then odd numbers in an ArrayList.** | |

**Problem Statement:**

**Program to separate even numbers & odd numbers and place first even numbers, then odd numbers in an ArrayList.**

**Key Concept:**

**Adding even first numbers and odd numbers after all even numbers in same arraylist.**

**Program:**

**import java.util.\*;**

**class evenodd**

**{**

**public static void main(String args[])**

**{**

**int i,j,n,k;**

**Scanner sc=new Scanner(System.in);**

**ArrayList<Integer> ar=new ArrayList<Integer>();**

**ArrayList<Integer> b=new ArrayList<Integer>();**

**System.out.println("enter n:");**

**n=sc.nextInt();**

**System.out.println("enter array elements");**

**for(i=0;i<n;i++)**

**{**

**k=sc.nextInt();**

**ar.add(k);**

**}**

**k=0;**

**for(i=0;i<n;i++)**

**{**

**if(((ar.get(i))%2)==0)**

**{**

**ar.set(k,ar.get(i));**

**k++;**

**}**

**else**

**b.add(ar.get(i));**

**}**

**for(i=0;i<b.size();i++)**

**{**

**ar.set(k,b.get(i));**

**k++;**

**}**

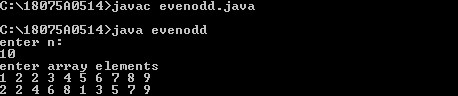
**for(i=0;i<n;i++)**

**System.out.print(ar.get(i)+" ");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:20** | **Date:7/1/19** |
| **Program Title: : JAVA program to find the second most frequent character in a String.** | |

**Problem Statement:**

**Program to find the second most frequent character in a String.**

**Key Concept:**

**Frequency of characters in a string.**

**Program:**

**import java.util.\*;**

**class second**

**{**

**static final int NO\_OF\_CHARS = 256;**

**static char getSecondMostFreq(String str)**

**{**

**int[] count = new int[NO\_OF\_CHARS];**

**int i;**

**for (i=0; i< str.length(); i++)**

**(**

**count[str.charAt(i)])++;**

**int first = 0, second = 0;**

**for (i = 0; i < NO\_OF\_CHARS; i++)**

**{**

**if (count[i] > count[first])**

**{**

**second = first;**

**first = i;**

**}**

**else if (count[i] > count[second] &&**

**count[i] != count[first])**

**second = i;**

**}**

**return (char)second;**

**}**

**public static void main(String args[])**

**{**

**Scanner SC=new Scanner(System.in);**

**String str =SC.nextLine();**

**char res = getSecondMostFreq(str);**

**if (res != '\0')**

**System.out.println("Second most frequent char"+**

**" is " + res);**

**else**

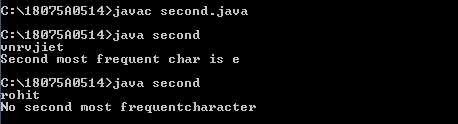
**System.out.println("No second most frequent"+**

**"character");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:21** | **Date:21/1/19** |
| **Program Title: JAVA program to print all permutations of a given string with repetition.** | |

**Problem Statement:**

**Program to print all permutations of a given string with repetition.**

**Key Concept:**

**Permutations of a string.**

**Program:**

**import java.util.\*;**

**class prgm**

**{**

**public static void main(String[] args)**

**{**

**Scanner SC=new Scanner(System.in);**

**System.out.println("Enter String");**

**String s=SC.nextLine();**

**permutationWithRepeation(s);**

**}**

**static void permutationWithRepeation(String str1)**

**{**

**System.out.println("The given string is: PQR");**

**System.out.println("The permuted strings are:");**

**showPermutation(str1,"");**

**}**

**static void showPermutation(String str1, String**

**NewStringToPrint)**

**{**

**if (NewStringToPrint.length() == str1.length())**

**{**

**System.out.println(NewStringToPrint);**

**return;**

**}**

**for (int i = 0; i < str1.length(); i++)**

**{**

**showPermutation(str1, NewStringToPrint +**

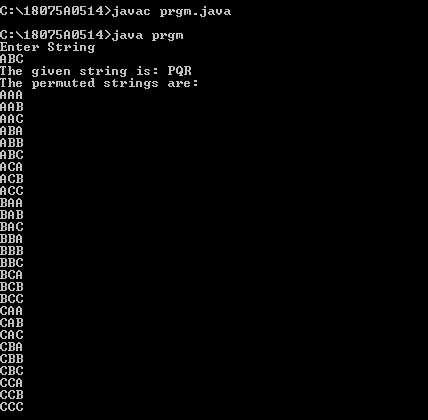
**str1.charAt(i));**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:22** | **Date:21/1/19** |
| **Program Title: JAVA program to implement Telephone directory using HashMap.** | |

**Problem Statement:**

**Program to implement Telephone directory using HashMap.**

**Key Concept:**

**Hashmaps.**

**Program:**

**import java.util.\*;**

**class telephone1**

**{**

**public static void main(String[] args)**

**{**

**Scanner SC=new Scanner(System.in);**

**int flag=-1;**

**HashMap<Integer,String> names = new**

**HashMap<Integer,String>();**

**names.put(1234,"flash");**

**names.put(1235,"atom");**

**names.put(1236,"arrow");**

**names.put(1237,"vibe");**

**names.put(1238,"killerfrost");**

**names.put(1239,"spartan");**

**names.put(1240,"overwatch");**

**names.put(1241,"blacksiren");**

**System.out.println("Enter name of customer to display the respective telephone number");**

**String s= SC.nextLine();**

**for(int i=1234;i<1234+names.size();i++)**

**{**

**if(s.equals(names.get(i)))**

**{**

**flag=1;**

**System.out.println("Telephone number is:"+i);**

**break;**

**}**

**}**

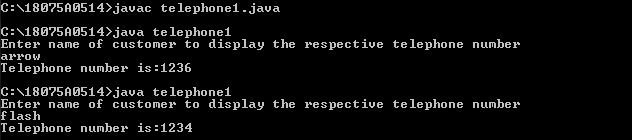
**if(flag==-1)**

**System.out.println("RECORD NOT FOUND");**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:23** | **Date:21/1/19** |
| **Program Title :JAVA program to polynomial addition using LinkedList.** | |

**Problem Statement:**

**Program to polynomial addition using LinkedList.**

**Key Concept:**

**Arthimetic operations on polynomials using Linkedlist.**

**Program:**

**import java.util.\*;**

**class Node**

**{**

**public int exp,coeff;**

**public Node next;**

**public Node(int x,int y)**

**{**

**coeff=x;**

**exp=y;**

**}**

**}**

**class LinkList**

**{**

**public Node first;**

**public LinkList()**

**{**

**first=null;**

**}**

**public void insertFirst(int x,int y)**

**{**

**Node newNode=new Node(x,y);**

**newNode.next=first;**

**first=newNode;**

**}**

**public void insertPos(int x,int y,int p)**

**{**

**Node current=first;**

**Node newNode=new Node(x,y);**

**for(int i=p;i>1;i--)**

**current=current.next;**

**newNode.next=current.next;**

**current.next=newNode;**

**}**

**public void insertLast(int x,int y)**

**{**

**Node newNode=new Node(x,y);**

**newNode.next=null;**

**if(isEmpty())**

**first=newNode;**

**else**

**{**

**Node current=first;**

**while(current.next!=null)**

**current=current.next;**

**newNode.next=current.next;**

**current.next=newNode;**

**}**

**}**

**public boolean find(int key)**

**{**

**Node current=first;**

**while(current!=null)**

**{**

**if(current.exp==key)**

**return true;**

**current=current.next;**

**}**

**return false;**

**}**

**public boolean isEmpty()**

**{**

**return(first==null);**

**}**

**}**

**class Polynomial**

**{**

**private LinkList l1;**

**public Polynomial()**

**{**

**l1=new LinkList();**

**}**

**public boolean insert(int x,int y)**

**{**

**Node current=l1.first;**

**int pos=0;**

**while(current!=null)**

**{**

**if(current.exp==y)**

**{**

**System.out.println("Not a valid term. Insert again");**

**return false;**

**}**

**else if(current.exp<y)**

**break;**

**pos++;**

**current=current.next;**

**}**

**if(pos==0)**

**l1.insertFirst(x,y);**

**else**

**l1.insertPos(x,y,pos);**

**return true;**

**}**

**public void displayPoly()**

**{**

**int f=0;**

**Node current=l1.first;**

**while(current!=null)**

**{**

**if(f!=0&&current.coeff>0&&current.coeff!=-1)**

**System.out.print("+");**

**if(current.coeff!=0)**

**{**

**if(current.coeff>1||current.coeff<-1||current.exp==0)**

**System.out.print(current.coeff);**

**else if(current.coeff==-1)**

**System.out.print("-");**

**if(current.exp==1)**

**System.out.print("X");**

**else if(current.exp>1||current.exp<0)**

**System.out.print("X^"+current.exp);**

**f=1;**

**}**

**current=current.next;**

**}**

**System.out.println("");**

**}**

**public void add(Polynomial poly1,Polynomial poly2)**

**{**

**int x,y;**

**Node current1=poly1.l1.first;**

**Node current2=poly2.l1.first;**

**while(current1!=null&&current2!=null)**

**{**

**if(current1.exp==current2.exp)**

**{**

**x=current1.coeff+current2.coeff;**

**y=current1.exp;**

**current1=current1.next;**

**current2=current2.next;**

**}**

**else if(current1.exp>current2.exp)**

**{**

**x=current1.coeff;**

**y=current1.exp;**

**current1=current1.next;**

**}**

**else**

**{**

**x=current2.coeff;**

**y=current2.exp;**

**current2=current2.next;**

**}**

**l1.insertLast(x,y);**

**}**

**while(current1!=null)**

**{**

**x=current1.coeff;**

**y=current1.exp;**

**current1=current1.next;**

**l1.insertLast(x,y);**

**}**

**while(current2!=null)**

**{**

**x=current2.coeff;**

**y=current2.exp;**

**current2=current2.next;**

**l1.insertLast(x,y);**

**}**

**}**

**}**

**class prgm1**

**{**

**public static void main(String args[])**

**{**

**Scanner SC=new Scanner(System.in);**

**int ch=1;**

**int n,co,ex;**

**while(ch==1)**

**{**

**Polynomial p1=new Polynomial();**

**Polynomial p2=new Polynomial();**

**Polynomial p3=new Polynomial();**

**System.out.println("Enter the no: of terms of 1st polynomial");**

**n=SC.nextInt();**

**while(n!=0)**

**{**

**System.out.println("Enter the coefficent ");**

**co=SC.nextInt();**

**System.out.println("Enter the exponent");**

**ex=SC.nextInt();**

**if(p1.insert(co,ex))**

**n--;**

**}**

**System.out.println("Enter the no: of terms of 2nd polynomial");**

**n=SC.nextInt();**

**while(n!=0)**

**{**

**System.out.println("Enter the coefficent ");**

**co=SC.nextInt();**

**System.out.println("Enter the exponent");**

**ex=SC.nextInt();**

**if(p2.insert(co,ex))**

**n--;**

**}**

**System.out.print("1st Polynomial:- ");**

**p1.displayPoly();**

**System.out.print("2nd Polynomial:- ");**

**p2.displayPoly();**

**p3.add(p1,p2);**

**System.out.print("Added Polynomial:- ");**

**p3.displayPoly();**

**System.out.print("Enter 1 to continue ");**

**ch=SC.nextInt();**

**}**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:24** | **Date:21/1/19** |
| **Program Title :JAVA program to demonstrate Stack.** | |

**Problem Statement:**

**Program to demonstrate Stack.**

**Key Concept:**

**Stack operations-push,pop,peek,isempty,display,search,size.**

**Program:**

**import java.util.\*;**

**public class demostack**

**{**

**public static void main(String args[])**

**{**

**int ele,choice;**

**Stack<Integer> stack=new Stack<>();**

**Scanner sc=new Scanner(System.in);**

**System.out.println("enter 1:to push");**

**System.out.println("enter 2:to pop");**

**System.out.println("enter 3:to find size");**

**System.out.println("enter 4:for top element");**

**System.out.println("enter 5:to search an element");**

**System.out.println("enter 6:to display");**

**System.out.println("enter 7:to find is empty or not");**

**do**

**{**

**System.out.println("enter choice");**

**choice=sc.nextInt();**

**switch(choice)**

**{**

**case 1:**

**System.out.println("enter ele to push into stack");**

**ele=sc.nextInt();**

**stack.push(ele);**

**break;**

**case 2:**

**System.out.println("poped element="+stack.pop());**

**break;**

**case 3:**

**System.out.println("size="+stack.size());**

**break;**

**case 4:**

**System.out.println("top element="+stack.peek());**

**break;**

**case 5:**

**System.out.println("enter element to search");**

**ele=sc.nextInt();**

**int f=stack.search(ele);**

**if(f==-1)**

**System.out.println("element not found");**

**else**

**System.out.println("Element is found at position "**

**+ f);**

**case 6:**

**System.out.println("stack="+stack);**

**break;**

**case 7:**

**System.out.println("is empty:"+stack.isEmpty());**

**break;**

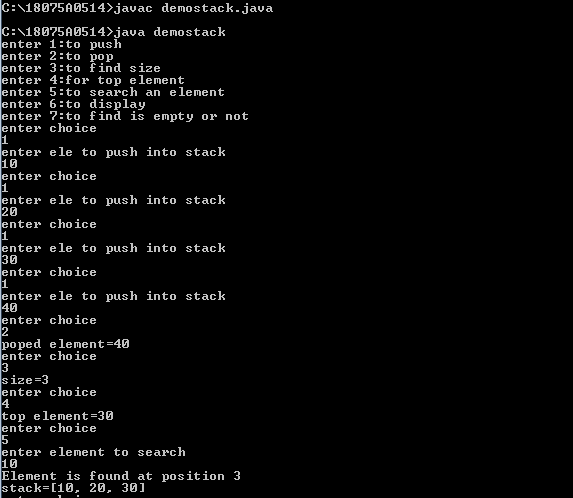
**}**

**}while(choice!=8);**

**}**

**}**

**Input/Output:**

****

|  |  |
| --- | --- |
| **Program No:25** | **Date:21/1/19** |
| **Program Title : JAVA program to demonstrate Vectors.** | |

**Problem Statement:**

**Program to demonstrate Vectors.**

**Key Concept:**

**Vector implements a dynamic array that means it can grow or shrink as required.**

**Program:**

**import java.util.\*;**

**class demovector**

**{**

**public static void main(String args[])**

**{**

**int n;**

**Scanner sc=new Scanner(System.in);**

**System.out.println("enter n:");**

**n=sc.nextInt();**

**Vector<Integer> v=new Vector<Integer>(n);**

**System.out.println("enter elements");**

**for(int i=0;i<n;i++)**

**{**

**int x=sc.nextInt();**

**v.add(x);**

**}**

**System.out.println("size="+v.size());**

**System.out.println("vector="+v);**

**System.out.println("enter element to be search");**

**int ele=sc.nextInt();**

**if(v.contains(ele))**

**System.out.println(ele+" is present at the index "+v.indexOf(ele));**

**else**

**System.out.println("Tiger is not present in thelist.");**

**System.out.println("The first element of the vector is ="+v.firstElement());**

**System.out.println("The last element of the vector is ="+v.lastElement());**

**}**

**}**

**Input/Output:**

