```
Diagonal sorting (both left and right)
 Solution:
import java.util.*;
class Diag sorting
 static Scanner sc-new Scanner(System.in);
 void sort left(int arr[]],int n)
  int i, j, t=0;
  for(i=0;i<n;i++) //sorting left diagonal
    for(j=0;j<(n-1);j++)
        if(arr[j][j]>arr[j+1][j+1])
          t=arr[j][j];
          arr[j][j]=arr[j+1][j+1];
          arr[j+1][j+1]=t;
void sort right(int arr[][],int n)
  int i,j,t=0;
  for(i=0;i<n;i++) //sorting right diagonal
     for(j=0;j<(n-1);j++)
        if(arr[j][2-j]>arr[j+1][(n-1)-(j+1)])
           t=arr[j][(n-1)-j];
           arr[j][(n-1)-j]=arr[j+1][(n-1)-(j+1)];
           arr[j+1][(n-1)-(j+1)]=t;
```

```
void print_matrix(int arr[][].int n)
   for(int i=0;i< n;i++)
      for(int j=0;j< n;j++)
         System.out.print(arr[i][j]+"\t");
       System.out.println();
 public static void main()
  System.out.println("Enter the size of M X N the matrix:");
  int n=sc.nextInt();
  int arr1[][]=new int [n][n];
  System.out.println("Enter matrix elements:");
  for(int i=0;i<n;i++)
   for(int j=0;j< n;j++)
     arr1[i][j]=sc.nextInt();
Diag sorting dl=new Diag_sorting();
System.out.println("....Original Matrix....: ");
d1.print_matrix(arr1,arr1.length);
System.out.println("Matrix....After Sorting left Diagonal: ");
d1.sort left(arr1,arr1.length);
d1.print_matrix(arr1,arr1.length);
System.out.println("Matrix....After Sorting Right Diagonal: ");
dl.sort_right(arrl,arrl.length);
d1.print matrix(arr1,arr1.length);
                    Clockwise Year :2015
```