9. Find Minimum Cost Spanning Tree of a given connected undirected graph using **Prim's algorithm**.

```
import java.util.Scanner;
public class PrimsDemo
{
   public static void main(String [] args)
   {
    int i,j,k,n,source;
    int w[][] = new int[50][50]; //Two dimensional array to store Weight/Cost of the graph
    int visited [] = new int[20];  // Array to keep track of visited and unvisited vertices
    int minWt,totalCost=0,ev=0,sv=0;

    Scanner in = new Scanner(System.in);
    System.out.println("Enter the no of vertices/nodes in the graph");
    n = in.nextInt();

    System.out.println("Enter the weight/cost matrix");
    for(i=1;i<=n;i++)
     {
      for(j=1;j<=n;j++)
      {
         w[i][j] = in.nextInt();
      }
     }
     System.out.println("Enter the source vertex to start");
     source = in.nextInt();

     for(i=1;i<=n;i++)
     {
        visited[i] = 0;  //Initially make all vertices as unvisited
     }
     visited[source] = 1;  //Make the Source vertex as visited

     for(i=1;i<n;i++)
     {
      minWt = 999;
      for(j=1;j<=n;j++)
      {
       if(visited[j] == 1)
       {
        for(k=1;k<=n;k++)
        {
         if( visited[k]!= 1 && w[j][k] < minWt)
         {
            sv = j;
```

```
                ev = k;
                minWt = w[j][k];

              }
            }
          }
        }
        totalCost += minWt;
        visited[ev] = 1;
        System.out.println(sv+ " " + "--->" + " "+ ev + " " + "Cost:"+" "+ minWt);
      }
    System.out.println("The total cost of minimum spanning tree is"+ totalCost);
  }
}
```