PRACTICA SOBRE METODOS DE ORDENACION CON LA ESTRUCTURA STRATEGY

CLASE MAIN:

```
package Strategy;

import java.util.Scanner;


public class Main {


    public static void main(String[] args) {

    Scanner leer = new Scanner(System.in);

        int resp;

        System.out.println("***************** Ingresa los Datos
******************");

        int [] vector = new int [5];

        for(int i= 0;i<vector.length;i++)
        {
            System.out.println("Ingrese el Valor " + (i+1) + " = ");
            vector [i] =leer.nextInt();
        }

        System.out.println("*********** Selecciona un Metodo para ordenarlo
************ ");
        System.out.println("1)    Burbuja ");
        System.out.println("2)    Insercion ");
        System.out.println("3)    Seleccion ");
        System.out.println();
        resp = leer.nextInt();

        switch(resp){

        case 1:
            MetBurbuja b = new MetBurbuja();
            vector = b.ordenar(vector);
            System.out.println();
            System.out.println("Los Valores Ordenados del Vector son: ");
            for(int i=0;i<vector.length;i++)
            {
                System.out.print(vector[i]+" - ");
```

```
            }
            break;


        case 2:

            MetInsercion I = new MetInsercion ();
            vector = I.ordenar(vector);
            System.out.println();
            System.out.println("Los Valores Ordenados del Vector son: ");
            for(int i=0;i<vector.length;i++)
            {
                System.out.print(vector [i]+ " - ");
            }
            break;


        case 3:
            MetSeleccion S = new MetSeleccion ();
            vector = S.ordenar(vector);
            System.out.println();
            System.out.println("Los Valores Ordenados del Vector son: ");
            for(int i=0;i<vector.length;i++)
            {
                System.out.print(vector [i]+ " - ");

            }

            break;

        }
    }

}
```

CLASE METODOS:

```
package Strategy;

abstract class Metodos {

    abstract int [] ordenar (int [] vector);

}
}
```

CLASE METBURBUJA:

```
package Strategy;


public class MetBurbuja extends Metodos{

        int [] ordenar (int[] vector){

        for(int i=0; i<vector.length; i++)
        {
            for(int j=i; j<vector.length; j++)
                if(!(vector[i]<vector[j])){
                    int aux= vector[i];
                    vector[i]=vector[j];
                    vector[j]=aux;
                }
        }

        return vector;
    }
```

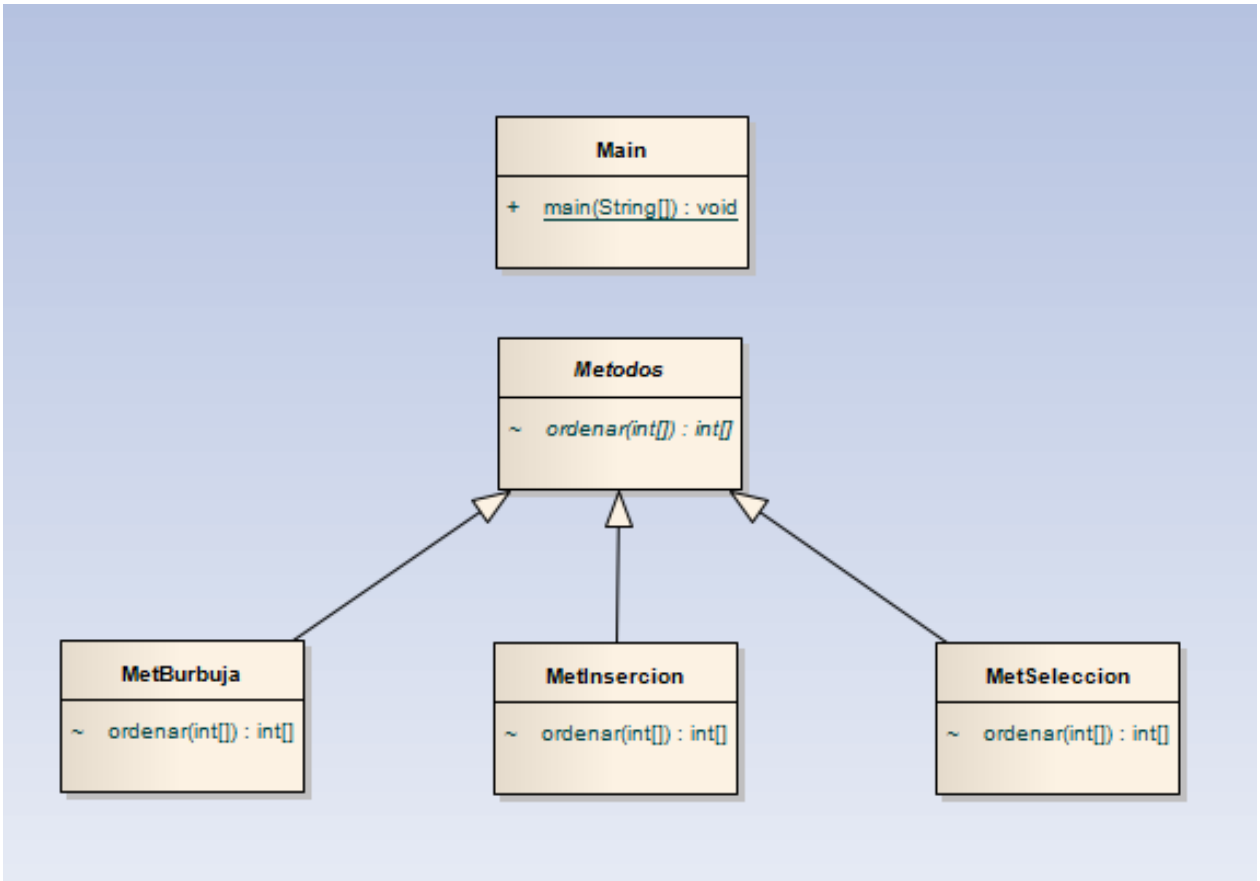
CLASE METINSERCION:

```
package Strategy;

public class MetInsercion extends Metodos {

    int [] ordenar (int[] vector) {
      for (int i=1; i<vector.length; i++) {
         int aux = vector[i];
         int j;
         for (j=i-1; j>=0 && vector[j]>aux; j--)
            vector[j+1] = vector[j];
         vector[j+1] = aux;
      }

      return vector;
   }
```

CLASE METSELECCION:

```
package Strategy;

public class MetSeleccion extends Metodos {

    int [] ordenar (int[] vector)
{
     for (int i = 0; i < vector.length - 1; i++)
     {
          int min = i;
          for (int j = i + 1; j < vector.length; j++)
          {
               if (vector[j] < vector[min])
               {
                    min = j;
               }
          }
          if (i != min)
          {
               int aux= vector[i];
               vector[i] = vector[min];
               vector[min] = aux;
          }
     }

        return vector;
}

}
```

DIAGRAMA DE CLASES


Main
+ main(String[]) : void
Metodos
~ ordenar(int[]) : int[]
MetBurbuja
~ ordenar(int[]) : int[]
MetInsercion
~ ordenar(int[]) : int[]
MetSeleccion
~ ordenar(int[]) : int[]