

```

package operadorConjuntos;

import java.util.Scanner;

public class ConjuntoEnteros {

    static Scanner sc = new Scanner(System.in);
    private boolean elementos[];

    public ConjuntoEnteros() {
        elementos = new boolean[10];
        for (int i = 0; i < 10; i++)
            elementos[i] = false; //por defecto se inicializan a false
    }

    public ConjuntoEnteros union(ConjuntoEnteros c) {
        ConjuntoEnteros caux = new ConjuntoEnteros();
        for (int i = 0; i < elementos.length; i++)
            if (elementos[i] || c.elementos[i])
                caux.elementos[i] = true;
        return caux;
    }

    public ConjuntoEnteros interseccion(ConjuntoEnteros c) {
        ConjuntoEnteros caux = new ConjuntoEnteros();
        for (int i = 0; i < elementos.length; i++)
            if (elementos[i] & c.elementos[i])
                caux.elementos[i] = elementos[i];
        return caux;
    }

    public void insert(int e) {
        elementos[e] = true;
    }

    public void delete(int e) {
        elementos[e] = false;
    }

    public boolean equals(ConjuntoEnteros c) {
        boolean iguales = true;
        int i = 0;
        while (i < elementos.length & iguales)
            if (elementos[i] != c.elementos[i])
                i++;
            else
                iguales = false;
        return iguales;
    }

    public void mostrar() {
        System.out.print("{");
        for (int i = 0; i < elementos.length; i++)
            if (elementos[i])
                System.out.print(i + " ");
        System.out.println("}");
    }
}

```

```

public String toString() {
    String cadaux = ("{"");
    for (int i = 0; i < elementos.length; i++)
        if (elementos[i])
            cadaux += (i + ",");
    cadaux += ("}\\n");
    return cadaux;
}

public void leer() {
    System.out.print("introduce los valores del conjunto, -1 para finalizar");
    int x = -1;
    x = sc.nextInt();
    while (x != -1) {
        elementos[x] = true;
        x = sc.nextInt();
    }
}
}
}

```

*****Clase Operador*****

```

package operadorConjuntos;

import java.util.Scanner;

public class Operador {

    static Scanner sc = new Scanner(System.in);
    private ConjuntoEnteros[] valores;

    public Operador() {
        valores = new ConjuntoEnteros[6];
        for (int i = 0; i < 6; i++)
            valores[i] = new ConjuntoEnteros();
    }

    public void operar() {
        int opcion;

        System.out.println("Bienvenidos al operador de Conjuntos");
        System.out.println(" Opciones disponibles en este operador:");
        System.out.println(" 1: Unión de conjuntos");
        System.out.println(" 2: Intrsección de conjuntos");
        System.out.println(" 3: Insertar un elemneto en un conjunto");
        System.out.println(" 4: Eliminar un elemneto en un conjunto");
        System.out.println(" 5: Comprobar si conjuntos son iguales");
        System.out.println(" 6: Visualizar conjuntos");
        System.out.println(" 7: Salir del operador");
    }
}

```

```

do {

    System.out.println("introducir los conjuntos con los que se quiere operar");
    for (int i = 1; i < valores.length; i++) {
        System.out.println("Introducir conjunto en la posición " + i);
        (valores[i]).leer();
    }

    do {
        System.out.println(" Introducir una opcion o 0 para cambiar de conjuntos");
        System.out.println(" 1: Unión de conjuntos");
        System.out.println(" 2: Intersección de conjuntos");
        System.out.println(" 3: Insertar un elemneto en un conjunto");
        System.out.println(" 4: Eliminar un elemneto en un conjunto");
        System.out.println(" 5: Comprobar si conjuntos son iguales");
        System.out.println(" 6: Visualizar conjuntos");
        System.out.println(" 7: Salir del operador");
        opcion = sc.nextInt();
        switch (opcion) {

            case 0:
                System.out.println("Opcion cambio conjuntos");
                ;
                break;
            case 1:
                this.union();
                break;
            case 2:
                this.interseccion();
                break;
            case 3:
                this.insertar();
                break;
            case 4:
                this.eliminar();
                break;
            case 5:
                this.iguales();
                break;
            case 6:
                this.visualizar2();
                break;
            case 7:
                System.out.println("Operador finalizado");
                Break;// System.exit(0);
            default:
                System.out.println("Opcion incorrecta");
                break;
        }
    } while ((opcion != 0) && (opcion != 7));

} while (opcion != 7);

}

```

```

private void union() {
    int pos = 0;
    System.out.println("Indica las posiciones de los conjuntos a realizar la unión, 0 para finalizar");
    System.out.println("¿Posición del primer conjunto?");
    pos = sc.nextInt();
    while ((pos > 0) & (pos < 6)) {
        valores[0] = valores[0].union(valores[pos]);
        System.out.println("¿Posición del siguiente conjunto? o 0 para finalizar");
        pos = sc.nextInt();
    }
    System.out.println("El resultado de la unión es ");
    valores[0].mostrar();
}

private void interseccion() {
    ConjuntoEnteros aux = new ConjuntoEnteros();
    for (int i = 0; i < 10; i++)
        aux.insert(i);
    int pos = 0;
    System.out.println("Indica las posiciones de los conjuntos a realizar la intersección, 0 para finalizar");
    System.out.println("¿Posición del primer conjunto?");
    pos = sc.nextInt();
    valores[0] = aux;
    while ((pos > 0) & (pos < 6)) {
        valores[0] = valores[0].interseccion(valores[pos]);
        System.out.println("¿Posición del siguiente conjunto? o 0 para finalizar");
        pos = sc.nextInt();
    }
    System.out.println("El resultado de la intersección es ");
    valores[0].mostrar();
}

private void insertar() {
    int pos = 0;
    int num;
    System.out.println("Indica las posiciones de los conjuntos a insertar un elemento, 0 para finalizar");
    System.out.println("¿Posición del primer conjunto?");
    pos = sc.nextInt();
    while ((pos > 0) & (pos < 6)) {
        System.out.println("Escribe el elemento que quieres insertar");
        num = sc.nextInt();
        valores[pos].insert(num);
        System.out.println("¿Posición del siguiente conjunto? o 0 para finalizar");
        pos = sc.nextInt();
    }
}

private void eliminar() {
    int pos = 0;
    System.out.println("Indica las posiciones de los conjuntos a eliminar un elementos, 0 para finalizar");
    System.out.println("¿Posición del primer conjunto?");
    pos = sc.nextInt();
    int num;
    while ((pos > 0) & (pos < 6)) {
        System.out.println("Escribe el elemento que quieres eliminar");
        num = sc.nextInt();
        valores[pos].delete(num);
        System.out.println("¿Posición del siguiente conjunto? o 0 para finalizar");
    }
}

```

```

        pos = sc.nextInt();
        sc.nextLine();
    }
}

private void iguales() {
    int pos1, pos2;
    char otro;
    System.out.println("Indica las posiciones de los conjuntos para comprobar si son iguales, 0 para finalizar");
    System.out.println("¿Posición del primer conjunto?");
    pos1 = sc.nextInt();
    System.out.println("¿Posición del segundo conjunto?");
    pos2 = sc.nextInt();
    if (valores[pos1].equals(valores[pos2])) {
        System.out.println("Estos conjuntos sí son iguales");
        System.out.println("¿introducir otro conjunto para comparar con estos?" + "S/N");
        otro = sc.next().charAt(0);
        while (otro != 'N') {
            System.out.println("introducir posición del siguiente conjuntos a comparar con estos");
            pos2 = sc.nextInt();
            if (valores[pos1].equals(valores[pos2])) {
                System.out.println("Este conjunto sí es igual a los anteriores");
                System.out.println("¿introducir otro conjunto para comparar con estos DEL SI?" + "S/N");
                otro = sc.next().charAt(0);
                System.out.println(otro);
            } else {
                System.out.println("Este conjunto no es igual a los anteriores");
                System.out.println("¿introducir otro conjunto para comparar con estos? DEL NO" + "S/N");
                otro = sc.next().charAt(0);
                System.out.println(otro);
            }
        }
    } else {
        System.out.println("Los conjuntos no son iguales");
    }
}

private void visualizar() {
    System.out.println("El resultado de la ultima operación fue ");
    valores[0].mostrar();
    System.out.println("Los conjuntos operandos son ");
    for (int i = 1; i < valores.length; i++)
        if (valores[i] == null)
            System.out.println("indefinido");
        else
            valores[i].mostrar();
}

private void visualizar2() {
    System.out.println("El resultado de la ultima operación fue ");
    System.out.println(valores[0]);
    System.out.println("Los conjuntos operandos son ");
    for (int i = 1; i < valores.length; i++)
        if (valores[i] == null)
            System.out.println("indefinido");
        else
            System.out.println(valores[i]);
}

```

```
}
```

```
*****Principal*****
```

```
package operadorConjuntos;
```

```
public class Principal {
```

```
    public static void main(String[] args) {
```

```
        Operador c = new Operador();  
        c.operar();
```

```
    }
```

```
}
```