



UNIVERSIDAD PRIVADA FRANZ TAMAYO

DEFENSA HITO 2 - TAREA FINAL

Nombre Completo: **Nicolas Gonzalo Aguilar Arimoza**

Asignatura: **PROGRAMACIÓN III**

Carrera: **INGENIERÍA DE SISTEMAS**

Paralelo: **PROG (1)**

Docente: **Lic. William R. Barra Paredes**

fecha: **30/03/2020**

github: **LINK de la carpeta hito2 de GIT**

- Defina y muestre ejemplos de la clase Scanner

Scanner que nos permitirá leer datos desde ficheros de texto o incluso desde teclado

```
public Scanner leer;  
  
public ejercicios(){  
    leer = new Scanner(System.in);  
}  
  
public void seri1(int n){  
    for(int i = 0; i < n;i++){  
        System.out.println(i);  
    }  
}
```

- Que es la programación orientada a objetos(POO).

una manera de programar específica, donde se organiza el código en unidades denominadas clases, de las cuales se crean objetos que se relacionan entre sí para conseguir los objetivos de las aplicaciones.

- Cual es la diferencia entre interfaz y herencia.

La herencia es un *mecanismo* para extender las funcionalidades y atributos de una clase. Una interfaz es un tipo especial de clase que no posee funcionalidad implementada

- Qué elementos crees que definen a un objeto.

De que tipo son pueden ser int char string array

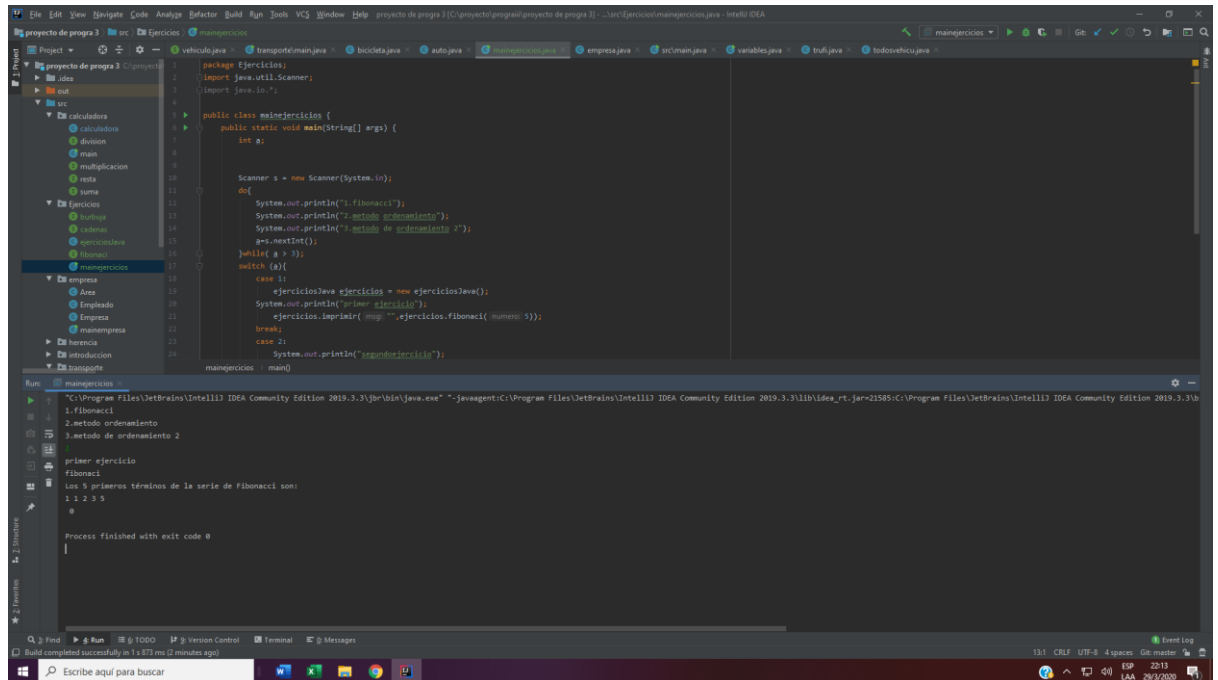
- Que es una clases abstracta y muestre un ejemplo.

Las clases abstractas en Java son clases que se declaran con la palabra “abstract” y pueden o no contener métodos abstractos.

Los métodos abstractos son iguales que en las interfaces, solo son la declaración del método sin el cuerpo.

```
package cursojava.clasesabstractas;  
  
public abstract class OperacionesAbstract {  
  
    public abstract int suma(int a, int b);  
  
    public int resta(int a, int b){  
        return (a - b);  
    }  
}
```

Generar la serie **fibonacci** hasta un valor n leído por teclado



Cree un menú para separar cada ejercicio y coloque para que ingresara un numero por teclado y solo mostrara la cantidad de la serie que coloque

main

```
package Ejercicios;
import java.util.Scanner;

public class mainejercicios {
    public static void main(String[] args) {
        int a;

        Scanner s = new Scanner(System.in);
        do{
            System.out.println("1.fibonacci");
            System.out.println("2.metodo ordenamiento");
            System.out.println("3.metodo de la burbuja 2");
            System.out.println("4.contador de cadenas");
            a=s.nextInt();
        }while( a > 4);
        switch (a) {
            case 1:
                EjerciciosJava ejercicios = new EjerciciosJava();
                System.out.println("fibonaci");
                ejercicios.imprimir("", ejercicios.fibonaci(5));
                break;
            case 2:
                System.out.println("metodo de seleccion");
                EjerciciosJava selectionSort = new EjerciciosJava();
                int[] array = {10, 1, 5, 40, 12, 34, 44, 12, 11, 9};
```

```

        selectionSort.printArray(array);
        selectionSort.sortArray(array);
        selectionSort.printArray(array);
        break;
    case 3:
        System.out.println("metodo burbuja");
        ejerciciosJava burbuja = new ejerciciosJava();
        int nn;
        Scanner sc = new Scanner(System.in);
        System.out.println("Introduzca longitud del array: ");
        nn = sc.nextInt();
        int ArrayN[] = new int[nn];
        for (int i = 0; i < ArrayN.length; i++) {
            System.out.println("Introduzca número para el elemento " +
(i + 1) + ": ");
            ArrayN[i] = sc.nextInt();
        }

        System.out.println("¿Desea ordenar los números? S/N");
        Scanner sc2 = new Scanner(System.in);
        String res = sc2.nextLine();
        if (res.equalsIgnoreCase("S")) {
            System.out.println("");
            burbuja.ordenaBurbuja(ArrayN);
        }
        break;
    case 4:{

        break;
    }
    default:
        System.out.println("ingrese una opcion correcta");
    }
}
}

```

Serie Fibonacci

```
package Ejercicios;

public interface fibonacci {
    public int fibonacci(int numero);
}
```

```
package Ejercicios;

import java.util.Scanner;

public class ejerciciosJava implements fibonacci{

    public int fibonacci(int numero){
        Scanner sc = new Scanner(System.in);
        int fibo1,fibo2,i;
        String resp;
        do{
            System.out.println("fibonacci");
        }while(numero<=1);
        System.out.println("Los " + numero + " primeros términos de la serie de Fibonacci son:");

        fibo1=1;
        fibo2=1;

        System.out.print(fibo1 + " ");
        for(i=2;i<=numero;i++){
            System.out.print(fibo2 + " ");
            fibo2 = fibo1 + fibo2;
            fibo1 = fibo2 - fibo1;
        }

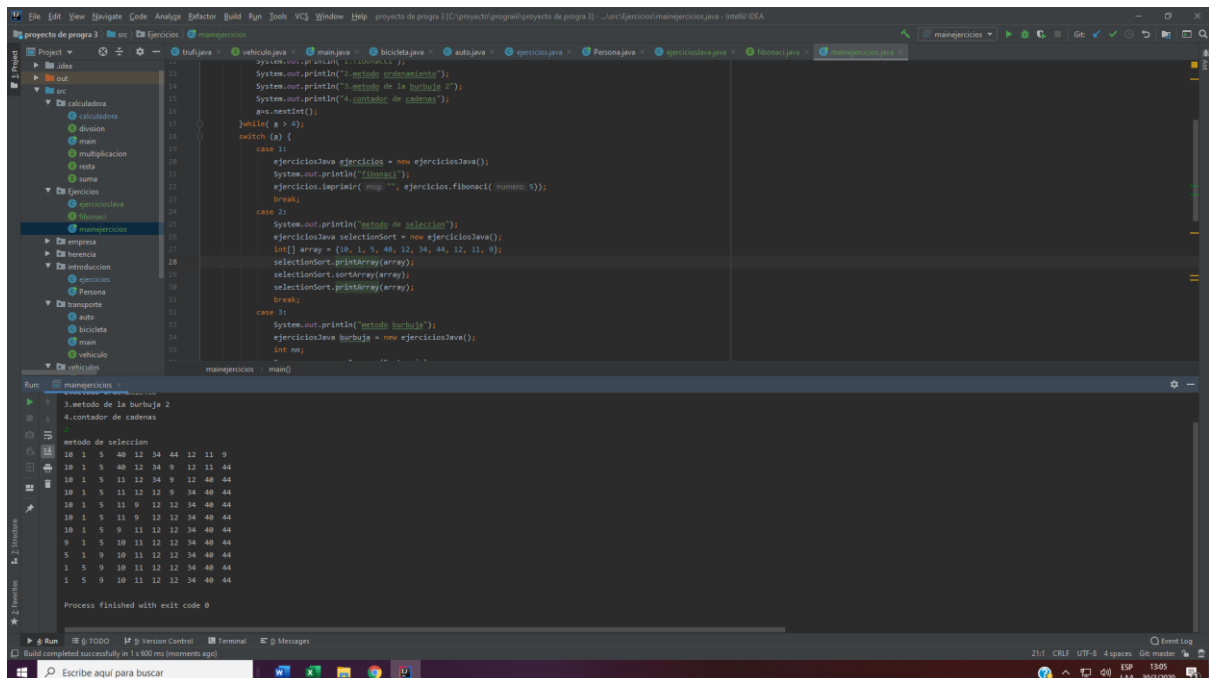
        System.out.println();
        return 0;
    }

    public void imprimir(String msg,int value){
        System.out.printf("%s %d %s",msg,value,"\n");
    }

}
```

Mostrar 2 metodos de ordenacion de vectores.

Método de selección



The screenshot shows an IDE with a project named 'manejercicios'. The code in 'main.java' includes a switch statement with cases for 'metodo de burbuja' and 'metodo de seleccion'. The 'metodo de seleccion' case calls 'ejerciciosJava.selectionSort()' and 'ejerciciosJava.burbuja()'. The 'Run' console shows the output of the selection sort algorithm, displaying the array [10, 1, 5, 40, 12, 34, 44, 12, 11, 9] and its sorted version [1, 5, 9, 10, 11, 12, 12, 34, 40, 44].

lo que se hace es ingresar las variables en el main antes de hacer correr para lograr el ordenamiento que muestra paso a paso como lo resuelve

```
public void sortArray(int[] array) {
    for (int i = array.length - 1; i > 0; i--) {
        int maxValue = 0;
        for (int j = 0; j < i; j++) {
            if (array[j + 1] > array[maxValue]) {
                maxValue = j + 1;
            }
        }
        swap(array, i, maxValue);
        printArray(array);
    }
}

public void printArray(int[] array) {
    for (int i = 0; i < array.length; i++) {
        System.out.printf("%d \t", array[i]);
    }
    System.out.println();
}

public void swap(int[] array, int a, int b) {
    int value = array[b];
    array[b] = array[a];
    array[a] = value;
}
```

Metodo de burbuja

```
public static void ordenaBurbuja(int ArrayN[]) {
    for (int i = 0; i < ArrayN.length - 1; i++) {
        for (int j = 0; j < ArrayN.length - 1; j++) {
            if (ArrayN[j] > ArrayN[j + 1]) {
                int temp = ArrayN[j + 1];
                ArrayN[j + 1] = ArrayN[j];
                ArrayN[j] = temp;
            }
        }
    }
    mostrarN(ArrayN);
}

public static void mostrarN(int ArrayN[]) {
    System.out.println("|-----|");

    for (int i = 0; i < ArrayN.length; i++) { System.out.print(" Elemento " + (i
+ 1) + " -----> " + ArrayN[i] + "\n");
    }
}
```

Todo esta incluido en un main de la primera parte.en este ejercicio utilizo un mostrar para lograr ver el ordenamiento por el método burbuja aparte de el ingreso mediante teclado y la opción de decir si o no para lograr el ordenamiento

