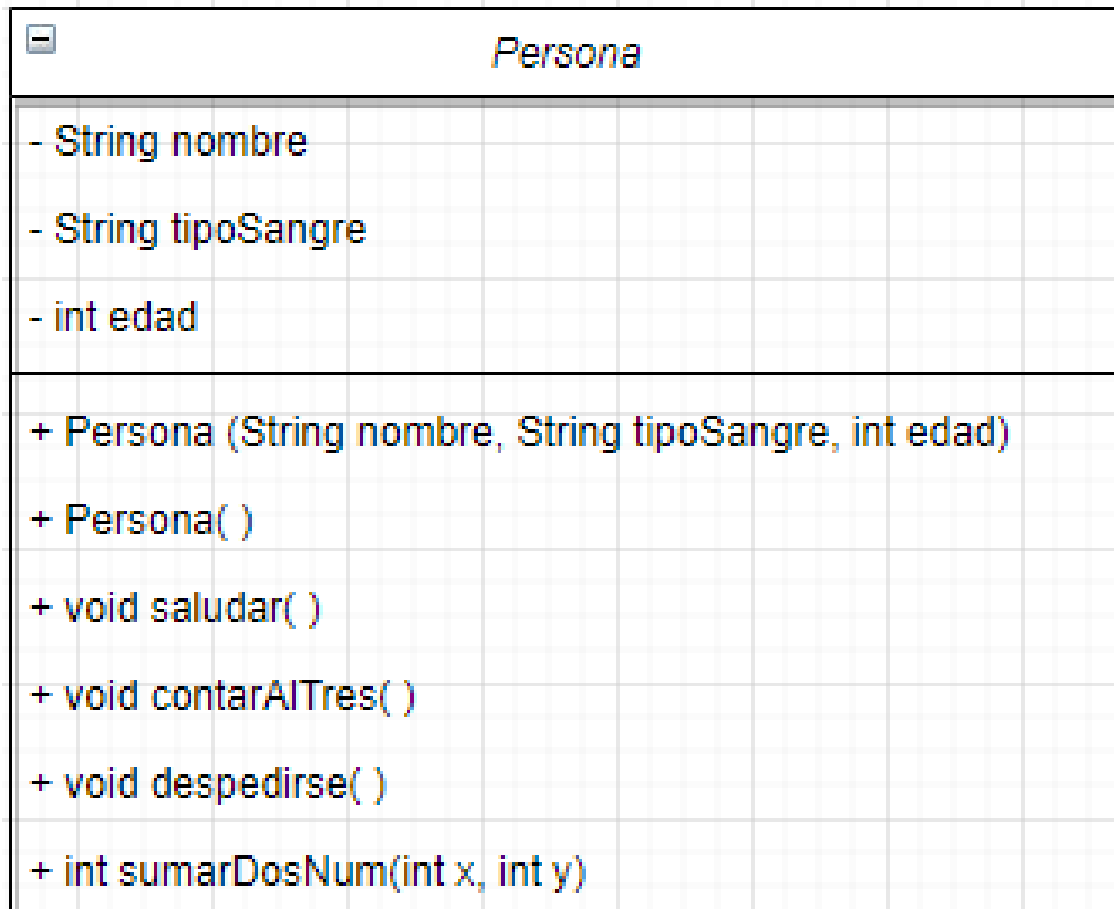


# Mi programa en JAVA

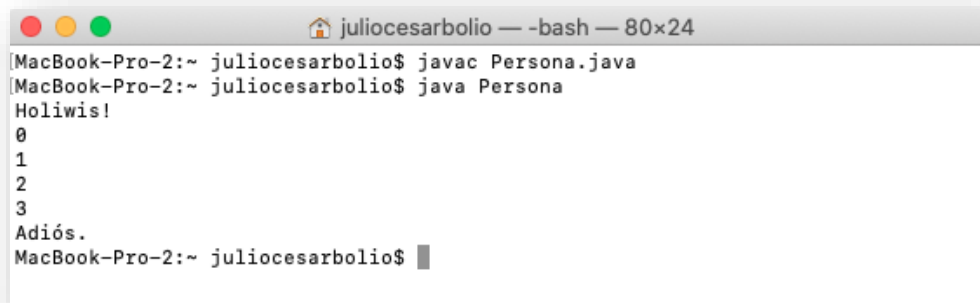
Diagrama de UML



## Código fuente

```
Persona.java
1  public class Persona {
2
3      private String nombre;
4      private String tipoSangre;
5      private int edad;
6
7      public Persona(String nombre, String tipoSangre, int edad){
8          this.nombre = nombre;
9          this.tipoSangre = tipoSangre;
10         this.edad = edad;
11     }
12
13     public Persona(){
14         this.nombre = "X";
15         this.tipoSangre = "0+";
16         this.edad = 0;
17     }
18
19     public void saludar(){
20         System.out.println("Holiwis!");
21     }
22
23     public void contarAlTres(){
24         for (int i = 0; i<4; i++) {
25             System.out.println(i);
26         }
27     }
28
29     public void despedirse(){
30         System.out.println("Adiós.");
31     }
32
33     public int sumarDosNum(int x, int y){
34         return x+y;
35     }
36
37     public static void main(String args[]) {
38         Persona personita = new Persona("César","0-",22);
39         personita.saludar();
40         personita.contarAlTres();
41         personita.sumarDosNum(2, 1);
42         personita.despedirse();
43     }
44 }
45
```

## Ejecución por terminal



```
juliocesarbolio — -bash — 80x24
MacBook-Pro-2:~ juliocesarbolio$ javac Persona.java
MacBook-Pro-2:~ juliocesarbolio$ java Persona
Holiwis!
0
1
2
3
Adiós.
MacBook-Pro-2:~ juliocesarbolio$
```

Al ejecutar la instrucción `javac Persona.java` se crea `Persona.class` como se visualiza en la imagen siguiente. Posteriormente, se ejecuta en la terminal la instrucción `java Persona` la cual ejecuta la clase con el `main` y da como resultado lo que se visualiza en la terminal.

