

Hoja de ejercicios 45_47:

- **Atributos y métodos estáticos.**

45. Crea una clase llamada *Empleado*:

- Con los **atributos privados** *nombre*, y *teléfono*.
- Implementa los **métodos** necesarios para acceder a estos dos atributos de manera **pública** (*getters* y *setters*).
- Añade el **atributo estático y privado** *numeroEmpleados*. Este atributo almacenará el número de instancias que se han creado de la clase *Empleado*.
- Implementa un **método público** para conocer el valor de *numeroEmpleados*.
- Crea una clase aparte con el método *main* para probar todas las funcionalidades de la clase *Empleado*.

46. Crea una clase llamada *MiString*, sin atributos y con los siguientes **métodos públicos y estáticos**:

- *String alReves(String)*: Devuelve la cadena pasada como parámetro pero al revés.
- *String limpiaCaracteres(String, String)*: Devuelve la cadena pasada como primer parámetro pero eliminando los caracteres pasados en la cadena como segundo parámetro.
- *String quitaTildes(String)*: Devuelve la cadena pasada como parámetro pero sin tildes.
- *boolean esPalindromo(String)*: Devuelve *true* o *false*, dependiendo de si es o no palíndromo la frase pasada como parámetro.
- *boolean esNumero(String)*: Devuelve *true* o *false*, dependiendo de si la cadena pasada como parámetro es un número o no.
- Crea una clase aparte con el método *main* para probar todas las funcionalidades de la clase *MiString*.

47. Crea una clase llamada *Número*, sin atributos y con los siguientes **métodos públicos y estáticos**:

- *boolean esPrimo(int)*: Devuelve *true* o *false*.
- *boolean esPerfecto(int)*: Devuelve *true* o *false*.
- *boolean esCapicua(int)*: Devuelve *true* o *false*.
- *boolean sonAmigos(int,int)*: Devuelve *true* o *false* (necesita dos números como parámetro).
- *int[] extraeDivisores(int)*: Devuelve un array con los divisores del número pasado como parámetro.
- *void duplica(int[])*: Duplica el contenido de un array de enteros pasado como parámetro.
- *void ordena(int[],char)*: Ordena el contenido de un array de enteros pasado como primer parámetro. El segundo parámetro indica el tipo de orden, siendo los valores posibles 'A' (ascendente) o 'B' (descendente).
- Crea una clase aparte con el método *main* para probar todas las funcionalidades de la clase *Numero*.