

# Taller

Vamos a hacer la aplicación para la gestión de un taller, para ello tendremos las siguientes clases:

## **FASE1**

### **Persona {a}**

#### Atributos

Todos los atributos tendrán la visibilidad privada.

Texto: dni, nombre y apellidos.

Fecha: fecha de nacimiento.

#### Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

### **Cliente**

Hereda de Persona

#### Atributos

Todos los atributos tendrán la visibilidad privada.

Texto: número de cliente.

#### Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

### **Mecánico**

Hereda de Persona

#### Atributos

Todos los atributos tendrán la visibilidad privada.

Texto: número de la seguridad social.

Decimal: sueldo bruto anual.

## Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

## GestorPersona

### Atributos

Todos los atributos tendrán la visibilidad privada.

Tendremos un array de clientes de 10 posiciones y otro de mecánicos de 5 posiciones.

### Métodos

#### Crear Cliente

Pides por consola todos los datos de un cliente, y lo añades en el array buscando el primer espacio que haya (la primera posición que tenga como valor un null).

#### Borrar Cliente por DNI

Se pide un DNI por consola, se busca en el array de clientes el cliente con el DNI seleccionado, si se encuentra se elimina del array (poner null en vez del valor que tenía anteriormente) y si no existe el cliente en el array se muestra por consola un mensaje indicandolo.

#### Modificar Cliente por DNI

Se solicita por consola un DNI, con este buscamos al cliente que corresponda (si no existe mostrar un mensaje indicandolo). Se solicitan todos los datos del cliente por consola y se modifican del cliente que hemos buscado en el array de Clientes.

#### Crear Mecánico

Igual que "Crear Cliente" pero solicitando los datos del mecánico y añadiendolo en el array de Mecánicos.

#### Borrar Mecánico por DNI

Igual que "Borrar Cliente por DNI" pero buscando y eliminando (si existe) del array de Mecánicos.

#### Modificar Mecánico por DNI

Igual que "Modificar Cliente por DNI" pero utilizando el array de Mecánicos.

#### Calcular sueldo neto mensual de un Mecánico

Se solicita el DNI del mecánico por consola, se busca en el array de mecánicos y si existe se calcula su sueldo mensual neto en base a su sueldo anual. Para calcular esto se eliminará un 15% en concepto de impuestos varios y se dividirá entre 12.

## **Vehículo {a}**

### Atributos

Todos los atributos tendrán la visibilidad privada.

Texto: matricula, color, número de bastidor, tipo de motor

Enteros: cv

### Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

## **Coche**

### Atributos

Todos los atributos tendrán la visibilidad privada.

Enteros: potencia

### Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

## **Moto**

### Atributos

Todos los atributos tendrán la visibilidad privada.

Enteros: cilindrada

### Métodos

Constructor con todos los atributos.

Métodos getter y setter para cada uno de los atributos.

## **GestorVehiculo**

### Atributos

Todos los atributos tendrán la visibilidad privada.

Tendremos un array de vehículos de 20 posiciones.

## Métodos

### Dar de alta vehículo

Se debe añadir al array de vehículos un nuevo coche o moto, para añadir un nuevo vehículo primero se le pide al usuario por consola que tipo de vehículo es y en consecuencia a su respuesta se creará un nuevo coche o una nueva moto que se introducirá en el array de vehículos

### Dar de baja un vehículo por matrícula.

Si hay algún vehículo en el array con esa matrícula, se debe eliminar del array (poner la posición que ocupa el vehículo seleccionado a null). En caso contrario se mostrará un mensaje de vehículo no encontrado.

## Main

Crear un menú en el que podamos utilizar todos los métodos programados anteriormente.

## FASE 2

Vamos a añadir alguna funcionalidad extra a nuestra aplicación.

## Cliente

Añadiremos un atributo privado vehículo de tipo vehículo con los getter y setter para este atributo. Este atributo contendrá el vehículo que tiene el cliente. Un cliente puede no tener ningún vehículo en su posesión en ese caso el atributo será null.

## Mecánico

Añadiremos un atributo vehiculoReparando de tipo vehículo con los getter y setter para este atributo. Este campo contendrá el vehículo que está reparando el mecánico, si no está reparando ninguno el valor de este atributo será null.

## GestionPersona

### Añadir vehículo a cliente

Primero se pregunta el id cliente por consola. Si existe se continúa con el proceso en caso contrario se muestra un error. si el vehículo que se quiere añadir es un coche o una moto y

se piden los datos necesarios por consola. Una vez recogidos los datos se añadirá el vehículo al cliente. En caso de que el cliente ya tenga un vehículo se le preguntará si quiere o no cambiar el vehículo por el nuevo introducido.

## Añadir Vehículo para reparar a Mecánico

Primero se solicita el id de un cliente por consola, después en caso de que el cliente exista, se busca en el array de mecánicos el primer mecánico libre (este será el primero que tenga su atributo `vehiculoReparar` a null. Por último se le asigna el vehículo del cliente al mecánico.

## Terminar reparación

Se solicita por consola el dni del mecánico el cual quiere terminar la reparación si el mecánico tiene un coche asignado a `vehiculoReparar` se le pone null en caso contrario se indica que el mecánico no tiene ningún coche asignado.

## GestionVehiculo

### Añadir vehiculo de cliente a la venta

Este método recibe un cliente por parámetro. Se guarda el coche del cliente en el array de vehículos y se pone a null el vehículo del cliente.

## Main

El main tendrá un menú que llame a los métodos de las clases `GestionPersona` y `GestionVehiculo`.

- 1.-Crear Cliente
- 2.-Modificar Cliente
- 3.-Añadir Vehiculo a cliente
- 4.-Eliminar Vehiculo a cliente
- 5.-Borrar Cliente
- 6.-Crear Mecánico
- 7.-Modificar Mecánico
- 8.-Añadir Vehiculo para reparar a Mecánico
- 9.-Terminar reparación
- 10.-Calcular sueldo mecanico
- 11.-Borrar Mecánico
- 12.-Añadir vehiculo de cliente a la venta
- 13.-Añadir Vehiculo sin cliente
- 14.-Borrar Vehiculo por matricula
- 0.-Salir