

Adrián Tirado Ramos

1º De DAW

Programación

TAREA UD 6

ÍNDICE

CAMBIOS EN EL PROGRAMA.....	2
CAMBIO-001.....	2
CAMBIO-002.....	3
CAMBIO-003.....	5
CAMBIO-004.....	6
CAMBIO-005.....	7

Cambios En El programa

CAMBIO-001

Se comprueba el código original para comprobar que todo funciona correctamente

El código en sí está correcto, pero cuando se crea un vehículo, aparecen varias instancias de texto pidiendo el nombre del dueño del vehículo, el problema que ocurre, es que no se limpia el buffer de texto al hacer un `scan.next()` y hay que agregar un `scan.nextLine()`;

También en caso de error, he inicializado todas las variables vacías para que así, definitivamente el buffer esté limpio

Ambas cosas parecen corregir el error.

```
Introduce la matricula
1234ABC
Introduce el numero de km
50000
Introduce el dia de la fecha de matriculacion
15
Introduce el mes de la fecha de matriculacion
5
Introduce el año de la fecha de matriculacion
2020
Introduce la descripcion
Sedan compacto
Introduce el nombre del propietario
Adrian Tirado
Introduce el nombre del propietario
Adrian Tirado Ramos
Introduce el dni del propietario
12345678A
Introduce el precio
15000
Vehiculo insertado correctamente
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
Elige una opcion
2
Vehiculo{marca=Toyota, matricula=1234ABC, numKM=50000, descripcion=Sedan compacto, precio=15000.0}
```

CAMBIO-002

Se hacen las modificaciones a la clase Vehículo para que sean objetos comparables por matrícula

Complemento la interfaz comparable para poder sobrescribir el método compareTo, equals y hashCode para que se compare por la matrícula

En la clase Vehículo

```

Concesionario.java  Principal.java  Validacion.java  Vehiculo.java x
5
6 public class Vehiculo implements Comparable<Vehiculo> {
7
8     private String marca;
9     private String matricula;
10    private int numKM;
11    private LocalDate fechaMatriculacion;
12    private String descripcion;
13    private double precio;
14    private String nombrePropietario;
15    private String dniPropietario;
16
17    @Override
18    public int compareTo(Vehiculo otroVehiculo) {
19        return this.matricula.compareTo(otroVehiculo.matricula);
20    }
21
22    @Override
23    public boolean equals(Object obj) {
24        if (this == obj) return true;
25        if (obj == null || getClass() != obj.getClass()) return false;
26        Vehiculo vehiculo = (Vehiculo) obj;
27        return matricula.equals(vehiculo.matricula);
28    }
29
30    @Override
31    public int hashCode() {
32        return matricula.hashCode();
33    }
34

```

En la clase Concesionario

```

public Vehiculo buscaVehiculo(String matricula) {
    for (Vehiculo v : vehiculos) {
        if (v.getMatricula().equals(matricula)) {
            return v;
        }
    }
    return null;
}

public int insertarVehiculo(Vehiculo v) {
    if (this.buscaVehiculo(v.getMatricula()) != null) {
        return -2; // El vehiculo existe
    } else {
        vehiculos.add(v);
        return 0; // Vehiculo insertado correctamente
    }
}

public void listarVehiculos() {
    for (Vehiculo v : vehiculos) {
        System.out.println(v.toString());
    }
}

public boolean actualizaKms(String matricula, int kms) {
    Vehiculo vehiculo = buscaVehiculo(matricula);
    if (vehiculo != null) {
        vehiculo.setNumKM(kms);
        return true; // Los kms se han actualizado correctamente
    }
    return false; // No existe el vehiculo con la matricula introducida
}

```

Prueba de que el código funciona

```

19 Vehiculo vehiculo1 = new Vehiculo("Marcal", "AAA2221", 10000, LocalDate.now(), "Descripción1", 10000, "Propietar
20 Vehiculo vehiculo2 = new Vehiculo("Marca2", "AAA2222", 20000, LocalDate.now(), "Descripción2", 20000, "Propietar
21
22 int comparacion = vehiculo1.compareTo(vehiculo2);
23
24 if (comparacion < 0) {
25     System.out.println("vehiculo1 es menor que vehiculo2");
26 } else if (comparacion > 0) {
27     System.out.println("vehiculo1 es mayor que vehiculo2");
28 } else {
29     System.out.println("vehiculo1 es igual a vehiculo2");
30 }
31
32 Scanner scan = new Scanner(System.in);
33
34 boolean salir = false;
35
36

```

Principal [Java Application] C:\Users\adrit\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.10.v20240120-1143\jre\bin\javaw.exe (8 abr 2024, 12:13:41) [pid: 3464]

```

vehiculo1 es menor que vehiculo2
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
6. Eliminar vehiculo
Elige una opcion

```

```

17● public static void main(String[] args) {
18
19     Vehiculo vehiculo1 = new Vehiculo("Marcal", "AAA2222", 10000, LocalDate.now(), "Descripción1", 10000, "Propietar
20     Vehiculo vehiculo2 = new Vehiculo("Marca2", "AAA2222", 20000, LocalDate.now(), "Descripción2", 20000, "Propietar
21
22     int comparacion = vehiculo1.compareTo(vehiculo2);
23
24     if (comparacion < 0) {
25         System.out.println("vehiculo1 es menor que vehiculo2");
26     } else if (comparacion > 0) {
27         System.out.println("vehiculo1 es mayor que vehiculo2");
28     } else {
29         System.out.println("vehiculo1 es igual a vehiculo2");
30     }
31
32     Scanner scan = new Scanner(System.in);
33
34     boolean salir = false;
35
36

```

Principal [Java Application] C:\Users\adrit\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.10.v20240120-1143\jre\bin\javaw.exe (8 abr 2024, 12:14:15) [pid: 14120]

```

vehiculo1 es igual a vehiculo2
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
6. Eliminar vehiculo
Elige una opcion

```

```

17● public static void main(String[] args) {
18
19     Vehiculo vehiculo1 = new Vehiculo("Marcal", "AAE2222", 10000, LocalDate.now(), "Descripción1", 10000, "Propietar
20     Vehiculo vehiculo2 = new Vehiculo("Marca2", "AAA2222", 20000, LocalDate.now(), "Descripción2", 20000, "Propietar
21
22     int comparacion = vehiculo1.compareTo(vehiculo2);
23
24     if (comparacion < 0) {
25         System.out.println("vehiculo1 es menor que vehiculo2");
26     } else if (comparacion > 0) {
27         System.out.println("vehiculo1 es mayor que vehiculo2");
28     } else {
29         System.out.println("vehiculo1 es igual a vehiculo2");
30     }
31
32     Scanner scan = new Scanner(System.in);
33
34     boolean salir = false;
35
36

```

Principal [Java Application] C:\Users\adrit\p2\pool\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64.17.0.10.v20240120-1143\jre\bin\javaw.exe (8 abr 2024, 12:14:43) [pid: 7344]

```

vehiculo1 es mayor que vehiculo2
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
6. Eliminar vehiculo
Elige una opcion

```

CAMBIO-003

Cambio de Array a TreeSet, ya que con esta herramienta de la interfaz Map:

- No pueden existir claves duplicadas
- Mantiene los elementos en orden ascendente, lo que permite al compareTo comparar las matrículas
- TreeSet tiene una mayor eficiencia con funciones básicas como agregar, eliminar y buscar elementos.

```
//CAMBIO-003
private TreeSet<Vehiculo> vehiculos;

public Concesionario() {
    this.vehiculos = new TreeSet<>();
}
```

CAMBIO-004

Se añade la función de eliminar un vehículo, pero para ello, primero comprueba que primero exista ese vehículo previamente.

Archivo Concesionario

```
//CAMBIO-004|
public boolean eliminarVehiculo(String matricula) {
    Vehiculo vehiculo = buscaVehiculo(matricula);
    if (vehiculo != null) {
        vehiculos.remove(vehiculo);
        return true; // Vehículo eliminado correctamente
    }
    return false; // El vehículo no existe
}
```

Archivo Principal

```
//CAMBIO-004
case 6:
    System.out.println("Inserta la matricula del vehículo que deseas eliminar: ");
    matricula = scan.next();
    if (concesionario.eliminarVehiculo(matricula)) {
        System.out.println("Vehículo eliminado correctamente");
    } else {
        System.out.println("No existe un vehículo con la matrícula introducida");
    }
    break;
```

Prueba de su funcionamiento

```
Elige una opcion
2
Vehiculo: {marca=Toyota, matricula=1234ABC, numKM=50000, descripcion=Sedan compacto, precio=15000.0}
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
6. Eliminar vehiculo
Elige una opcion
6
Inserta la matricula del vehículo que deseas eliminar:
1234ABC
Vehículo eliminado correctamente
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
5. Salir
6. Eliminar vehiculo
Elige una opcion
2
1. Nuevo vehiculo
2. Listar vehiculos
3. Buscar vehiculo
4. Modificar kms vehiculo
```


CAMBIO-005

Se añaden los comentarios para que se genere el autodoc de manera correcta