**AE-1. Manejo de Ficheros**

**Ivan Nuñez Rodriguez**

2º DAM

Acceso a Datos

**Índice:**

1. Código.
2. Ejecución y resultado.
3. Enlace a GitHub.

**Código:**

Clase GestorCoches

package controller;  
  
import lombok.Getter;  
import lombok.Setter;  
import model.Coche;  
  
import java.io.\*;  
import java.util.ArrayList;  
import java.util.Scanner;  
  
public class GestorCoches {  
 private final ArrayList<Coche> coches;  
 @Setter  
 @Getter  
 private String nombreArchivoDat = "src/main/java/resources/coches.dat";  
  
 public GestorCoches() {  
 coches = new ArrayList<>();  
 cargarCochesDesdeArchivo();  
 }  
  
 public void cargarCochesDesdeArchivo() {  
 File archivo = new File(nombreArchivoDat);  
 if (archivo.exists()) {  
 try (ObjectInputStream objectInputStream = new ObjectInputStream(new FileInputStream(nombreArchivoDat))) {  
 ArrayList<Coche> cochesCargados = (ArrayList<Coche>) objectInputStream.readObject();  
 coches.clear();  
 coches.addAll(cochesCargados);  
 System.*out*.println("Coches cargados desde el archivo.");  
 } catch (FileNotFoundException e) {  
 System.*err*.println("Archivo no encontrado. Se creará uno nuevo.");  
 } catch (IOException | ClassNotFoundException e) {  
 System.*err*.println("Error al cargar los coches desde el archivo: " + e.getMessage());  
 }  
 } else {  
 System.*out*.println("El archivo no existe, se creará uno nuevo al guardar.");  
 }  
 }  
  
 public void guardarCochesEnArchivo() {  
 try (ObjectOutputStream objectOutputStream = new ObjectOutputStream(new FileOutputStream(nombreArchivoDat))) {  
 objectOutputStream.writeObject(coches);  
 System.*out*.println("Coches guardados en " + getNombreArchivoDat());  
 } catch (IOException e) {  
 System.*err*.println("Error al guardar los coches en el archivo: " + e.getMessage());  
 }  
 }  
  
 public void agregarCoche() {  
 Scanner scanner = new Scanner(System.*in*);  
 Coche coche = new Coche();  
 System.*out*.print("Ingrese el ID del coche: ");  
 int id = scanner.nextInt();  
 if (buscarCochePorId(id) != null) {  
 System.*out*.println("Error: Ya existe un coche con el ID " + id);  
 return;  
 }  
 System.*out*.print("Ingrese la matrícula del coche: ");  
 String matricula = scanner.next();  
 if (buscarCochePorMatricula(matricula) != null) {  
 System.*out*.println("Error: Ya existe un coche con la matrícula " + matricula);  
 return;  
 }  
  
 coche.setId(id);  
 coche.setMatricula(matricula);  
 System.*out*.print("Ingrese la Marca del coche: ");  
 coche.setMarca(scanner.next());  
 System.*out*.print("Ingrese el modelo del coche: ");  
 coche.setModelo(scanner.next());  
 System.*out*.print("Ingrese el color del coche: ");  
 coche.setColor(scanner.next());  
 coches.add(coche);  
 guardarCochesEnArchivo();  
 System.*out*.println("Coche añadido con éxito.");  
 }  
  
 public Coche buscarCochePorMatricula(String matricula) {  
 for (Coche coche : coches) {  
 if (coche.getMatricula().equalsIgnoreCase(matricula)) {  
 return coche;  
 }  
 }  
 return null;  
 }  
  
 public Coche buscarCochePorId(int id) {  
 for (Coche coche : coches) {  
 if (coche.getId() == id) {  
 return coche;  
 }  
 }  
 return null;  
 }  
  
 public void eliminarCocheId(int id) {  
 if (coches.removeIf(coche -> coche.getId() == id)) {  
 System.*out*.println("Coche eliminado.");  
 } else {  
 System.*out*.println("No se encontró el coche con ID: " + id);  
 }  
 guardarCochesEnArchivo();  
 }  
  
 public void mostrarCoches() {  
 if (coches.isEmpty()) {  
 System.*out*.println("No hay coches registrados.");  
 } else {  
 for (Coche coche : coches) {  
 System.*out*.println(coche);  
 }  
 }  
 }  
  
 public void exportarCochesACSV() {  
 String nombreArchivoCsv = "src/main/java/resources/coches.csv";  
 File file = new File(nombreArchivoCsv);  
 try (FileWriter fileWriter = new FileWriter(file)) {  
 fileWriter.write("ID;Marca;Modelo;Color;Matricula\n");  
 for (Coche coche : coches) {  
 fileWriter.write(coche.getId() + ";" + coche.getMarca() + ";" + coche.getModelo() + ";" + coche.getColor() + ";" + coche.getMatricula() + "\n");  
 }  
 System.*out*.println("Coches exportados a " + file);  
 } catch (IOException e) {  
 System.*err*.println("Error al exportar coches a CSV: " + e.getMessage());  
 }  
 }  
  
 public void menu() {  
 Scanner scanner = new Scanner(System.*in*);  
 int opcion;  
 do {  
 System.*out*.println("Introduce qué quieres hacer:");  
 System.*out*.println("1. Añadir coche");  
 System.*out*.println("2. Borrar coche");  
 System.*out*.println("3. Consultar coche por ID");  
 System.*out*.println("4. Listar coches");  
 System.*out*.println("5. Exportar coches a archivo CSV");  
 System.*out*.println("6. Guardar y salir");  
  
 opcion = scanner.nextInt();  
 switch (opcion) {  
 case 1:  
 System.*out*.print("Ingrese Datos");  
 agregarCoche();  
 break;  
 case 2:  
 System.*out*.print("Ingrese el ID del coche a borrar: ");  
 int idBorrar = scanner.nextInt();  
 eliminarCocheId(idBorrar);  
 break;  
 case 3:  
 System.*out*.print("Ingrese el ID del coche a buscar: ");  
 int id = scanner.nextInt();  
 Coche cocheEncontrado = buscarCochePorId(id);  
 if (cocheEncontrado != null) {  
 System.*out*.println("Coche encontrado: " + cocheEncontrado);  
 } else {  
 System.*out*.println("No se encontró ningún coche con el ID " + id);  
 }  
 break;  
 case 4:  
 System.*out*.println("Mostrando coches");  
 mostrarCoches();  
 break;  
 case 5:  
 System.*out*.println("Exportando a CSV");  
 exportarCochesACSV();  
 break;  
 case 6:  
 guardarCochesEnArchivo();  
 System.*out*.println("Saliendo del programa");  
 break;  
 default:  
 System.*out*.println("Opción no válida. Intente de nuevo.");  
 break;  
 }  
 } while (opcion != 6);  
 }  
}

Clase Coche

package model;  
  
import lombok.\*;  
  
import java.io.Serial;  
import java.io.Serializable;  
  
@Getter  
@Setter  
@ToString  
@AllArgsConstructor  
@NoArgsConstructor  
  
public class Coche implements Serializable {  
  
 @Serial  
 private static final long *serialVersionUID* = 2229836618165669008L;  
 private int id;  
 private String matricula;  
 private String marca;  
 private String modelo;  
 private String color;  
  
  
}

Clase Menu

import controller.GestorCoches;  
  
public class Menu {  
 public static void main(String[] args) {  
 GestorCoches gestor = new GestorCoches();  
 gestor.menu();  
 }  
}

**Ejecución y resultado.**

Inicio

**Texto

Descripción generada automáticamente**

Opción añadir coche

**Texto

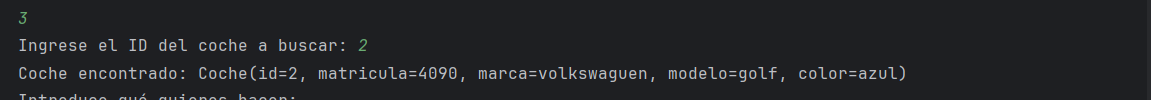
Descripción generada automáticamente**

Opción Listar coche

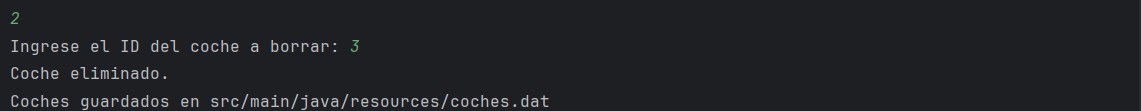
**Texto

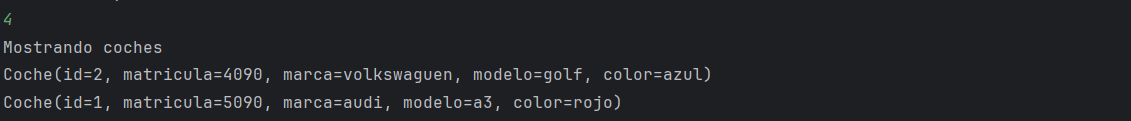
Descripción generada automáticamente**

Opción consultar por ID

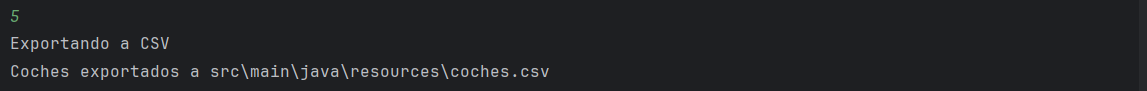
****

Opción borrar coche

****

****

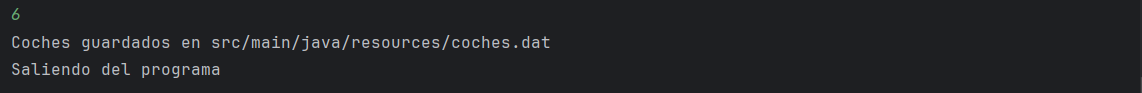
Opción exportar a CSV

****

**Interfaz de usuario gráfica, Texto

Descripción generada automáticamente**

Opción salir y guardar

****

**Enlace a GitHub:**

<https://github.com/Ivannunezrodriguez/Acceso_a_Datos_24-25/tree/94ff5f751d8a8f2ed079fc3cf3de5ab8765d2399/Acitvidades/tema_1/AE-1.%20Manejo%20de%20Ficheros>