



FICHERO DE MULTAS

MARIO JIMÉNEZ MARSET

ÍNDICE

1. ENUNCIADO – OBJETIVOS	3
2. DESARROLLO – PROCEDIMIENTO	3

1. ENUNCIADO – OBJETIVOS

En esta práctica se pedía realizar un programa en Java que crease un fichero de texto o binario para gestionar las multas de la DGT, almacenando en ellas el número de multa, el número de agente, la localidad, el coste y si está pagada o no.

Para tratar estas multas, se creaba un menú en consola de comandos que gestionara las operaciones sobre las multas:

- Crear multa.
- Eliminar multa.
- Modificar multa.
- Pagar multa.
- Consultar todas las multas.
- Buscar multas por agente.
- Buscar multas por número de multa.
- Buscar multas por localidad.

2. DESARROLLO – PROCEDIMIENTO

En primer lugar, se muestra el código de cada clase creada, explicando el cometido de cada una.

Clase Principal:

```
package multasprincipio;
import java.io.*;
import java.util.*;
public class Principal {
    static final File fichero=new File("C:\\Users\\jimen\\eclipse-
workspace\\Multas\\ficheroMultas.txt");
    ListaMultas listamultas=new ListaMultas();
    Scanner entrada=new Scanner(System.in);
    Principal(){
    }
    public static void main(String[] args) throws Exception {
        Principal p=new Principal();
        p.menu();
    }
    public void menu() throws Exception {
        MetodosMultas metodosmultas=new MetodosMultas();
        LeerMultas leermulta=new LeerMultas();
        metodosmultas.cargarListaMultas(fichero);
        int eleccion;
        do {
            System.out.println("Bienvenido al Menu de Multas");
            try {
                Thread.sleep(500);
            }catch (InterruptedException e) {
                e.printStackTrace();
            }
            System.out.println("Elige entre las siguientes
opciones\n1-Crear multa\n2-Eliminar multa\n3-Guardar multas\n4-Modificar multa"
+ "\n5-Pagar multa\n6-Consultar las
multas\n7-Buscar multas por agente\n8-Buscar multas por numMulta"
+ "\n9-Buscar multas por localidad\n10-
Salir del Menu");
            eleccion=entrada.nextInt();
            switch(eleccion) {
```

```

        case 1:
            metodosmultas.crearMulta();
            break;
        case 2:
            metodosmultas.eliminarMulta();
            break;
        case 3:
            metodosmultas.guardarMulta(fichero);
            break;
        case 4:
            metodosmultas.modificarMulta();
            break;
        case 5:
            metodosmultas.pagarMulta();
            break;
        case 6:
            leermulta.leer(fichero);
            break;
        case 7:
            metodosmultas.buscarMultaNumeroAgente();
            break;
        case 8:
            metodosmultas.buscarMultaIdMulta();
            break;
        case 9:
            metodosmultas.buscarMultaLocalidad();
            break;
        case 10:
            System.out.println("Saliendo del
Menu...");
            System.out.println("Ejecucion del
Programa finalizada");
            break;
    }
}while(eleccion!=10);
entrada.close();
}
}

```

En esta clase, se crea el fichero con su correspondiente ruta absoluta, haciendo que sea static para que no cambie en ningún momento.

Se crea un objeto de la clase ListaMultas. En el main, se crea un objeto de la propia clase y se llama al método menú para ejecutarse.

En este método también se crean objetos de las clases MetodosMultas y LeerMulta. En primer lugar, se llama al método que carga las multas del fichero.

Después de eso, se crea el menú de consola demandado; a través de un switch, se elige la opción que se quiera. Por ejemplo, primero se crea la multa, se guarda y se puede consultar: hecho esto, las demás opciones ya pueden elegirse para realizar su función, pero no antes de realizar estos pasos. Además, se puede salir del menú terminando la ejecución.

Ejemplo Resultado:

```

1
Crear Multa
Variables de la Multa
Numero de Agente
1
Localidad
Madrid
Coste
34,5
Pagada-->true
No Pagada-->false
true
|Creando Multa...
Multa Creada

```

En estas capturas se muestra cómo se crea una multa y cómo se guarda en el arraylist (el cual es volcado en el fichero).

```

3
Guardar Multa
Deseas guardar todas las multas creadas?\nSi\nNo
si
|Fichero guardado en la ruta C:\Users\jimen\eclipse-workspace\Multas\ficheroMultas.txt
Todas las multas han sido guardadas

```

```

6
|Las Multas almacenadas son:
Multa{NumMulta=1, NumAgente=1, Localidad=Madrid, Coste=34.5, Pagada=true}

```

```

2
Eliminar Multa
Cual es el idMulta?
1
Multa eliminada con exito

```

Al consultar la multa creada, se muestra la misma. Sin embargo, posteriormente se elimina y, si se vuelve a elegir la opción de consultar, ya no existirá esa multa.

```

4
Modificar Multa
Cual es el idMulta de la multa a modificar?
1
Que variable de la multa deseas modificar?
1-NumAgente
2-Localidad
3-Coste
4-Pagada,NoPagada
3
Introduce el nuevo Coste
56

```

En la siguiente captura aparece la opción de modificar multa. Por ejemplo, en este caso se ha elegido cambiar el número de coste.

Al volver a consultar la multa, se ha cambiado su coste correctamente.

```

6
Las Multas almacenadas son:
Multa{NumMulta=2, NumAgente=2, Localidad=Barcelona, Coste=437.89, Pagada=false}
Multa{NumMulta=1, NumAgente=1, Localidad=Madrid, Coste=56.0, Pagada=true}

```







```

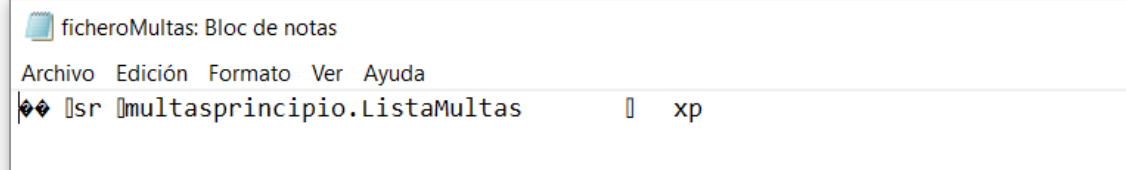
5
Pagar Multa
Cual es el idMulta de la multa a verificar su pago?
2
Esta multa esta pendiente de pago
El coste de la misma es de 437.89
Pagando...
Multa Pagada

```

Al pagar la multa, esta cambia el boolean 'Pagada' de false a true.

Finalmente, los apartados 7, 8 y 9 del menú realizan la misma función, cuyo procedimiento es buscar según el número de agente, de multa o localidad la multa o multas asociadas a ello. Al fin y al cabo, es lo mismo que consultar multas, pero con filtros.

	.settings	01/12/2022 10:09	Carpeta de archivos	
	bin	19/12/2022 10:28	Carpeta de archivos	
	src	01/12/2022 10:09	Carpeta de archivos	
	.classpath	01/12/2022 10:09	Archivo CLASSPATH	1 KB
	.project	01/12/2022 10:09	Archivo PROJECT	1 KB
	ficheroMultas	19/12/2022 11:17	Documento de tex...	1 KB



Dentro del fichero, se guarda el arraylist con las multas serializado.

GuardarMulta:

```
package multasprincipio;
import java.io.*;

public class GuardarMulta {
    ListaMultas listamultas=new ListaMultas();
    public void guardar(File fichero) {
        try(ObjectOutputStream oos = new
ObjectOutputStream(new FileOutputStream(fichero));){
            oos.writeObject(listamultas);
        }catch(IOException e) {
            System.out.println("Error al ESCRIBIR");
            e.printStackTrace();
        }
        System.out.println("Fichero guardado en la ruta
"+fichero.getAbsolutePath());
    }
}
```

A través del ObjectOutputStream, dentro del cual se crea un FileOutputStream con el fichero como parámetro, se escribe objeto listamultas de la clase ListaMultas, el cual es el arraylist de multas. Se muestra un mensaje de la ruta donde se ha guardado el fichero.

LeerMulta:

```
package multasprincipio;
import java.io.*;

public class LeerMulta {
    ListaMultas listamultas=new ListaMultas();
    public void leer(File fichero) throws IOException,
ClassNotFoundException{
        ObjectInputStream ois=null;
        if(fichero.exists()) {
            ois=new ObjectInputStream(new
FileInputStream(fichero));
            try {
                listamultas
                =(ListaMultas)ois.readObject();
                System.out.println("Las Multas
almacenadas son:");
                for(Multa m:
listamultas.getLista()) {
                    System.out.println(m);
                }
            }catch(EOFException e) {
                System.out.println("Final del
Fichero");
            }finally {
                ois.close();
            }
        }
    }
}
```

En esta clase se hace el mismo proceso que en la anterior, cambiando que para leer se utiliza el `ObjectInputStream` y el `FileInputStream`. Además, para mostrar las multas se recorre el `arraylist` con un `for each`.

ListaMultas:

```
package multasprincipio;
import java.io.*;
import java.util.*;
public class ListaMultas implements Serializable{
    private static final long serialVersionUID = 1L;
    private static ArrayList<Multa>lista=new ArrayList<>();
    public void insertarMulta(Multa multa) {
        lista.add(multa);
    }
    public ArrayList<Multa> getLista() {
        return lista;
    }
    @SuppressWarnings("static-access")
    public void setLista(ArrayList<Multa> lista) {
        this.lista = lista;
    }
}
```

En esta clase, simplemente se crea el `arraylist` de almacenamiento de multas, además del método que permite insertar multas dentro de ese `arraylist`, además de su `getter` y `setter`.

MetodosMultas:

```
package multasprincipio;
import java.util.*;

import java.io.*;
public class MetodosMultas {
    Scanner entrada=new Scanner(System.in);
    ListaMultas listamultas=new ListaMultas();
    public void crearMulta() {
        System.out.println("Crear Multa\nVariables de la
Multa\nNumero de Agente");
        int numAgente=entrada.nextInt();
        entrada.nextLine();
        System.out.println("Localidad");
        String localidad=entrada.nextLine();
        System.out.println("Coste");
        double coste=entrada.nextDouble();
        System.out.println("Pagada-->true\nNo Pagada-->false");
        boolean pagada=entrada.nextBoolean();
        System.out.println("Creando Multa...");
        Multa m=new
Multa(numAgente,localidad,coste,pagada);
        this.listamultas.insertarMulta(m);
        System.out.println("Multa Creada");
    }
}
```


Esta clase contiene todos los métodos a llamar desde la clase Principal para realizar una acción. El primer método consiste en crear multa, el cual se hace preguntando los datos de la multa a través de consola y, teniendo esos datos, crear el objeto multa, el cual será insertado en el arraylist de multas.

```

public void eliminarMulta() {
    System.out.println("Eliminar Multa\nCual es el idMulta?");
    int id=entrada.nextInt();
    int posicion=0;
    for(int i=0;i<listamultas.getList().size();i++) {

        if(id==listamultas.getList().get(i).getNumMulta()) {
            posicion=i;
            System.out.println("Multa
elimnada con exito");
        }
    }
    listamultas.getList().remove(posicion);
}

```

El método de eliminar multa busca, a través del id multa pasado por el usuario la multa que debe eliminar. Al encontrarla, la elimina del arraylist de multas (removiendo la posición donde se alojaba esa multa).

```

public void guardarMulta(File fichero) {
    GuardarMulta gm=new GuardarMulta();
    System.out.println("Guardar Multa\nDeseas guardar todas las
multas creadas?\nSi\nNo");
    entrada.nextLine();
    String respuesta=entrada.nextLine();
    if(respuesta.equalsIgnoreCase("si")) {
        gm.guardar(fichero);
    }
    System.out.println("Todas las multas han sido guardadas");
}

```

El método de guardar multa llama al método de la clase GuardarMulta (el cual escribe el arraylist de multas en el fichero).

```

public void modificarMulta() {
    System.out.println("Modificar Multa\nCual es el idMulta de la
multa a modificar?");
    int id=entrada.nextInt();
    int posicion=0;
    for(int i=0;i<listamultas.getList().size();i++) {

        if(id==listamultas.getList().get(i).getNumMulta()) {
            posicion=i;
        }
    }
    ModificarMulta mm=new ModificarMulta();
    mm.modificar(posicion);
}

```

El método de modificar multa encuentra, a través del idMulta pasado por el usuario, la multa a modificar. Teniéndola controlada, llama a un método de la clase ModificarMulta, el cual realiza la modificación.

```

        public void pagarMulta() {
            System.out.println("Pagar Multa\nCual es el idMulta de la multa
a verificar su pago?");
            int id=entrada.nextInt();
            int posicion=0;
            for(int i=0;i<listamultas.getList().size();i++) {
                if(id==listamultas.getList().get(i).getNumMulta()) {
                    posicion=i;
                }
            }
            PagarMulta mpm=new PagarMulta();
            mpm.pagar(posicion);
        }

```

El método de pagar multa, al igual que el método anterior, encuentra la multa a través de un idMulta e invoca un método de la clase PagarMulta, el cual hace el pago.

```

        public void buscarMultaNumeroAgente() {
            System.out.println("Buscar Multas por Numero de Agente\nCual
es el numAgente de la multa o multas a encontrar?");
            int numAgente=entrada.nextInt();
            int posicion=0;
            System.out.println("Las multas son:");
            for(int i=0;i<listamultas.getList().size();i++) {
                if(numAgente==listamultas.getList().get(i).getNumAgente()) {
                    posicion=i;
                }
            }
            System.out.println(listamultas.getList().get(posicion));
        }

```

El método de buscar multa por número de agente demanda al usuario el número de agente con multas asociadas. Según el número de agente, se busca dentro del arraylist las multas con ese número de agente, imprimiéndose por consola (si es que existen).

Los dos siguientes métodos son exactamente iguales a este (cambiando el número de agente por el idMulta o la localidad.)

```

        public void buscarMultaldMulta() {
            System.out.println("Buscar Multas por Numero de Multa\nCual
es el idMulta de la multa o multas a encontrar?");
            int idMulta=entrada.nextInt();
            int posicion=0;
            System.out.println("Las multas son:");
            for(int i=0;i<listamultas.getList().size();i++) {
                if(idMulta==listamultas.getList().get(i).getNumMulta()) {
                    posicion=i;
                }
            }

```

```

        System.out.println(listamultas.getList().get(posicion));
    }
}

public void buscarMultasLocalidad() {
    System.out.println("Buscar Multas por Localidad\nCual es la
localidad de la multa o multas a encontrar?");
    entrada.nextLine();
    String localidad=entrada.nextLine();
    int posicion=0;
    System.out.println("Las multas son:");
    for(int i=0;i<listamultas.getList().size();i++) {

        if(localidad.equalsIgnoreCase(listamultas.getList().get(i).getLocalidad())) {
            posicion=i;

            System.out.println(listamultas.getList().get(posicion));
        }
    }

    public ListaMultas cargarListaMultas(File fichero) throws Exception {
        System.out.println("Cargando Multas...");
        DataInputStream dis=new DataInputStream(new
FileInputStream(fichero));
        ObjectInputStream ois=new ObjectInputStream(dis);
        listamultas=(ListaMultas)ois.readObject();
        ois.close();
        dis.close();
        for(Multa m:listamultas.getList()) {
            System.out.println(m);
        }
        System.out.println("Multas cargadas con exito");
        return listamultas;
    }
}

```

Por último, el método de cargar multas trata de volcar las multas ya guardadas en el fichero. Con DataInputStream se lee el arraylist almacenado en el fichero. Con ObjectInputStream se pasa por parámetro este objeto DataInputStream. El objeto del ObjectInputStream entonces es leído con readObject y casteado por ListaMultas. Se recorre este arraylist con un for each para ver su contenido.

ModificarMulta:

```

package multasprincipio;

import java.util.Scanner;

public class ModificarMulta {
    Scanner sc=new Scanner(System.in);
    ListaMultas listamultas=new ListaMultas();
    public void modificar(int posicion) {
        System.out.println("Que variable de la multa deseas
modificar?\n1-NumAgente\n2-Localidad\n3-Coste\n4-Pagada,NoPagada");
        int opcion=sc.nextInt();
        switch(opcion) {
            case 1:
                System.out.println("Introduce el nuevo
NumAgente");
                int
nuevoNumAgente=sc.nextInt();
                Multa
multaAgente=new
Multa(listamultas.getList().get(posicion).getNumMulta(),nuevoNumAgente,listamultas.g
etLista().get(posicion).getLocalidad(),listamultas.getList().get(posicion).getCoste(),
listamultas.getList().get(posicion).isPagada());
                listamultas.getList().remove(posicion);
                listamultas.getList().add(multaAgente);
                break;
            case 2:
                System.out.println("Introduce la nueva
Localidad");
                sc.nextLine();
                String nuevaLocalidad=sc.nextLine();
                System.out.println("Cambiando
localidad...");
                Multa multaLocalidad=new
Multa(listamultas.getList().get(posicion).getNumMulta(),listamultas.getList().get(posici
on).getNumAgente(),nuevaLocalidad, listamultas.getList().get(posicion).getCoste(),
listamultas.getList().get(posicion).isPagada());
                listamultas.getList().remove(posicion);
                listamultas.getList().add(multaLocalidad);
                System.out.println("LOCALIDAD
CAMBIADA");
                break;
            case 3:
                System.out.println("Introduce el nuevo
Coste");
                double
nuevoCoste=sc.nextDouble();
                Multa multaCoste=new
Multa(listamultas.getList().get(posicion).getNumMulta(),listamultas.getList().get(posici
on).getNumAgente(),listamultas.getList().get(posicion).getLocalidad(), nuevoCoste,

```

```

        listamultas.getList().get(posicion).isPagada());

        listamultas.getList().remove(posicion);

        listamultas.getList().add(multaCoste);
                                break;
                                case 4:
                                    boolean cambio;

        if(listamultas.getList().get(posicion).isPagada()==true) {
                                                    cambio=false;
                                                }
                                                else {
                                                    cambio=true;
                                                }
                                                Multa
multaCambio=new
Multa(listamultas.getList().get(posicion).getNumMulta(),listamultas.getList().get(posici
on).getNumAgente(),listamultas.getList().get(posicion).getLocalidad(),

        listamultas.getList().get(posicion).getCoste(),cambio);

        listamultas.getList().remove(posicion);

        listamultas.getList().add(multaCambio);
                                break;
                                }
        }
    }
}

```

En esta clase, se crea un método, el cual será el que se llame en otras clases. A través de un switch, se elige qué opción escoger: si cambiar el número de agente, la localidad, el coste o si está pagada o no. Los cuatro casos son iguales, exceptuando si se cambia la localidad, el número de agente...

El procedimiento es crear una nueva multa con la nueva variable introducida por el usuario y los otros datos que no se quieren cambiar: se remueve del arraylist entonces la multa antigua y se añade la nueva.

Multa:

```

package multasprincipio;
import java.io.*;
public class Multa implements Serializable{
    private static final long serialVersionUID = 1L;
    private int numMulta;
    private static int contador=1;
    private int numAgente;
    private String localidad;
    private double coste;
    private boolean pagada;
    public Multa() {

    }
    public Multa(int numAgente, String localidad, double coste, boolean
pagada) {

```

```

        this.numMulta = contador;
        this.numAgente = numAgente;
        this.localidad = localidad;
        this.coste = coste;
        this.pagada = pagada;
        contador++;
    }
    public Multa(int numMulta, int numAgente, String localidad, double
coste, boolean pagada) {
        this.numMulta = numMulta;
        this.numAgente = numAgente;
        this.localidad = localidad;
        this.coste = coste;
        this.pagada = pagada;
    }

    public int getNumMulta() {
        return numMulta;
    }
    public void setNumMulta(int numMulta) {
        this.numMulta = numMulta;
    }
    public int getNumAgente() {
        return numAgente;
    }
    public void setNumAgente(int numAgente) {
        this.numAgente = numAgente;
    }
    public String getLocalidad() {
        return localidad;
    }
    public void setLocalidad(String localidad) {
        this.localidad = localidad;
    }
    public double getCoste() {
        return coste;
    }
    public void setCoste(double coste) {
        this.coste = coste;
    }
    public boolean isPagada() {
        return pagada;
    }
    public void setPagada(boolean pagada) {
        this.pagada = pagada;
    }
    public String toString() {
        return "Multa{"+"NumMulta="+numMulta+",
NumAgente="+numAgente+", Localidad="+localidad+", Coste="+coste+",
Pagada="+pagada+"}";
    }
}

```

En esta clase se establecen las variables, constructores, getters, setters y método toString sobrescrito necesarios para realizar el programa. Se establece un contador static en el constructor, el cual será el número de multa. Cada vez que se cree una multa, así se hará que no se repita el id (en incremento constante, nunca será el mismo).

```

PagarMulta:

package multasprincipio;

public class PagarMulta {
    ListaMultas listamultas=new ListaMultas();
    public void pagar(int posicion) {
        boolean cambio;
        if(listamultas.getList().get(posicion).isPagada()==true)
        {
            System.out.println("La multa ya esta pagada");
        }
        else {
            System.out.println("Esta multa esta pendiente
de pago");
            System.out.println("El coste de la misma es de
"+listamultas.getList().get(posicion).getCoste());
            try {

                System.out.println("Pagando...");

                Thread.sleep(3000);
                cambio=true;
                Multa
                multaCambio=new
                Multa(listamultas.getList().get(posicion).getNumMulta(),listamultas.getList().get(posici
on).getNumAgente(),listamultas.getList().get(posicion).getLocalidad(),

                listamultas.getList().get(posicion).getCoste(),cambio);

                listamultas.getList().remove(posicion);

                listamultas.getList().add(multaCambio);

                System.out.println("Multa
Pagada");
            }catch (InterruptedException e) {
                e.printStackTrace();
            }
        }
    }
}

```

En esta clase se pagan las multas. Si ya está a true el boolean 'Pagada' de la multa, sólo saldrá una frase informando de que ya está pagada la multa. Si está a false, se crea una nueva multa idéntica a la antigua, cambiando el false por un true. Se remueve del arraylist la multa antigua y se inserta la nueva.