



P4: Accés a fitxers

Binaris.

Exercici 1

Crea un programa que guardi 5 temperatures (valors double) en un fitxer binari anomenat temperatures.bin.

Després, fes un mètode que llegeixi aquest fitxer i mostri les temperatures per pantalla.

```
1 package pl;  
2  
3 import java.io.FileOutputStream;  
4 import java.io.IOException;  
5 import java.io.DataInputStream;  
6 import java.io.DataOutputStream;  
7 import java.io.FileInputStream;  
8  
9 public class binaris_exercici_01 {  
10  
11     public static void main(String[] args) {  
12  
13         String nomFitxer = "temperatures.bin";  
14         double[] temperatures = {17.5, 18.8, 19.2, 16.6, 21.8};  
15         try {  
16  
17             FileOutputStream fitxer;  
18             DataOutputStream dades;  
19             fitxer = new FileOutputStream(nomFitxer);  
20             dades = new DataOutputStream(fitxer);  
21  
22             for (double num : temperatures) {  
23                 dades.writeDouble(num);  
24             }  
25             llegirTemperatures(nomFitxer);  
26  
27             dades.close();  
28             fitxer.close();  
29  
30         } catch (IOException e) {  
31             e.printStackTrace();  
32         }  
33     }  
34  
35     public static void llegirTemperatures(String nomFitxer) {  
36  
37         try {  
38             FileInputStream fitxer2;  
39             DataInputStream dades2;  
40             fitxer2 = new FileInputStream(nomFitxer);  
41             dades2 = new DataInputStream(fitxer2);  
42  
43             for (int i = 0; i < 5; i++) {  
44                 double numero = dades2.readDouble();  
45                 System.out.println("Número llegit: " + numero);  
46             }  
47  
48             dades2.close();  
49             fitxer2.close();  
50  
51         } catch (IOException e) {  
52             e.printStackTrace();  
53         }  
54     }  
55 }  
56
```

Console Output:

```
Número llegit: 17.5  
Número llegit: 18.8  
Número llegit: 19.2  
Número llegit: 16.6  
Número llegit: 21.8
```

Per començar importem les classes necessàries per treballar amb fitxers binaris en Java:

```
import java.io.FileOutputStream;  
import java.io.IOException;  
import java.io.DataInputStream;  
import java.io.DataOutputStream;  
import java.io.FileInputStream;
```

En el mètode main declarem una variable "nomFitxer" que conté el nom del fitxer binari "temperatures.bin". I també definim un array de valors del tipus double on hi hauran les temperatures que després s'escriuran al fitxer:

```
public class binaris_exercici_01 {  
    public static void main(String[] args) {
```

```
        String nomFitxer = "temperatures.bin";  
        double[] temperatures = {17.5, 18.8, 19.2, 16.6, 21.8};
```

Comencem un bloc try per capturar possibles excepcions durant l'escriptura i lectura del fitxer. I declarem dos objectes "FileOutputStream" anomenat "fitxer" serveix per escriure bytes al fitxer i "DataOutputStream" anomenat "dades" que serveix per escriure dades en format binari.

```
        try {  
            FileOutputStream fitxer;  
            DataOutputStream dades;
```

Iniciem "fitxer" amb un nou "FileOutputStream" que obre el fitxer "temperatures.bin" per introduir-li dades i un altre amb "DataOutputStream" per poder escriure valors double directament.

```
        fitxer = new FileOutputStream(nomFitxer);  
        dades = new DataOutputStream(fitxer);
```

Amb un bucle for recorrem cada valor del array "temperatures" i els escrivim en format binari en el fitxer amb "writeDouble()"

```
        for (double num : temperatures) {  
            dades.writeDouble(num);  
        }
```

Un cop escrit el fitxer, cridem el mètode per llegir i mostrarà les dades de l'arxiu.

```
        llegirTemperatures(nomFitxer);
```

Tanquem tant "dades" com "fitxer" per assegurar-nos de que tot es guarda correctament. I capturem les excepcions amb el "catch" per si es produeix algun error.

```
        dades.close();  
        fitxer.close();  
    } catch (IOException e) {  
        e.printStackTrace();  
    }  
}
```

Declarem el mètode que llegeix i mostra les temperatures del fitxer, iniciant un bloc "try" per capturar possibles errors de lectura.

```
public static void llegirTemperatures(String nomFitxer) {  
    try {
```

Declarem els fluxos de lectura:

```
        FileInputStream fitxer2;  
        DataInputStream dades2;  
        fitxer2 = new FileInputStream(nomFitxer);  
        dades2 = new DataInputStream(fitxer2);
```

I amb un bucle for que s'executarà cinc vegades, el número de temperatures guardades, anirem llegint les dades amb "readDouble()" i les mostrarem per pantalla:

```
for (int i = 0; i < 5; i++) {  
    double numero = dades2.readDouble();  
    System.out.println("Número llegit: " + numero);  
}
```

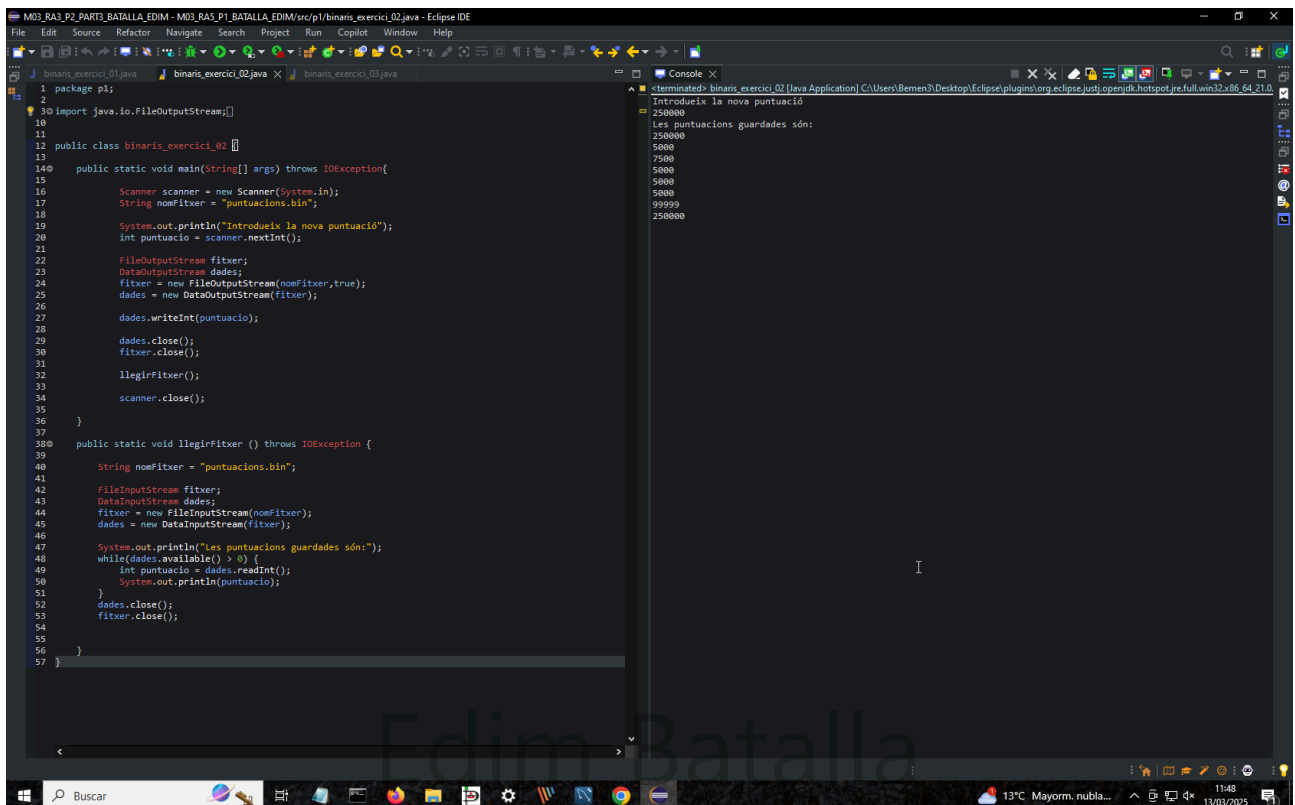
Tanquem els fluxos de lectura per alliberar recursos i capturem possibles errors amb el "catch"

```
dades2.close();  
fitxer2.close();  
} catch (IOException e) {  
    e.printStackTrace();  
}  
}  
}
```

Exercici 2

Desenvolupa un programa que guardi puntuacions (int) d'un videojoc en un fitxer binari puntuacions.bin.

Cada cop que s'executi, ha de permetre afegir una nova puntuació i mostrar totes les puntuacions guardades fins al moment.



```
1 package pl;  
2  
3 import java.io.FileOutputStream;  
4  
5  
6  
7  
8  
9  
10  
11  
12 public class binaris_exercici_02 {  
13  
14     public static void main(String[] args) throws IOException {  
15  
16         Scanner scanner = new Scanner(System.in);  
17         String nomFitxer = "puntuacions.bin";  
18  
19         System.out.println("Introdueix la nova puntuació");  
20         int puntuacio = scanner.nextInt();  
21  
22         FileOutputStream fitxer;  
23         DataOutputStream dades;  
24         fitxer = new FileOutputStream(nomFitxer,true);  
25         dades = new DataOutputStream(fitxer);  
26  
27         dades.writeInt(puntuacio);  
28  
29         dades.close();  
30         fitxer.close();  
31  
32         llegirFitxer();  
33  
34         scanner.close();  
35  
36     }  
37  
38     public static void llegirFitxer () throws IOException {  
39  
40         String nomFitxer = "puntuacions.bin";  
41  
42         FileInputStream fitxer;  
43         DataInputStream dades;  
44         fitxer = new FileInputStream(nomFitxer);  
45         dades = new DataInputStream(fitxer);  
46  
47         System.out.println("Les puntuacions guardades són:");  
48         while(dades.available() > 0) {  
49             int puntuacio = dades.readInt();  
50             System.out.println(puntuacio);  
51         }  
52         dades.close();  
53         fitxer.close();  
54  
55     }  
56 }  
57 }
```

```
<terminated> binaris_exercici_02 [Java Application] C:\Users\Beren\...  
Introdueix la nova puntuació  
250000  
Les puntuacions guardades són:  
250000  
50000  
5000  
5000  
5000  
99999  
250000
```

Després de importar les classes necessàries en el mètode main hi afegim "throws IOException" que és un mètode que indica que es pot generar alguna excepció d'entrada/sortida.

```
public static void main(String[] args) throws IOException{
```

Creem un "Scanner" i definim el nom del fitxer binari:

```
Scanner scanner = new Scanner(System.in);  
String nomFitxer = "puntuacions.bin";
```

Demanem a l'usuari que introdueixi una puntuació i la guardem en una variable de integers.

```
System.out.println("Introdueix la nova puntuació");  
int puntuacio = scanner.nextInt();
```

Declarem els fluxos de sortida per escriure dades en un fitxer amb el detall de que en el "FileOutputStream hi afegim el mode "true" per afegir dades al fitxer existent sense sobreescriure.

```
FileOutputStream fitxer;  
DataOutputStream dades;  
fitxer = new FileOutputStream(nomFitxer,true);  
dades = new DataOutputStream(fitxer);
```

Escrivim el valor de la puntuació amb la funció "writeInt" i tancem dades i fitxer per assegurar que es guardi tot correctament. Cridem el mètode llegir Fitxer i tancem l'escàner.

```
dades.writeInt(puntuacio);  
dades.close();  
fitxer.close();  
llegirFitxer();  
scanner.close();
```

En el mètode llegirFitxer igual que en el main hi afegim "throws IOException" indicant que es pot generar alguna excepció d'entrada/sortida. I definim el nom de l'arxiu a llegir:

```
public static void llegirFitxer () throws IOException {  
String nomFitxer = "puntuacions.bin";
```

Obrim el fitxer per la lectura i mostrem les puntuacions guardades.

```
FileInputStream fitxer;  
DataInputStream dades;  
fitxer = new FileInputStream(nomFitxer);  
dades = new DataInputStream(fitxer);  
System.out.println("Les puntuacions guardades són:");
```

Amb un bucle while indiquem que mentre el nombre de dades sigui major a zero vol dir que encara hi han dades al fitxer per llegir, les llegirem i les mostrarem per pantalla fins que s'hagin llegit totes.

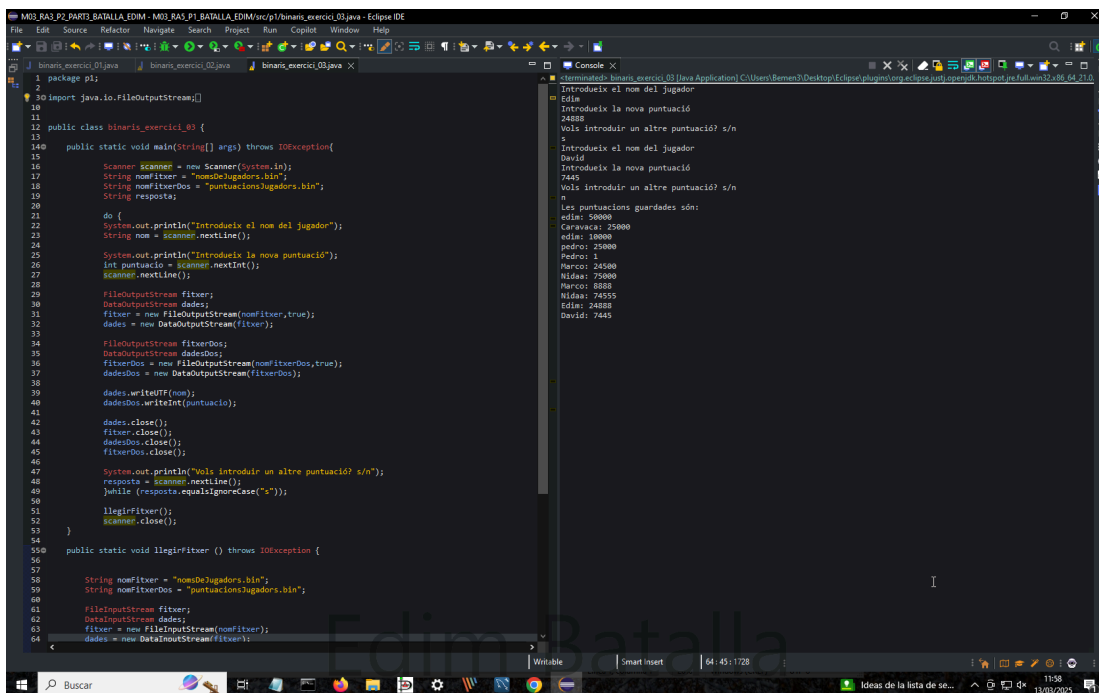
```
while(dades.available() > 0) {  
int puntuacio = dades.readInt();  
System.out.println(puntuacio);
```

Exercici 3

Fes un sistema on hi hagi dos fitxers binaris:

- jugadors.bin per desar noms de jugadors (String).
- puntuacions.bin per desar puntuacions (int).

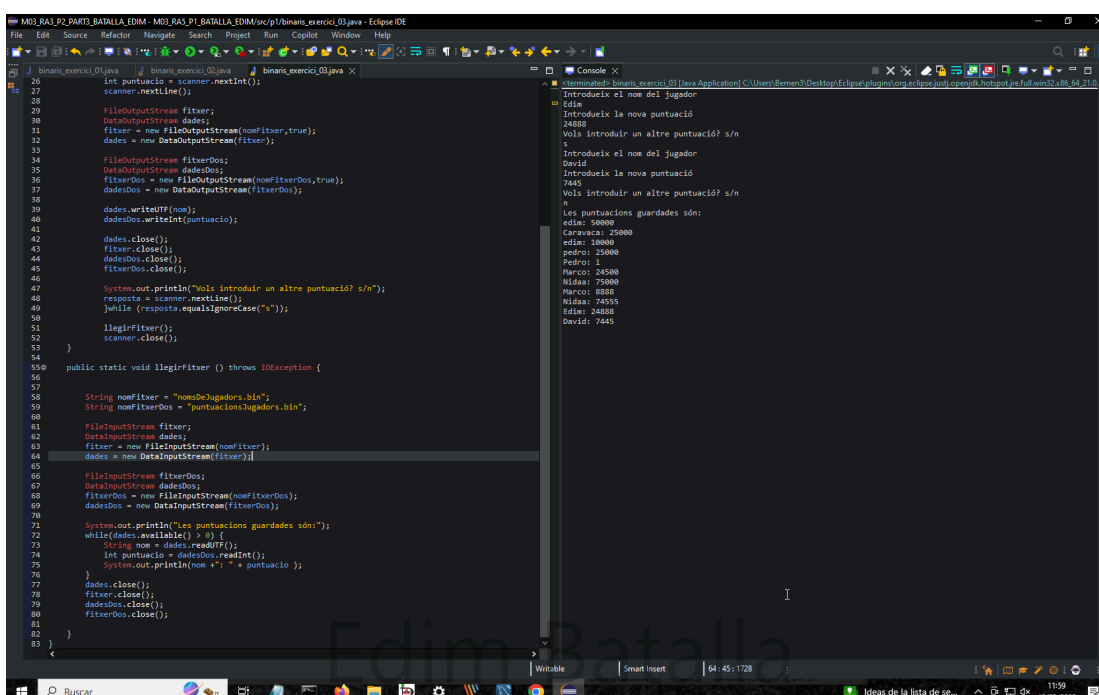
El programa ha de permetre afegir un jugador amb la seva puntuació i després mostrar tots els jugadors amb la seva puntuació corresponent. Els fitxers han de mantenir l'ordre, de manera que la primera línia de jugadors.bin correspongui a la primera puntuació de puntuacions.bin, la segona a la segona, etc.



```
1 package pl;
2
3 import java.io.FileOutputStream;
4
5 public class binaris_exercici_03 {
6
7     public static void main(String[] args) throws IOException {
8
9         Scanner scanner = new Scanner(System.in);
10         String nomFitxer = "nomsDeJugadors.bin";
11         String nomFitxerDos = "puntuacionsJugadors.bin";
12         String resposta;
13
14         do {
15             System.out.println("Introdueix el nom del jugador");
16             String nom = scanner.nextLine();
17
18             System.out.println("Introdueix la nova puntuació");
19             int puntuacio = scanner.nextInt();
20             scanner.nextLine();
21
22             FileOutputStream fitxer;
23             DataOutputStream dades;
24             fitxer = new FileOutputStream(nomFitxer, true);
25             dades = new DataOutputStream(fitxer);
26
27             FileOutputStream fitxerDos;
28             DataOutputStream dadesDos;
29             fitxerDos = new FileOutputStream(nomFitxerDos, true);
30             dadesDos = new DataOutputStream(fitxerDos);
31
32             dades.writeUTF(nom);
33             dades.writeInt(puntuacio);
34             dades.close();
35             fitxer.close();
36             dadesDos.close();
37             fitxerDos.close();
38
39             System.out.println("Vols introduir un altre puntuació? s/n");
40             resposta = scanner.nextLine();
41             while (resposta.equalsIgnoreCase("s"));
42
43             lllegirFitxer();
44             scanner.close();
45
46         } while (resposta.equalsIgnoreCase("s"));
47
48     }
49
50     public static void lllegirFitxer () throws IOException {
51
52         String nomFitxer = "nomsDeJugadors.bin";
53         String nomFitxerDos = "puntuacionsJugadors.bin";
54
55         FileInputStream fitxer;
56         DataInputStream dades;
57         fitxer = new FileInputStream(nomFitxer);
58         dades = new DataInputStream(fitxer);
59
60         while (dades.available() > 0) {
61             String nom = dades.readUTF();
62             int puntuacio = dades.readInt();
63             System.out.println(nom + ": " + puntuacio);
64         }
65
66         dades.close();
67         fitxer.close();
68         dadesDos.close();
69         fitxerDos.close();
70     }
71 }
```

Console Output:

```
Introdueix el nom del jugador
Edin
Introdueix la nova puntuació
24888
Vols introduir un altre puntuació? s/n
s
Introdueix el nom del jugador
David
Introdueix la nova puntuació
7445
Vols introduir un altre puntuació? s/n
n
Les puntuacions guardades són:
Edin: 24888
Caravaca: 25000
Edin: 10000
Pedro: 25000
Marco: 24500
Nidaa: 75000
Marco: 8888
Nidaa: 74555
Edin: 24888
David: 7445
```



```
1 package pl;
2
3 import java.io.FileOutputStream;
4
5 public class binaris_exercici_03 {
6
7     public static void main(String[] args) throws IOException {
8
9         Scanner scanner = new Scanner(System.in);
10         String nomFitxer = "nomsDeJugadors.bin";
11         String nomFitxerDos = "puntuacionsJugadors.bin";
12         String resposta;
13
14         do {
15             System.out.println("Introdueix el nom del jugador");
16             String nom = scanner.nextLine();
17
18             System.out.println("Introdueix la nova puntuació");
19             int puntuacio = scanner.nextInt();
20             scanner.nextLine();
21
22             FileOutputStream fitxer;
23             DataOutputStream dades;
24             fitxer = new FileOutputStream(nomFitxer, true);
25             dades = new DataOutputStream(fitxer);
26
27             FileOutputStream fitxerDos;
28             DataOutputStream dadesDos;
29             fitxerDos = new FileOutputStream(nomFitxerDos, true);
30             dadesDos = new DataOutputStream(fitxerDos);
31
32             dades.writeUTF(nom);
33             dades.writeInt(puntuacio);
34             dades.close();
35             fitxer.close();
36             dadesDos.close();
37             fitxerDos.close();
38
39             System.out.println("Vols introduir un altre puntuació? s/n");
40             resposta = scanner.nextLine();
41             while (resposta.equalsIgnoreCase("s"));
42
43             lllegirFitxer();
44             scanner.close();
45
46         } while (resposta.equalsIgnoreCase("s"));
47
48     }
49
50     public static void lllegirFitxer () throws IOException {
51
52         String nomFitxer = "nomsDeJugadors.bin";
53         String nomFitxerDos = "puntuacionsJugadors.bin";
54
55         FileInputStream fitxer;
56         DataInputStream dades;
57         fitxer = new FileInputStream(nomFitxer);
58         dades = new DataInputStream(fitxer);
59
60         while (dades.available() > 0) {
61             String nom = dades.readUTF();
62             int puntuacio = dades.readInt();
63             System.out.println(nom + ": " + puntuacio);
64         }
65
66         dades.close();
67         fitxer.close();
68         dadesDos.close();
69         fitxerDos.close();
70     }
71 }
```

Console Output:

```
Introdueix el nom del jugador
Edin
Introdueix la nova puntuació
24888
Vols introduir un altre puntuació? s/n
s
Introdueix el nom del jugador
David
Introdueix la nova puntuació
7445
Vols introduir un altre puntuació? s/n
n
Les puntuacions guardades són:
Edin: 24888
Caravaca: 25000
Edin: 10000
Pedro: 25000
Marco: 24500
Nidaa: 75000
Marco: 8888
Nidaa: 74555
Edin: 24888
David: 7445
```

En el mètode main hi indiquem que hi poden aparèixer errors de lectura o escriptura. Creem un objecte escàner i definim els noms de dos fitxers binaris per guardar els noms en un i les puntuacions a l'altre.

```
public static void main(String[] args) throws IOException{  
Scanner scanner = new Scanner(System.in);  
String nomFitxer = "nomsDeJugadors.bin";  
String nomFitxerDos = "puntuacionsJugadors.bin";
```

Declarem una variable resposta que servirà per saber si l'usuari vol afegir més dades més endavant. I Iniciem un bucle do-while que demanarà a l'usuari que introdueixi el nom del jugador i la seva puntuació.

```
String resposta;  
do {  
System.out.println("Introdueix el nom del jugador");  
String nom = scanner.nextLine();  
System.out.println("Introdueix la nova puntuació");  
int puntuacio = scanner.nextInt();  
scanner.nextLine();
```

Obrirem els fitxer en mode "append" afegint un "true" per poder anar sumant valors als documents sense que es sobre escriuin.

```
FileOutputStream fitxer;  
DataOutputStream dades;  
fitxer = new FileOutputStream(nomFitxer,true);  
dades = new DataOutputStream(fitxer);  
FileOutputStream fitxerDos;  
DataOutputStream dadesDos;  
fitxerDos = new FileOutputStream(nomFitxerDos,true);  
dadesDos = new DataOutputStream(fitxerDos);
```

Escrivim el nom del jugador en format "UTF" i la puntuació en format "int" i tancarem els fluxos.

```
dades.writeUTF(nom);  
dadesDos.writeInt(puntuacio);  
dades.close();  
fitxer.close();  
dadesDos.close();  
fitxerDos.close();
```

Preguntem a l'usuari si vol introduir més dades, si la resposta es si es seguirà repetint el bucle i si és no cridem el mètode per llegir el fitxer i tanquem el escàner.

```
System.out.println("Vols introduir un altre puntuació? s/n");  
resposta = scanner.nextLine();  
}while (resposta.equalsIgnoreCase("s"));  
llegirFitxer();  
scanner.close();  
}
```

En el mètode per llegir els fitxers primer de tot els obrim per la seva lectura.

```
public static void llegirFitxer () throws IOException {  
String nomFitxer = "nomsDeJugadors.bin";  
String nomFitxerDos = "puntuacionsJugadors.bin";  
FileInputStream fitxer;  
DataInputStream dades;  
fitxer = new FileInputStream(nomFitxer);  
dades = new DataInputStream(fitxer);  
FileInputStream fitxerDos;  
DataInputStream dadesDos;  
fitxerDos = new FileInputStream(nomFitxerDos);  
dadesDos = new DataInputStream(fitxerDos);
```

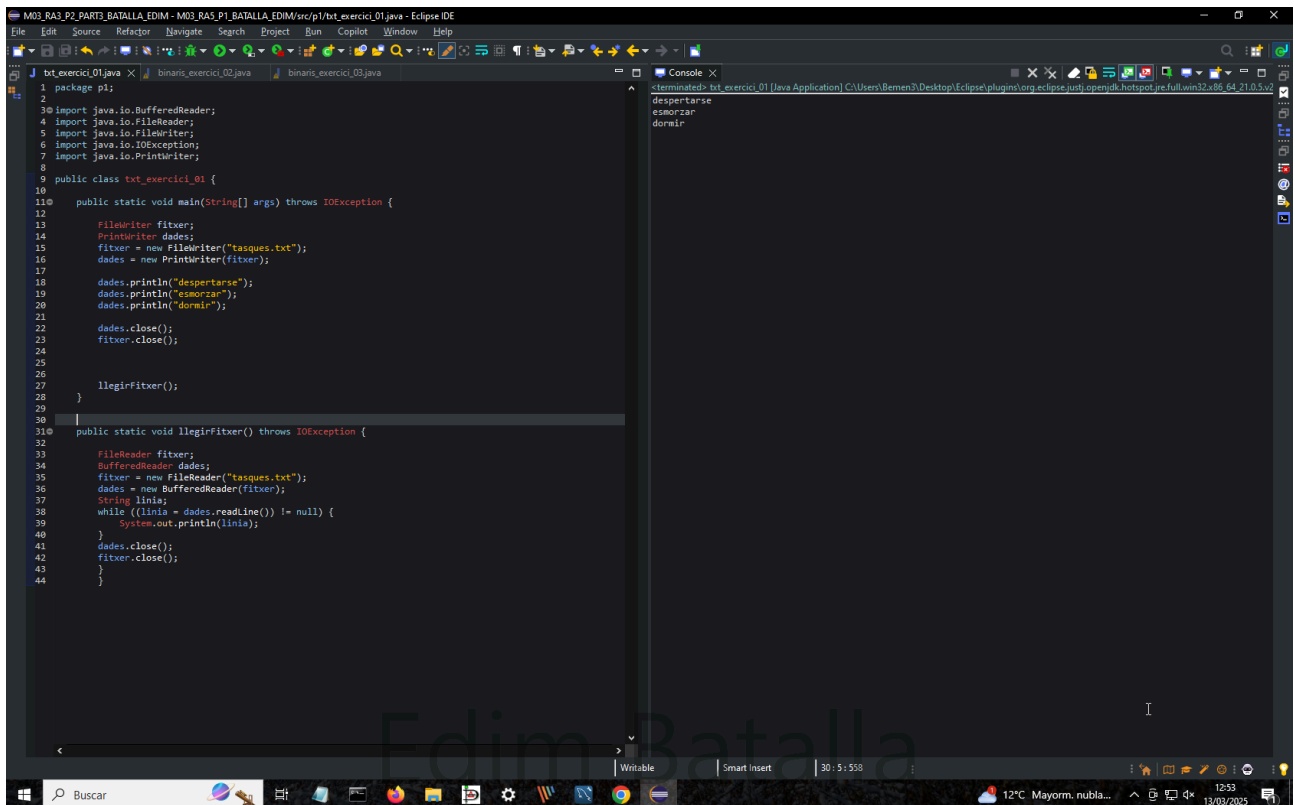
Amb un bucle while mostrarem les dades dels dos arxius mentre hi hagi noms per llegir, un cop no quedin dades s'acabarà el bucle. Tanquem els fluxos de lectura per acabar.

```
System.out.println("Les puntuacions guardades són:");  
while(dades.available() > 0) {  
String nom = dades.readUTF();  
int puntuacio = dadesDos.readInt();  
System.out.println(nom + ": " + puntuacio );  
}  
dades.close();  
fitxer.close();  
dadesDos.close();  
fitxerDos.close();  
}  
}
```

fitxers .txt

Exercici 1

Crea un programa que permeti guardar una llista de tasques en un fitxer `tasques.txt`, on cada línia sigui una tasca. Després, el programa que llegeixi el fitxer i mostri les tasques per pantalla.



The screenshot shows the Eclipse IDE with a Java project named 'M03_RAS_P2_PART3_BATALLA_EDIM'. The main editor displays the file `txt_exercici_01.java` with the following code:

```
1 package pi;
2
3 import java.io.BufferedReader;
4 import java.io.FileReader;
5 import java.io.FileWriter;
6 import java.io.IOException;
7 import java.io.PrintWriter;
8
9 public class txt_exercici_01 {
10
11     public static void main(String[] args) throws IOException {
12
13         FileWriter fitxer;
14         PrintWriter dades;
15         fitxer = new FileWriter("tasques.txt");
16         dades = new PrintWriter(fitxer);
17
18         dades.println("despertarse");
19         dades.println("esmorzar");
20         dades.println("dormir");
21
22         dades.close();
23         fitxer.close();
24
25
26
27         llegirFitxer();
28     }
29
30     public static void llegirFitxer() throws IOException {
31
32         FileReader fitxer;
33         BufferedReader dades;
34         fitxer = new FileReader("tasques.txt");
35         dades = new BufferedReader(fitxer);
36         String linia;
37         while ((linia = dades.readLine()) != null) {
38             System.out.println(linia);
39         }
40         dades.close();
41         fitxer.close();
42     }
43 }
44
```

The console window on the right shows the output of the program:

```
terminated: txt_exercici_01 [Java Application] C:\Users\Beren\Desktop\ eclipse\plugins\org.eclipse.justi.openjdk.hotspot.jre.full.win32.x86_64_21.0.5-2
despertarse
esmorzar
dormir
```

En el mètode `main` creem un arxiu anomenat `"tasques.txt"` i la funció per poder-l'ho escriure.

```
public static void main(String[] args) throws IOException {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("tasques.txt");  
dades = new PrintWriter(fitxer);
```

Escrivim tres línies de text amb tres tasques d'exemple i tanquem els fluxos. I cridem el mètode per llegir el fitxer.

```
dades.println("despertarse");  
dades.println("esmorzar");  
dades.println("dormir");  
dades.close();  
fitxer.close();  
llegirFitxer();  
}
```


En el mètode per llegir el fitxer creem le funcions per poder llegir.

```
public static void llegirFitxer() throws IOException {  
    FileReader fitxer;  
    BufferedReader dades;  
    fitxer = new FileReader("tasques.txt");  
    dades = new BufferedReader(fitxer);
```

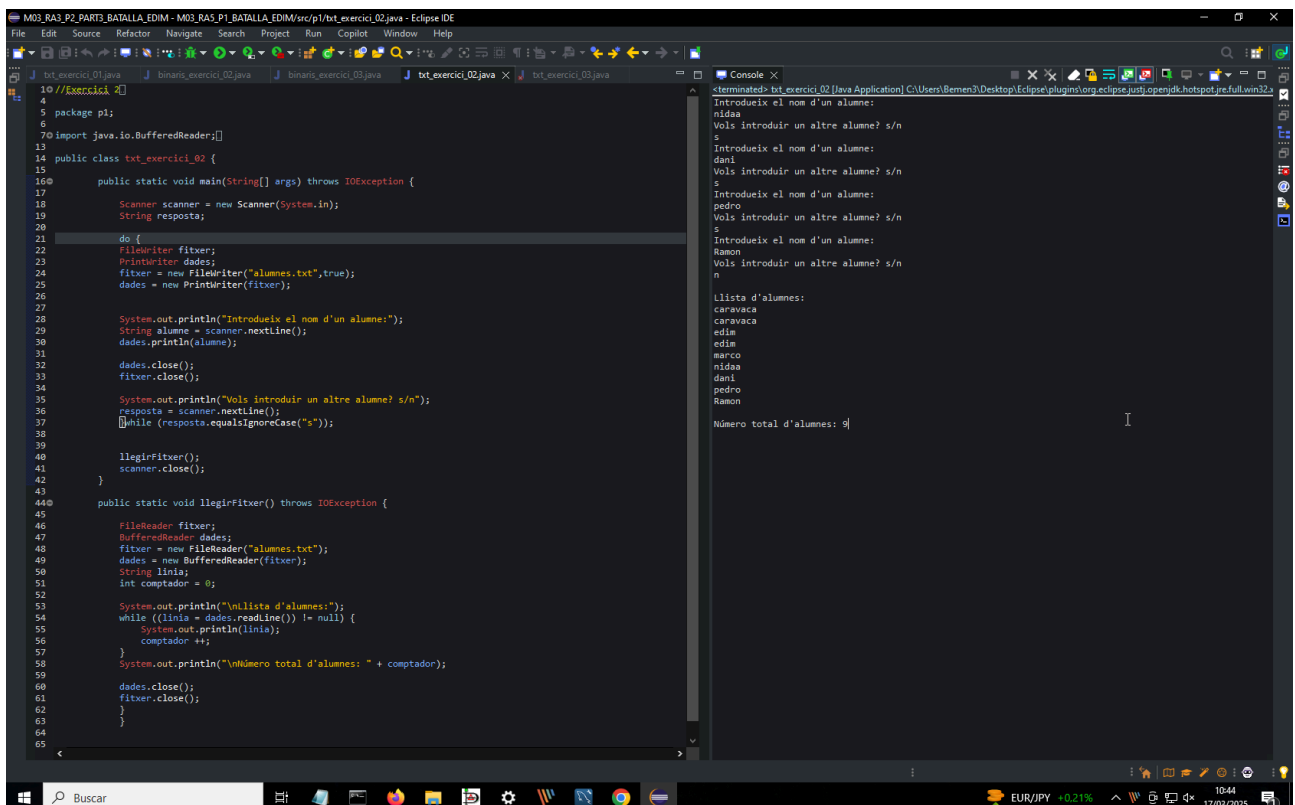
Declarem una variable "linia" que emmagatzemarà cada línia llegida del fitxer. Amb un bucle while fem que es llegeixi cada linea del fitxer amb "readLine()" fins que ens retorni un "null" que ens indicarà que hem arribat al final del fitxer i el bucle s'atura. Tanquem els fluxos de lectura per acabar.

```
String linia;  
while ((linia = dades.readLine()) != null) {  
    System.out.println(linia);  
}  
dades.close();  
fitxer.close();  
}  
}
```

Exercici 2

Escriu un programa que permeti introduir noms d'alumnes i guardar-los en un fitxer alumnes.txt (un per línia).

Després, crea una funció que llegeixi aquest fitxer i indiqui quants alumnes hi ha i mostri els seus noms.



```
10 //Exercici 2  
4  
5 package p1;  
6  
7 import java.io.BufferedReader;  
13  
14 public class txt_exercici_02 {  
15  
16     public static void main(String[] args) throws IOException {  
17  
18         Scanner scanner = new Scanner(System.in);  
19         String resposta;  
20  
21         do {  
22             FileWriter fitxer;  
23             PrintWriter dades;  
24             fitxer = new FileWriter("alumnes.txt", true);  
25             dades = new PrintWriter(fitxer);  
26  
27             System.out.println("Introdueix el nom d'un alumne:");  
28             String alumne = scanner.nextLine();  
29             dades.println(alumne);  
30  
31             dades.close();  
32             fitxer.close();  
33  
34             System.out.println("Vols introduir un altre alumne? s/n");  
35             resposta = scanner.nextLine();  
36             while (resposta.equalsIgnoreCase("s"));  
37  
38             llegirFitxer();  
39             scanner.close();  
40         }  
41     }  
42  
43     public static void llegirFitxer() throws IOException {  
44  
45         FileReader fitxer;  
46         BufferedReader dades;  
47         fitxer = new FileReader("alumnes.txt");  
48         dades = new BufferedReader(fitxer);  
49         String linia;  
50         int comptador = 0;  
51  
52         System.out.println("\nLlista d'alumnes:");  
53         while ((linia = dades.readLine()) != null) {  
54             System.out.println(linia);  
55             comptador++;  
56         }  
57         System.out.println("\nNúmero total d'alumnes: " + comptador);  
58  
59         dades.close();  
60         fitxer.close();  
61     }  
62 }  
63  
64  
65
```

Console Output:

```
<terminated> txt_exercici_02 [Java Application] C:\Users\Bemen3\Desktop\Eclipse\plugins\org.eclipse.justi.openjdk.hotspot.re.full.win32.x86_64.jre\bin\java.exe  
Introdueix el nom d'un alumne:  
nldaa  
Vols introduir un altre alumne? s/n  
s  
Introdueix el nom d'un alumne:  
dani  
Vols introduir un altre alumne? s/n  
s  
Introdueix el nom d'un alumne:  
pedro  
Vols introduir un altre alumne? s/n  
s  
Introdueix el nom d'un alumne:  
Ramon  
Vols introduir un altre alumne? s/n  
n  
  
Llista d'alumnes:  
caravaca  
caravaca  
edim  
edim  
marco  
nldaa  
dani  
pedro  
Ramon  
  
Número total d'alumnes: 9
```

Iniciem el mètode main amb el ja conegut mètode per controlar possibles errors d'entrada o sortida. Creem un escàner i una variable resposta que utilitzarem més endavant per saber si l'usuari vol afegir més dades o no.

```
public static void main(String[] args) throws IOException {  
Scanner scanner = new Scanner(System.in);  
String resposta;
```

Amb un bucle do-while obrim un arxiu anomenat "alumnes.txt" en mode append afegint un "true" després del nom per no sobrescriure les dades introduïdes anteriorment.

```
do {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("alumnes.txt",true);  
dades = new PrintWriter(fitxer);
```

Demanem a l'usuari el nom del alumne i fem que s'escrigui al fitxer amb la funció "println" i tanquem els fitxers.

```
System.out.println("Introdueix el nom d'un alumne:");  
String alumne = scanner.nextLine();  
dades.println(alumne);  
dades.close();  
fitxer.close();
```

Li preguntem al usuari si vol introduir més dades. Si la resposta es afirmativa es repeteix el bucle. Si es negativa s'acaba el bucle, tancem l'objecte escàner i cridem el mètode "llegirFitxer"

```
System.out.println("Vols introduir un altre alumne? s/n");  
resposta = scanner.nextLine();  
}while (resposta.equalsIgnoreCase("s"));  
llegirFitxer();  
scanner.close();  
}
```

En el mètode llegir fitxer obrim el fitxer, declarem una variable línia i iniciem un comptador.

```
public static void llegirFitxer() throws IOException {  
FileReader fitxer;  
BufferedReader dades;  
fitxer = new FileReader("alumnes.txt");  
dades = new BufferedReader(fitxer);  
String linia;  
int comptador = 0;
```

Imprimim les dades del fitxer amb un bucle while mentre les línies que llegim no retornin un null, si el retornen és perquè ja no hi ha més dades per llegir i s'acaba el bucle. Després de cada alumne llegit el comptador creix en una unitat.

```
System.out.println("\nLlista d'alumnes:");  
while ((linia = dades.readLine()) != null) {  
System.out.println(linia);  
comptador ++;  
}
```

Mostrem el número total d'alumnes gràcies a aquest comptador i tanquem el fitxer.

```
System.out.println("\nNúmero total d'alumnes: " + comptador);  
dades.close();  
fitxer.close();  
}  
}
```

Exercici 3

Desenvolupa un programa que permeti afegir productes a un fitxer magatzem.txt.

Cada producte té:

- **Nom (String)**
- **Quantitat (int)**
- **Preu (double)**

Cada producte s'ha de guardar en el fitxer seguint el format Nom,Quantitat,Preu.

El programa ha de permetre llegir el fitxer i mostrar tots els productes registrats.

```
10 public class txt_exercici_03 {  
11  
12     public static void main(String[] args) throws IOException {  
13  
14         Scanner scanner = new Scanner(System.in);  
15         String resposta;  
16  
17         do {  
18             FileWriter fitxer;  
19             PrintWriter dades;  
20             fitxer = new FileWriter("magatzem.txt", true);  
21             dades = new PrintWriter(fitxer);  
22  
23             System.out.println("Introdueix el nom del producte:");  
24             String nom = scanner.nextLine();  
25  
26             System.out.println("Introdueix la quantitat d'aquest producte:");  
27             int quantitat = scanner.nextInt();  
28  
29             System.out.println("Introdueix el preu d'aquest producte:");  
30             double preu = scanner.nextDouble();  
31  
32             scanner.nextLine();  
33             dades.println(nom + "," + quantitat + "," + preu);  
34  
35             dades.close();  
36             fitxer.close();  
37  
38             System.out.println("Vols introduir un altre producte? s/n");  
39             resposta = scanner.nextLine();  
40         } while (resposta.equalsIgnoreCase("s"));  
41  
42         llegirFitxer();  
43         scanner.close();  
44     }  
45  
46  
47     public static void llegirFitxer() throws IOException {  
48  
49         FileReader fitxer;  
50         BufferedReader dades;  
51         fitxer = new FileReader("magatzem.txt");  
52         dades = new BufferedReader(fitxer);  
53  
54         String linia;  
55         while ((linia = dades.readLine()) != null) {  
56             String[] parts = linia.split(",");  
57             String nom = parts[0];  
58             int quantitat = Integer.parseInt(parts[1]);  
59             double preu = Double.parseDouble(parts[2]);  
60             System.out.println("Nom: " + nom + ", Quantitat: " + quantitat + ", PVP: " + preu);  
61         }  
62  
63         dades.close();  
64         fitxer.close();  
65     }  
66 }  
67 }
```

Console Output:

```
<terminated> txt_exercici_03 [Java Application] C:\Users\Bemen3\De  
Introdueix el nom del producte:  
mitjons  
Introdueix la quantitat d'aquest producte:  
300  
Introdueix el preu d'aquest producte:  
45,25  
Vols introduir un altre producte? s/n  
s  
Introdueix el nom del producte:  
camises  
Introdueix la quantitat d'aquest producte:  
9999  
Introdueix el preu d'aquest producte:  
45,25  
Vols introduir un altre producte? s/n  
n  
Nom: bambes, Quantitat: 50, PVP: 99.0  
Nom: mitjons, Quantitat: 300, PVP: 45.25  
Nom: camises, Quantitat: 9999, PVP: 45.25
```

En el mètode main creem l'objecte escàner i una variable "resposta" per indicar si volem seguir executant el bucle més endavant.

```
public static void main(String[] args) throws IOException {  
Scanner scanner = new Scanner(System.in);  
String resposta;
```

Iniciem un bucle do-while on obrim l'arxiu "magatzem.txt" i la utilitat per escriure-hi dades.

```
do {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("magatzem.txt", true);  
dades = new PrintWriter(fitxer);
```

Seguint dintre el bucle demanem al usuari el nom del producte, la quantitat i el preu per guardar-ho en variables.

```
System.out.println("Introdueix el nom del producte:");  
String nom = scanner.nextLine();  
System.out.println("Introdueix la quantitat d'aquest producte:");  
int quantitat = scanner.nextInt();  
System.out.println("Introdueix el preu d'aquest producte:");  
double preu = scanner.nextDouble();  
scanner.nextLine();
```

Escrivim les tres variables guardades dintre de l'arxiu utilitzant la utilitat "println" per seguidament tancar l'arxiu i la funció per poder escriure-hi.

```
dades.println(nom + "," + quantitat + "," + preu);  
dades.close();  
fitxer.close();
```

Preguntem a l'usuari si vol afegir més productes i si la resposta és afirmativa es repeteix el bucle desde el inici i si és negativa es finalitza el bucle, tanquem l'objecte del escàner i cridem el mètode per llegir l'arxiu generat.

```
System.out.println("Vols introduir un altre producte? s/n");  
resposta = scanner.nextLine();  
}while (resposta.equalsIgnoreCase("s"));  
llegirFitxer();  
scanner.close();  
}
```

En el mètode per llegir l'arxiu iniciem les funcionalitats per llegir l'arxiu generat anteriorment.

```
public static void llegirFitxer() throws IOException {  
FileReader fitxer;  
BufferedReader dades;  
fitxer = new FileReader("magatzem.txt");  
dades = new BufferedReader(fitxer);
```

Declarem una variable string anomenada "linia" i iniciem un bucle que llegeixi línia per línia l'arxiu fins que retorni un null que indica que no hi ha més dades per llegir.

```
String linia;  
while ((linia = dades.readLine()) != null) {
```

Dividim cada línia en 3 parts utilitzant "split(",")" on "parts[0]" és el nom del producte, "parts[1]" és la quantitat convertida a integer i "parts[2]" és el preu convertit a double.

```
String[] parts = linia.split(",");  
String nom = parts[0];  
int quantitat = Integer.parseInt(parts[1]);  
double preu = Double.parseDouble(parts[2]);
```

Mostrem la informació del producte amb un format llegible i tanquem el fitxer.

```
System.out.println("Nom: " + nom + ", Quantitat: " + quantitat + ", PVP: " +  
preu);  
}  
dades.close();  
fitxer.close();  
}  
}
```

Exercici 4

Fes un programa que permeti registrar usuaris en un fitxer usuaris.txt.

Cada usuari ha de tenir:

- Nom (String)
- Edat (int)
- Estat d'activitat (boolean)

Cada usuari s'ha de guardar en una línia amb el format Nom,Edat,Actiu.
El programa ha de permetre afegir nous usuaris i consultar els registrats, indicant quants són actius i quants no. Per últim ha d'haver una opció per a eliminar usuaris, però solament aquells que l'estat actiu sigui NO actiu.

```
package pi;

import java.io.BufferedReader;
import java.io.File;
import java.io.FileReader;
import java.io.FileWriter;
import java.io.IOException;
import java.io.PrintWriter;
import java.util.Scanner;
import java.util.ArrayList;

public class txt_exercici_04 {

    public static void main(String[] args) throws IOException {

        Scanner scanner = new Scanner(System.in);
        int opcio;

        do {
            System.out.println("-----Menu Usuaris-----");
            System.out.println("1. Registrar usuari.");
            System.out.println("2. Consultar usuaris.");
            System.out.println("3. Eliminar usuari.");
            System.out.println("4. Sortir.");

            opcio = scanner.nextInt();
            scanner.nextLine();

            switch (opcio) {
                case 1:
                    registrarUsuari(scanner);
                    break;
                case 2:
                    llegirFitxer();
                    break;
                case 3:
                    eliminarUsuari(scanner);
                    break;
                case 4:
                    System.out.println("Sortint del programa...");
                    break;
                default:
                    System.out.println("Opció invàlida. Torna a intentar.");
            }
            System.out.println();
        } while (opcio != 4);

        scanner.close();
    }

    public static void registrarUsuari(Scanner scanner) throws IOException {

        FileWriter fitxer;
        PrintWriter dades;
        fitxer = new FileWriter("usuaris.txt", true);
        dades = new PrintWriter(fitxer);

        String resposta;

        while (true) {
            System.out.println("Introdueix el nom:");
            String nom = scanner.nextLine();

            System.out.println("Introdueix l'edat:");
            int edat = scanner.nextInt();
            scanner.nextLine();

            System.out.println("Introdueix l'estat de l'activitat: (True/False)");
            boolean actiu = scanner.nextBoolean();
            scanner.nextLine();

            System.out.println("Vols introduir un altre usuari? s/n");
            String resposta = scanner.nextLine();

            if (resposta.equalsIgnoreCase("s")) {
                continue;
            } else {
                break;
            }

            dades.println(nom + "," + edat + "," + actiu);
            dades.close();
        }
    }

    public static void llegirFitxer() {
        try {
            File fitxer = new File("usuaris.txt");
            if (!fitxer.exists()) {
                System.out.println("El fitxer no existeix.");
                return;
            }
            FileReader fr = new FileReader(fitxer);
            BufferedReader br = new BufferedReader(fr);

            String linea;
            while ((linea = br.readLine()) != null) {
                System.out.println(linea);
            }
            br.close();
        } catch (IOException e) {
            System.out.println("Error al llegir el fitxer: " + e.getMessage());
        }
    }

    public static void eliminarUsuari(Scanner scanner) {
        System.out.println("Introdueix el nom de l'usuari que vols eliminar:");
        String nom = scanner.nextLine();

        try {
            File fitxer = new File("usuaris.txt");
            if (!fitxer.exists()) {
                System.out.println("El fitxer no existeix.");
                return;
            }
            FileReader fr = new FileReader(fitxer);
            BufferedReader br = new BufferedReader(fr);

            String linea;
            boolean trobat = false;
            while ((linea = br.readLine()) != null) {
                String[] parts = linea.split(",");
                if (parts[0].equalsIgnoreCase(nom)) {
                    System.out.println("No s'ha trobat l'usuari o l'usuari està actiu.");
                    return;
                }
            }
            br.close();

            FileWriter fw = new FileWriter(fitxer);
            PrintWriter pw = new PrintWriter(fw);

            while ((linea = br.readLine()) != null) {
                String[] parts = linea.split(",");
                if (!parts[0].equalsIgnoreCase(nom)) {
                    pw.println(linea);
                }
            }
            pw.close();
            br.close();

            System.out.println("Usuari eliminat correctament.");
        } catch (IOException e) {
            System.out.println("Error al eliminar l'usuari: " + e.getMessage());
        }
    }
}
```

The screenshot displays the Eclipse IDE interface. The left pane shows the source code of `txt_exercici_04.java`, which implements a scanner application. The right pane shows the console output of the application.

Source Code (txt_exercici_04.java):

```
59 public static void registrarUsuari(Scanner scanner) throws IOException {
60     FileWriter fitxer;
61     PrintWriter dades;
62     fitxer = new FileWriter("usuaris.txt", true);
63     dades = new PrintWriter(fitxer);
64     String resposta;
65
66     do {
67
68         System.out.println("Introdueix el nom:");
69         String nom = scanner.nextLine();
70
71         System.out.println("Introdueix l'edat:");
72         int edat = scanner.nextInt();
73
74         System.out.println("Introdueix l'estat de l'activitat: (True/False)");
75         boolean actiu = scanner.nextBoolean();
76
77         scanner.nextLine();
78         dades.println(nom + " " + edat + " " + actiu);
79
80     } while (resposta.equalsIgnoreCase("s"));
81
82     System.out.println("Vols introduir un altre usuari? s/n");
83     resposta = scanner.nextLine();
84     while (resposta.equalsIgnoreCase("s")) {
85         dades.close();
86         fitxer.close();
87     }
88
89     public static void llegirFitxer() throws IOException {
90
91         FileReader fitxer;
92         BufferedReader dades;
93         fitxer = new FileReader("usuaris.txt");
94         dades = new BufferedReader(fitxer);
95
96         String llinia;
97         while ((llinia = dades.readLine()) != null) {
98             String[] parts = llinia.split(" ");
99             String nom = parts[0];
100             int edat = Integer.parseInt(parts[1]);
101             boolean actiu = Boolean.parseBoolean(parts[2]);
102             System.out.println("Nom: " + nom + ", Edat: " + edat + ", Actiu: " + actiu);
103         }
104         dades.close();
105         fitxer.close();
106     }
107
108     public static void eliminarUsuari(Scanner scanner) throws IOException {
109
110         FileReader fitxer;
111         BufferedReader dades;
112         FileWriter fitxerTemp;
```

Console Output:

```
Introdueix l'estat de l'activitat: (True/False)
false
Vols introduir un altre usuari? s/n
n

----Menu Usuaris-----
1. Registrar usuari.
2. Consultar usuaris.
3. Eliminar usuari.
4. Sortir.
2
Nom: Marc, Edat: 30, Actiu: true
Nom: Andreu, Edat: 30, Actiu: true
Nom: Dani, Edat: 25, Actiu: true
Nom: Nidau, Edat: 20, Actiu: false
Nom: Uxoze, Edat: 19, Actiu: false
Nom: Marco, Edat: 29, Actiu: false

----Menu Usuaris-----
1. Registrar usuari.
2. Consultar usuaris.
3. Eliminar usuari.
4. Sortir.
3
Introdueix el nom de l'usuari que vols eliminar:
Marc
No s'ha trobat l'usuari o l'usuari està actiu.

----Menu Usuaris-----
1. Registrar usuari.
2. Consultar usuaris.
3. Eliminar usuari.
4. Sortir.
4
Introdueix el nom de l'usuari que vols eliminar:
Nidau
Usuari eliminat correctament.

----Menu Usuaris-----
1. Registrar usuari.
2. Consultar usuaris.
3. Eliminar usuari.
4. Sortir.
2
Nom: Marc, Edat: 30, Actiu: true
Nom: Andreu, Edat: 30, Actiu: true
Nom: Dani, Edat: 25, Actiu: true
Nom: Uxoze, Edat: 19, Actiu: false
Nom: Marco, Edat: 29, Actiu: false

----Menu Usuaris-----
1. Registrar usuari.
2. Consultar usuaris.
3. Eliminar usuari.
4. Sortir.
```

The screenshot displays the Eclipse IDE interface. The main editor window shows a Java file named `txt_exercici_04.java` with the following code:

```
99 while ((linia = dades.readLine()) != null) {
100     String[] parts = linia.split(",");
101     String nom = parts[0];
102     int edat = Integer.parseInt(parts[1]);
103     boolean actiu = Boolean.parseBoolean(parts[2]);
104     System.out.println("Nom: " + nom + ", Edat: " + edat + ", Actiu: " + actiu);
105 }
106 dades.close();
107 fitxer.close();
108 }
109 }
110
111 public static void eliminarUsuari(Scanner scanner) throws IOException {
112
113     FileReader fitxer;
114     BufferedReader dades;
115     FileWriter fitxerTemp;
116     PrintWriter dadesTemp;
117     fitxer = new FileReader("usuari.txt");
118     dades = new BufferedReader(fitxer);
119
120     fitxerTemp = new FileWriter("usuariTemp.txt");
121     dadesTemp = new PrintWriter(fitxerTemp);
122
123     System.out.println("Introdueix el nom de l'usuari que vols eliminar:");
124     String nom = scanner.nextLine();
125     boolean eliminat = false;
126     String linia;
127
128     while ((linia = dades.readLine()) != null) {
129         String[] parts = linia.split(",");
130         if (parts[0].equalsIgnoreCase(nom) && !Boolean.parseBoolean(parts[2])) {
131             eliminat = true;
132         } else {
133             dadesTemp.println(linia);
134         }
135     }
136
137     dades.close();
138     fitxer.close();
139     dadesTemp.close();
140     fitxerTemp.close();
141
142     if (eliminat) {
143         File fitxerOriginal = new File("usuari.txt");
144         File fitxerNou = new File("usuariTemp.txt");
145         fitxerOriginal.delete();
146         fitxerNou.renameTo(fitxerOriginal);
147         System.out.println("Usuari eliminat correctament.");
148     } else {
149         System.out.println("No s'ha trobat l'usuari o l'usuari està actiu.");
150     }
151 }
152
153 }
154 }
155 }
156 }
```

The console window on the right shows the execution output:

```
Introdueix el nom de l'usuari que vols eliminar:
29
Introdueix l'estat de l'activitat: (True/False)
30
false
Vols introduir un altre usuari? s/n
n

----Menu Usuari-----
1. Registrar usuari.
2. Consultar usuari.
3. Eliminar usuari.
4. Sortir.
2
Nom: Marc, Edat: 30, Actiu: true
Nom: Andreu, Edat: 30, Actiu: true
Nom: Dani, Edat: 25, Actiu: true
Nom: Nidaa, Edat: 20, Actiu: false
Nom: Ibone, Edat: 19, Actiu: false
Nom: Marco, Edat: 29, Actiu: false

----Menu Usuari-----
1. Registrar usuari.
2. Consultar usuari.
3. Eliminar usuari.
4. Sortir.
3
Introdueix el nom de l'usuari que vols eliminar:
Marc
No s'ha trobat l'usuari o l'usuari està actiu.

----Menu Usuari-----
1. Registrar usuari.
2. Consultar usuari.
3. Eliminar usuari.
4. Sortir.
3
Introdueix el nom de l'usuari que vols eliminar:
Nidaa
Usuari eliminat correctament.

----Menu Usuari-----
1. Registrar usuari.
2. Consultar usuari.
3. Eliminar usuari.
4. Sortir.
2
Nom: Marc, Edat: 30, Actiu: true
Nom: Andreu, Edat: 30, Actiu: true
Nom: Dani, Edat: 25, Actiu: true
Nom: Ibone, Edat: 19, Actiu: false
Nom: Marco, Edat: 29, Actiu: false

----Menu Usuari-----
1. Registrar usuari.
2. Consultar usuari.
3. Eliminar usuari.
4. Sortir.
3
```

En el mètode main creem l'objecte escàner i una variable per determinar quina opció escull l'usuari.

```
public static void main(String[] args) throws IOException {  
Scanner scanner = new Scanner(System.in);  
int opcio;
```

Iniciem un bucle do-while on es mostra un menú amb quatre opcions: registrar, consultar i eliminar usuaris a part de sortir.

```
do {  
System.out.println("-----Menu Usuaris-----");  
System.out.println("1. Registrar usuari.");  
System.out.println("2. Consulta usuaris.");  
System.out.println("3. Eliminar usuari.");  
System.out.println("4. Sortir.");  
opcio = scanner.nextInt();  
scanner.nextLine();
```

Generem un switch case on cada cas crida el mètode corresponent a la funció del menú juntament amb el cas de sortida i el default per si s'introdueix una opció no vàlida. El bucle es repetirà indefinidament fins que l'usuari no esculli l'opció de sortir prement la tecla 4.

```
switch (opcio) {  
case 1:  
registrarUsuari(scanner);  
break;  
case 2:  
llegirFitxer();  
break;  
case 3:  
eliminarUsuari(scanner);  
break;  
case 4:  
System.out.println("Sortint del programa...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a intentar.");  
}  
System.out.println();  
}while (opcio !=4);  
scanner.close();  
}
```

La primera opció del menu crida el mètode registrar usuari on obrim l'arxiu "usuaris.txt" seguit de true perquè no es subscriguin les dades registrades anteriorment i la funció per poder escriure-hi.

```
public static void registrarUsuari(Scanner scanner) throws IOException {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("usuaris.txt", true);  
dades = new PrintWriter(fitxer);  
String resposta;
```


En un bucle do-while demanem al usuari que introdueix-hi el nom i l'edat de l'usuari i a continuació preguntem si l'estat de l'activitat es true o false. Aquestes dades es guardaran en variable tipus string, integer i booleà respectivament.

```
do {  
System.out.println("Introdueix el nom:");  
String nom = scanner.nextLine();  
System.out.println("Introdueix l'edat:");  
int edat = scanner.nextInt();  
System.out.println("Introdueix l'estat de l'activitat: (True/False)");  
boolean actiu = scanner.nextBoolean();  
scanner.nextLine();
```

Escriurem les tres variables en una línia del document "usuaris.txt" utilitzant la funció "dades.println" i l'hi preguntem al usuari si vol introduir més usuaris. Depenent de la seva resposta es repetirà el bucle o tancarem l'arxiu.

```
dades.println(nom + "," + edat + "," + actiu);  
System.out.println("Vols introduir un altre usuari? s/n");  
resposta = scanner.nextLine();  
} while (resposta.equalsIgnoreCase("s"));  
dades.close();  
fitxer.close();  
}
```

La segona opció del menú crida el mètode per llegir el fitxer, obrim l'arxiu i la funcionalitat per poder-lo llegir.

```
public static void llegirFitxer() throws IOException {  
FileReader fitxer;  
BufferedReader dades;  
fitxer = new FileReader("usuaris.txt");  
dades = new BufferedReader(fitxer);  
String linia;
```

Amb un bucle while llegim línia per línia el document fins que retorn-hi un null que ens indica que ja no hi han més línies per llegir.

```
while ((linia = dades.readLine()) != null) {
```

Llegim cada línia dividint les parts amb "split(",")" la part 0 és igual al nom, la part 1 és igual a l'edat i la part dos és igual al booleà que ens indica si l'usuari està actiu o no. Totes aquestes dades les convertim de text al format corresponent amb la funcionalitat "parse".

```
String[] parts = linia.split(",");  
String nom = parts[0];  
int edat = Integer.parseInt(parts[1]);  
boolean actiu = Boolean.parseBoolean(parts[2]);
```

Mostrem el contingut de les variables en una línia per la consola i tanquem el fitxer.

```
System.out.println("Nom: " + nom + ", Edat: " + edat + ", Actiu: " + actiu);  
}  
dades.close();  
fitxer.close();  
}
```

La opció tres crida el mètode per eliminar un usuari. Obrim l'arxiu "usuaris.txt" i generem un arxiu temporal "usuarisTemp" amb les funcionalitats de llegir i escriure-hi.

```
public static void eliminarUsuari(Scanner scanner) throws IOException {  
    FileReader fitxer;  
    BufferedReader dades;  
    FileWriter fitxerTemp;  
    PrintWriter dadesTemp;  
    fitxer = new FileReader("usuaris.txt");  
    dades = new BufferedReader(fitxer);  
    fitxerTemp = new FileWriter("usuarisTemp.txt");  
    dadesTemp = new PrintWriter(fitxerTemp);
```

Li demanem al usuari del programa el nom de l'usuari que vol eliminar i el guardem en una variable, declarem una variable booleà per determinar si un usuari ha estat eliminat o no, d'entrada la declarem "=false" ja que d'entrada cap usuari està eliminat.

```
System.out.println("Introdueix el nom de l'usuari que vols eliminar:");  
String nom = scanner.nextLine();  
boolean eliminat = false;  
String linia;
```

Amb un bucle while llegim tots les lineas mentres tinguin contingut i separem el contingut de la linea amb la funcionalitat "linia.split(",")".

```
while ((linia = dades.readLine()) != null) {  
    String[] parts = linia.split(",");
```

Comparem amb un condicional "if" si el nom introduït per l'usuari coincideix amb algun dels que estan registrats en la part[0] de la linea i comprovem si l'usuari està marcat com a no actiu comprovant el contingut de la "part[2]"

```
if (parts[0].equalsIgnoreCase(nom) && !Boolean.parseBoolean(parts[2])) {
```

Si hi ha coincidència en les dues condicions marquem l'usuari com com eliminat utilitzant el booleà i no l'imprimim en el nou arxiu temporal. Tot usuari que no compleixi-hi aquestes condicions si que l'escriurem en el nou arxiu temporal.

```
    eliminat = true;  
    } else {  
        dadesTemp.println(linia);  
    }  
}
```

Tanquem tots els documents.

```
dades.close();  
fitxer.close();  
dadesTemp.close();  
fitxerTemp.close();
```

Si hi ha hagut algun usuari eliminat, eliminarem l'arxiu original i el substituïrem amb el contingut de l'arxiu temporal on no només hi han els usuaris no eliminats. Si no hi hagut cap usuari eliminat deixarem l'arxiu original sense canvis.

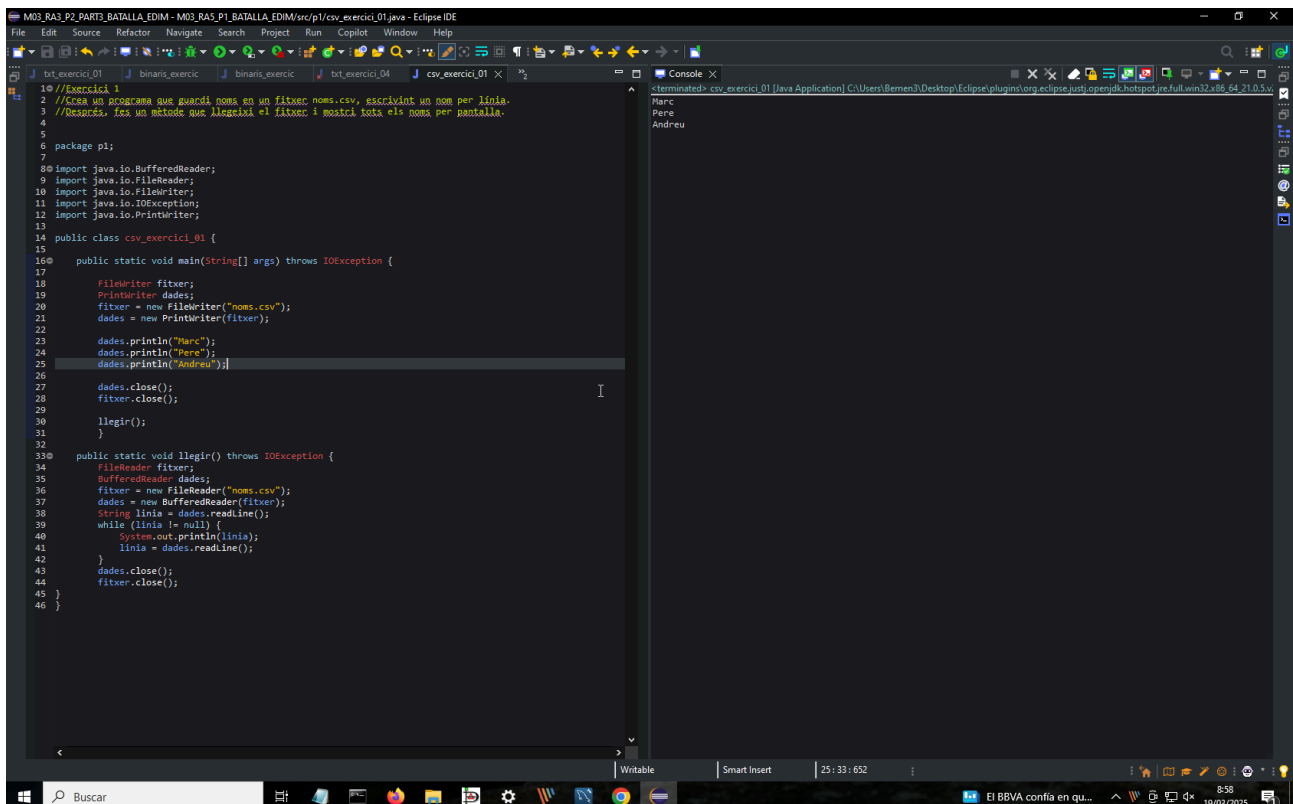
```
if (eliminat) {  
    File fitxerOriginal = new File("usuaris.txt");  
    File fitxerNou = new File("usuarisTemp.txt");  
    fitxerOriginal.delete();  
    fitxerNou.renameTo(fitxerOriginal);  
    System.out.println("Usuari eliminat correctament.");  
} else {  
    System.out.println("No s'ha trobat l'usuari o l'usuari està actiu.");  
}  
}  
}  
}
```

fitxers .csv

Exercici 1

Crea un programa que guardi noms en un fitxer noms.csv, escrivint un nom per línia.

Després, fes un mètode que llegeixi el fitxer i mostri tots els noms per pantalla.



```
1 //Exercici 1  
2 //Crea un programa que guardi noms en un fitxer noms.csv, escrivint un nom per línia.  
3 //Després, fes un mètode que llegeixi el fitxer i mostri tots els noms per pantalla.  
4  
5  
6 package pl;  
7  
8 import java.io.BufferedReader;  
9 import java.io.FileReader;  
10 import java.io.FileWriter;  
11 import java.io.IOException;  
12 import java.io.PrintWriter;  
13  
14 public class csv_exercici_01 {  
15  
16     public static void main(String[] args) throws IOException {  
17         FileWriter fitxer;  
18         PrintWriter dades;  
19         fitxer = new FileWriter("noms.csv");  
20         dades = new PrintWriter(fitxer);  
21  
22         dades.println("Marc");  
23         dades.println("Pere");  
24         dades.println("Andreu");  
25  
26         dades.close();  
27         fitxer.close();  
28  
29         llegir();  
30     }  
31  
32  
33     public static void llegir() throws IOException {  
34         FileReader fitxer;  
35         BufferedReader dades;  
36         fitxer = new FileReader("noms.csv");  
37         dades = new BufferedReader(fitxer);  
38         String linia = dades.readLine();  
39         while (linia != null) {  
40             System.out.println(linia);  
41             linia = dades.readLine();  
42         }  
43         dades.close();  
44         fitxer.close();  
45     }  
46 }
```

The screenshot shows the Eclipse IDE with the file 'csv_exercici_01.java' open. The code implements a program that writes names to a 'noms.csv' file and then reads them back to the console. The console output shows the names 'Marc', 'Pere', and 'Andreu'.

En el mètode main obrim un arxiu anomenat "noms.csv" i la funcionalitat per poder-l'ho escriure.

```
public static void main(String[] args) throws IOException {  
    FileWriter fitxer;  
    PrintWriter dades;  
    fitxer = new FileWriter("noms.csv");  
    dades = new PrintWriter(fitxer);
```

Escrivim tres línies dintre del arxiu creat amb tres noms d'exemple gràcies a la funcionalitat "dades.println("")". Cridem el mètode per llegir l'arxiu i tanquem el fitxer.

```
    dades.println("Marc");  
    dades.println("Pere");  
    dades.println("Andreu");  
    dades.close();  
    fitxer.close();  
    llegir();  
}
```

En el mètode per llegir el fitxer, obrim el fitxer i la funcionalitat per llegir-l'ho.

```
public static void llegir() throws IOException {  
    FileReader fitxer;  
    BufferedReader dades;  
    fitxer = new FileReader("noms.csv");  
    dades = new BufferedReader(fitxer);
```

Amb un bucle while llegim cada línia mentres no hi hagi el retorn de un null ja que voldrà dir que no hi han més línies per llegir. Mostrem cada línia llegida per la consola i tanquem el fitxer.

```
    String linia = dades.readLine();  
    while (linia != null) {  
        System.out.println(linia);  
        linia = dades.readLine();  
    }  
    dades.close();  
    fitxer.close();  
}
```

Exercici 2

Desenvolupa un programa que permeti afegir llibres a un fitxer llibres.csv, on cada

llibre té:

- Títol (String)
- Autor (String)
- Any de publicació (int)

Cada llibre s'ha de guardar seguint el format Títol,Autor,Any.

El programa ha de permetre llegir el fitxer i mostrar tots els llibres registrats.

```
11 package p1;
12
13 import java.io.BufferedReader;
14 import java.io.FileReader;
15 import java.io.IOException;
16 import java.io.PrintWriter;
17 import java.util.Scanner;
18
19 public class csv_exercici_02 {
20
21     public static void main(String[] args) throws IOException {
22
23         Scanner scanner = new Scanner(System.in);
24         int opcio;
25
26         do {
27             System.out.println("-----Menu Usuari-----");
28             System.out.println("1. Registrar llibre.");
29             System.out.println("2. Consultar llibres.");
30             System.out.println("3. Sortir.");
31
32             opcio = scanner.nextInt();
33             scanner.nextLine();
34
35             switch (opcio) {
36                 case 1:
37                     registrarllibre(scanner);
38                     break;
39                 case 2:
40                     consultarllibres();
41                     break;
42                 case 3:
43                     System.out.println("Sortint del programa...");
44                     break;
45                 default:
46                     System.out.println("Opció invàlida. Torna a intentar.");
47             }
48             System.out.println();
49         } while (opcio != 3);
50         scanner.close();
51     }
52
53     public static void registrarllibre (Scanner scanner) throws IOException {
54
55         PrintWriter fitxer;
56         Printwriter dades;
57         fitxer = new PrintWriter("llibres.csv", true);
58         dades = new Printwriter(fitxer);
59         String resposta;
60
61         do {
62             System.out.println("Introdueix el títol:");
63             String titol = scanner.nextLine();
64
65             System.out.println("Introdueix l'autor:");
66             String autor = scanner.nextLine();
67
68             System.out.println("Introdueix l'any de publicació:");
69             int anyPublicacio = scanner.nextInt();
70             scanner.nextLine();
71
72             dades.println(titol + "," + autor + "," + anyPublicacio);
73
74             System.out.println("Vols introduir un altre llibre? s/n");
75             resposta = scanner.nextLine();
76             while (resposta.equalsIgnoreCase("s"));
77         } while (resposta.equalsIgnoreCase("s"));
78         dades.close();
79         fitxer.close();
80     }
81
82     public static void consultarllibres () throws IOException {
83
84         BufferedReader fitxer;
85         BufferedReader dades;
86         fitxer = new BufferedReader("llibres.csv");
87         dades = new BufferedReader(fitxer);
88         String linia;
89
90         while ((linia = dades.readLine()) != null) {
91             String[] parts = linia.split(",");
92             String titol = parts[0].trim();
93             String autor = parts[1].trim();
94             String any = Integer.parseInt(parts[2].trim());
95             System.out.println("Títol: " + titol + ", Autor: " + autor + ", Any publicació: " + any);
96         }
97         dades.close();
98         fitxer.close();
99     }
100 }
```

Console Output:

```
-----Menu Usuari-----
1. Registrar llibre.
2. Consultar llibres.
3. Sortir.
Introdueix el títol:
Romania
Introdueix l'autor:
Cosmin Loto
Introdueix l'any de publicació:
1777
Vols introduir un altre llibre? s/n
s
-----Menu Usuari-----
1. Registrar llibre.
2. Consultar llibres.
3. Sortir.
2
Títol: El señor de los Anillos, Autor: Tolkien, Any publicació: 1950
Títol: Caravaggio 2, Autor: Marc, Any publicació: 1950
Títol: Marco Polo, Autor: Da Paolo, Any publicació: 1888
Títol: Romania, Autor: Cosmin Loto, Any publicació: 1777
-----Menu Usuari-----
1. Registrar llibre.
2. Consultar llibres.
3. Sortir.
3
Sortir.
```

```
53 scanner.close();
54 }
55
56 public static void registrarllibre (Scanner scanner) throws IOException {
57
58     PrintWriter fitxer;
59     Printwriter dades;
60     fitxer = new PrintWriter("llibres.csv", true);
61     dades = new Printwriter(fitxer);
62     String resposta;
63
64     do {
65         System.out.println("Introdueix el títol:");
66         String titol = scanner.nextLine();
67
68         System.out.println("Introdueix l'autor:");
69         String autor = scanner.nextLine();
70
71         System.out.println("Introdueix l'any de publicació:");
72         int anyPublicacio = scanner.nextInt();
73         scanner.nextLine();
74
75         dades.println(titol + "," + autor + "," + anyPublicacio);
76
77         System.out.println("Vols introduir un altre llibre? s/n");
78         resposta = scanner.nextLine();
79         while (resposta.equalsIgnoreCase("s"));
80     } while (resposta.equalsIgnoreCase("s"));
81     dades.close();
82     fitxer.close();
83 }
84
85 public static void consultarllibres () throws IOException {
86
87     BufferedReader fitxer;
88     BufferedReader dades;
89     fitxer = new BufferedReader("llibres.csv");
90     dades = new BufferedReader(fitxer);
91     String linia;
92
93     while ((linia = dades.readLine()) != null) {
94         String[] parts = linia.split(",");
95         String titol = parts[0].trim();
96         String autor = parts[1].trim();
97         String any = Integer.parseInt(parts[2].trim());
98         System.out.println("Títol: " + titol + ", Autor: " + autor + ", Any publicació: " + any);
99     }
100     dades.close();
101     fitxer.close();
102 }
103 }
```

Console Output:

```
-----Menu Usuari-----
1. Registrar llibre.
2. Consultar llibres.
3. Sortir.
2
Títol: El señor de los Anillos, Autor: Tolkien, Any publicació: 1950
Títol: Caravaggio 2, Autor: Marc, Any publicació: 1950
Títol: Marco Polo, Autor: Da Paolo, Any publicació: 1888
Títol: Romania, Autor: Cosmin Loto, Any publicació: 1777
-----Menu Usuari-----
1. Registrar llibre.
2. Consultar llibres.
3. Sortir.
3
Sortir.
```

En el mètode main creem un objecte escaner i un menú d'opcions que es repetirà indefinidament amb un bucle do-while fins que l'usuari premi l'opció de sortir.

```
public static void main(String[] args) throws IOException {  
Scanner scanner = new Scanner(System.in);  
int opcio;  
do {  
System.out.println("-----Menu Usuaris-----");  
System.out.println("1. Registrar llibre.");  
System.out.println("2. Consultar llibres.");  
System.out.println("3. Sortir.");  
opcio = scanner.nextInt();  
scanner.nextLine();  
switch (opcio) {  
case 1:  
registrarLlibre(scanner);  
break;  
case 2:  
consultarLlibres();  
break;  
case 3:  
System.out.println("Sortint del programa...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a intentar.");  
}  
System.out.println();  
}while (opcio !=3);  
scanner.close();  
}
```

La primera opció del menú crida el mètode per registrar un llibre. Obrim el fitxer "llibres.csv" i l'hi preguntem al usuari el títol del llibre, l'autor i l'any de publicació. Guardem aquestes 3 dades en les seves variables corresponents: String, String i integer.

```
public static void registrarLlibre (Scanner scanner) throws IOException {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("llibres.csv", true);  
dades = new PrintWriter(fitxer);  
String resposta;  
do {  
System.out.println("Introdueix el títol:");  
String titol = scanner.nextLine();  
System.out.println("Introdueix l'autor:");  
String autor = scanner.nextLine();  
System.out.println("Introdueix l'any de publicació");  
int anyPublicacio = scanner.nextInt();  
scanner.nextLine();
```

Escrivim el contingut de les 3 variables en una línia del fitxer i l'hi preguntem al usuari si vol introduir més llibres amb un bucle do-while, depenent de la resposta el bucle es repetirà o es tancarà l'arxiu i retornarem al menú principal.

```
dades.println(titol + "," + autor + "," + anyPublicacio);  
System.out.println("Vols introduir un altre llibre? s/n");  
resposta = scanner.nextLine();  
} while (resposta.equalsIgnoreCase("s"));  
dades.close();  
fitxer.close();  
}
```

La segona opció del menú crida el mètode per consultar els llibres registrats.

```
public static void consultarLlibres () throws IOException {  
FileReader fitxer;  
BufferedReader dades;  
fitxer = new FileReader("llibres.csv");  
dades = new BufferedReader(fitxer);
```

Fem que es llegeixi-hi línia per línia el fitxer mentre hi hagin dades i separem el contingut en tres parts que es delimiten per les comes. amb la funcionalitat "linia.split(",")".

```
String linia;  
dades.readLine();  
while ((linia = dades.readLine()) != null) {  
String[] parts = linia.split(",");
```

Definim cada part com una variable i la del any la passem a Integer amb a funcionalitat "parse". I ho mostrem per la consola. Tanquem l'arxiu per finalitzar.

```
String titol = parts[0].trim();  
String autor = parts[1].trim();  
int any = Integer.parseInt(parts[2].trim());  
System.out.println("Títol: " + titol + ", Autor: " + autor + ", Any publicació: "  
+ any);  
}  
dades.close();  
fitxer.close();  
}  
}
```

Exercici 3

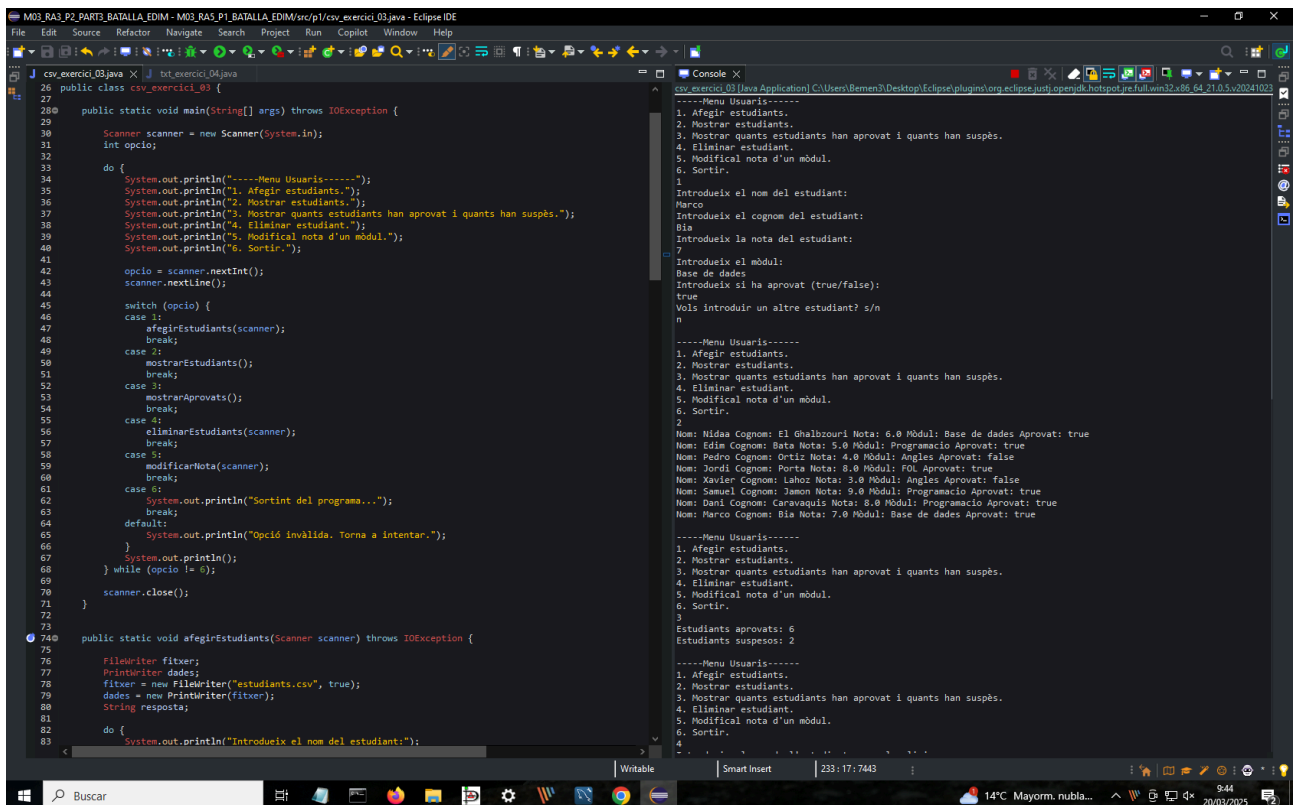
Crea un sistema que permeti registrar estudiants amb les seves notes en un fitxer estudiants.csv.

Cada estudiant ha de tenir:

- **Nom (String)**
- **Cognom (String)**
- **Nota (double)**
- **Mòdul (String)**
- **Aprovat (boolean)**

El programa ha de permetre:

- 1. Afegir nous estudiants.**
- 2. Mostrar la llista d'estudiants.**
- 3. Mostrar quants han aprovat i quants han suspès.**
- 4. Eliminar un alumne i totes les seves notes**
- 5. Modificar una nota d'un mòdul.**




```
1  M03_RAS_P2_PART2_BATALLA_EDIM - M03_RAS_P1_BATALLA_EDIM/src/p/cv_exercici_03.java - Eclipse IDE
2  File Edit Source Refactor Navigate Search Project Run Copilot Window Help
3
4  [Icons]
5
6  J 'cvs_exercici_03.java' X 'btl_exercici_04.java'
7
8  72 public static void afegirEstudiants(Scanner scanner) throws IOException {
9  73
10 74     FileWriter fitxer;
11 75     PrintWriter dades;
12 76     fitxer = new FileWriter("estudiants.csv", true);
13 77     dades = new PrintWriter(fitxer);
14 78     String resposta;
15 79
16 80     do {
17 81
18 82         System.out.println("Introdueix el nom del estudiant:");
19 83         String nom = scanner.nextLine();
20 84         System.out.println("Introdueix el cognom del estudiant:");
21 85         String cognom = scanner.nextLine();
22 86         System.out.println("Introdueix la nota del estudiant:");
23 87         double nota = scanner.nextDouble();
24 88         scanner.nextLine();
25 89         System.out.println("Introdueix el mòdul:");
26 90         String modul = scanner.nextLine();
27 91         System.out.println("Introdueix si ha aprovat (true/false):");
28 92         boolean aprovat = scanner.nextBoolean();
29 93         scanner.nextLine();
30 94         dades.println(nom + " " + cognom + " " + nota + " " + modul + " " + aprovat);
31 95
32 96         System.out.println("Vols introduir un altre estudiant? s/n");
33 97         resposta = scanner.nextLine();
34 98     } while (resposta.equalsIgnoreCase("s"));
35 99
36 100     dades.close();
37 101     fitxer.close();
38 102 }
39 103
40 104 public static void mostrarEstudiants() throws IOException {
41 105
42 106     FileReader fitxer;
43 107     BufferedReader dades;
44 108     fitxer = new FileReader("estudiants.csv");
45 109     dades = new BufferedReader(fitxer);
46 110     String linia;
47 111     while ((linia = dades.readLine()) != null) {
48 112         String[] camps = linia.split(",");
49 113         System.out.println("Nom: " + camps[0] + " Cognom: " + camps[1] + " Nota: " + camps[2] + " Mòdul: "
50 114             + camps[3] + " Aprovat: " + camps[4]);
51 115     }
52 116     dades.close();
53 117     fitxer.close();
54 118 }
55 119
56 120 public static void mostrarAprovats() throws IOException {
57 121
58 122     FileReader fitxer;
59 123     BufferedReader dades;
60 124     fitxer = new FileReader("estudiants.csv");
61 125     dades = new BufferedReader(fitxer);
62 126     String linia;
63 127     int aprovats = 0;
64 128     int suspesos = 0;
65 129     while ((linia = dades.readLine()) != null) {
```

```
-----Menu Usuaris-----
1. Afegir estudiants.
2. Mostrar estudiants.
3. Mostrar quants estudiants han aprovat i quants han suspès.
4. Eliminar estudiant.
5. Modificar nota d'un mòdul.
6. Sortir.
7
Introdueix el nom de l'estudiant que vols eliminar:
Jordi
Introdueix el cognom de l'estudiant que vols eliminar:
Porta
Estudiant eliminat correctament.
-----Menu Usuaris-----
1. Afegir estudiants.
2. Mostrar estudiants.
3. Mostrar quants estudiants han aprovat i quants han suspès.
4. Eliminar estudiant.
5. Modificar nota d'un mòdul.
6. Sortir.
7
Nom: Midaa Cognom: El Ghalbzouri Nota: 6.0 Mòdul: Base de dades Aprovat: true
Nom: Edin Cognom: Bata Nota: 5.0 Mòdul: Programació Aprovat: true
Nom: Pedro Cognom: Ortiz Nota: 4.0 Mòdul: Angles Aprovat: false
Nom: Xavier Cognom: Lahoz Nota: 3.0 Mòdul: Angles Aprovat: false
Nom: Samuel Cognom: Jamon Nota: 9.0 Mòdul: Programació Aprovat: true
Nom: Dani Cognom: Caravaquis Nota: 8.0 Mòdul: Programació Aprovat: true
Nom: Marco Cognom: Bta Nota: 7.0 Mòdul: Base de dades Aprovat: true
-----Menu Usuaris-----
1. Afegir estudiants.
2. Mostrar estudiants.
3. Mostrar quants estudiants han aprovat i quants han suspès.
4. Eliminar estudiant.
5. Modificar nota d'un mòdul.
6. Sortir.
7
Introdueix el nom del estudiant del qual vols modificar una nota:
Samuel
Introdueix el cognom de l'estudiant del qual vols modificar una nota:
Jamon
Introdueix el mòdul del qual vols modificar la nota:
Programació
Introdueix el nou valor de la nota:
4
Introdueix si ha aprovat (true/false):
false
La nota s'ha modificat correctament.
-----Menu Usuaris-----
1. Afegir estudiants.
2. Mostrar estudiants.
3. Mostrar quants estudiants han aprovat i quants han suspès.
4. Eliminar estudiant.
5. Modificar nota d'un mòdul.
```

```

72 public static void afegirEstudiants(Scanner scanner) throws IOException {
73     // ...
74     FileWriter fitxer;
75     PrintWriter dades;
76     fitxer = new FileWriter("estudiants.csv", true);
77     dades = new PrintWriter(fitxer);
78     String resposta;
79     do {
80         System.out.println("Introdueix el nom del estudiant:");
81         String nom = scanner.nextLine();
82         System.out.println("Introdueix el cognom del estudiant:");
83         String cognom = scanner.nextLine();
84         System.out.println("Introdueix la nota del estudiant:");
85         double nota = scanner.nextDouble();
86         scanner.nextLine();
87         System.out.println("Introdueix el mòdul:");
88         String modul = scanner.nextLine();
89         System.out.println("Introdueix si ha aprovat (true/false):");
90         boolean aprovat = scanner.nextBoolean();
91         scanner.nextLine();
92         dades.println(nom + "," + cognom + "," + nota + "," + modul + "," + aprovat);
93     } while (resposta.equalsIgnoreCase("s"));
94     System.out.println("Vols introduir un altre estudiant? s/n");
95     resposta = scanner.nextLine();
96     while (resposta.equalsIgnoreCase("s")) {
97         // ...
98     }
99     dades.close();
100     fitxer.close();
101 }
102
103 public static void mostrarEstudiants() throws IOException {
104     // ...
105     FileReader fitxer;
106     BufferedReader dades;
107     fitxer = new FileReader("estudiants.csv");
108     dades = new BufferedReader(fitxer);
109     String linea;
110     while ((linea = dades.readLine()) != null) {
111         String[] camps = linea.split(",");
112         System.out.println("Nom: " + camps[0] + " Cognom: " + camps[1] + " Nota: " + camps[2] + " Mòdul: " + camps[3] + " Aprovat: " + camps[4]);
113     }
114     dades.close();
115     fitxer.close();
116 }
117
118 public static void mostrarAprovats() throws IOException {
119     // ...
120     FileReader fitxer;
121     BufferedReader dades;
122     fitxer = new FileReader("estudiants.csv");
123     dades = new BufferedReader(fitxer);
124     String linea;
125     int aprovats = 0;
126     int suspesos = 0;
127     while ((linea = dades.readLine()) != null) {
128         // ...
129     }

```


En el mètode main creem l'objecte escàner i generem un menú amb un bucle do-while que es repetirà fins que l'usuari premi la opció de sortir.

```
public static void main(String[] args) throws IOException {  
Scanner scanner = new Scanner(System.in);  
int opcio;  
do {  
System.out.println("-----Menu Usuaris-----");  
System.out.println("1. Afegir estudiants.");  
System.out.println("2. Mostrar estudiants.");  
System.out.println("3. Mostrar quants estudiants han aprovat i quants han suspès.");  
System.out.println("4. Eliminar estudiant.");  
System.out.println("5. Modifical nota d'un mòdul.");  
System.out.println("6. Sortir.");  
opcio = scanner.nextInt();  
scanner.nextLine();  
switch (opcio) {  
case 1:  
afegirEstudiants(scanner);  
break;  
case 2:  
mostrarEstudiants();  
break;  
case 3:  
mostrarAprovats();  
break;  
case 4:  
eliminarEstudiants(scanner);  
break;  
case 5:  
modificarNota(scanner);  
break;  
case 6:  
System.out.println("Sortint del programa...");  
break;  
default:  
System.out.println("Opció invàlida. Torna a intentar.");  
}  
System.out.println();  
} while (opcio != 6);  
scanner.close();  
}
```

La primera opció crida el mètode per afegir estudiants. Obrim l'arxiu "estudiants.csv" i la funcionalitat per poder-hi escriure.

```
public static void afegirEstudiants(Scanner scanner) throws IOException {  
FileWriter fitxer;  
PrintWriter dades;  
fitxer = new FileWriter("estudiants.csv", true);  
dades = new PrintWriter(fitxer);
```

Dintre un bucle do-while demanem al usuari que introdueix-hi el nom del estudiant, el cognom, la nota que ha tret, en quina matèria i si ha aprovat o suspès. Totes aquestes dades es guarden en les seves variables corresponents, tipus String les tres primeres, tipus double la nota i booleà la de si ha suspès.

```
String resposta;  
do {  
System.out.println("Introdueix el nom del estudiant:");  
String nom = scanner.nextLine();  
System.out.println("Introdueix el cognom del estudiant:");  
String cognom = scanner.nextLine();  
System.out.println("Introdueix la nota del estudiant:");  
double nota = scanner.nextDouble();  
scanner.nextLine();  
System.out.println("Introdueix el mòdul:");  
String modul = scanner.nextLine();  
System.out.println("Introdueix si ha aprovat (true/false):");  
boolean aprovat = scanner.nextBoolean();  
scanner.nextLine();  
dades.println(nom + "," + cognom + "," + nota + "," + modul + "," +  
aprovat);
```

Un cop guardades les tres variables l'hi preguntem al usuari si vol introduir un altre estudiant per repetir el bucle o si tanquem i guardem el fitxer per tornar al menú principal.

```
System.out.println("Vols introduir un altre estudiant? s/n");  
resposta = scanner.nextLine();  
} while (resposta.equalsIgnoreCase("s"));  
dades.close();  
fitxer.close();  
}
```

La segona opció crida el mètode per mostrar els estudiants registrats en l'arxiu. Obrim l'arxiu i la funcionalitat per poder-l'ho llegir.

```
public static void mostrarEstudiants() throws IOException {  
FileReader fitxer;  
BufferedReader dades;  
fitxer = new FileReader("estudiants.csv");  
dades = new BufferedReader(fitxer);
```

Llegim cada línia mentres hi hagi contingut, separem el contingut de la línia amb la funcionalitat split i l'ús de comes i mostrem els valors per la consola. Tanquem l'arxiu.

```
String linia;  
while ((linia = dades.readLine()) != null) {  
String[] camps = linia.split(",");  
System.out.println("Nom: " + camps[0] + " Cognom: " + camps[1] + " Nota:  
" + camps[2] + " Mòdul: " + camps[3] + " Aprovat: " + camps[4]);  
}  
dades.close();  
fitxer.close();  
}
```

La tercera opció crida el mètode per mostrar aprovats. Obrim l'arxiu i la funcionalitat per llegir-l'ho. Creem dues variables per comptar tant els aprovats com els suspesos.

```
public static void mostrarAprovats() throws IOException {  
    FileReader fitxer;  
    BufferedReader dades;  
    fitxer = new FileReader("estudiants.csv");  
    dades = new BufferedReader(fitxer);  
    String linia;  
    int aprovats = 0;  
    int suspesos = 0;
```

Llegim les línies que continguin dades de l'arxiu amb un bucle while i comprovem si el camp[4] que és el del booleà de aprovats i suspesos conté un true o un false, depenent de la comprovació sumem un valor al comptador de suspesos o aprovats.

```
while ((linia = dades.readLine()) != null) {  
    String[] camps = linia.split(",");  
    if (camps[4].equalsIgnoreCase("true")) {  
        aprovats++;  
    } else {  
        suspesos++;  
    }  
}
```

Al acabar el bucle mostrem la suma total de cada comptador per pantalla.

```
System.out.println("Estudiants aprovats: " + aprovats);  
System.out.println("Estudiants suspesos: " + suspesos);  
dades.close();  
fitxer.close();  
}
```

La quarta opció crida el mètode per eliminar estudiants. Li demanem al usuari el nom i el cognom del estudiant que vol eliminar. I obrim l'arxiu original i en creem un de temporal. Obrim les funcionalitats per poder llegir el primer (original) i escriure en el segon(temporal)

```
public static void eliminarEstudiants(Scanner scanner) throws IOException {  
    System.out.println("Introdueix el nom de l'estudiant que vols eliminar:");  
    String nom = scanner.nextLine();  
    System.out.println("Introdueix el cognom de l'estudiant que vols eliminar:");  
    String cognom = scanner.nextLine();  
    FileReader fitxer;  
    BufferedReader dades;  
    FileWriter fitxerTemp;  
    PrintWriter dadesTemp;  
    fitxer = new FileReader("estudiants.csv");  
    dades = new BufferedReader(fitxer);  
    fitxerTemp = new FileWriter("estudiantsTemp.csv");  
    dadesTemp = new PrintWriter(fitxerTemp);
```

Llegim cada línia del fitxer mentre tinguin dades i comprovem si els camps 0 i 1 son iguals que els que ha introduït el usuari. Si és així amb un booleà els marquem com eliminats.

```
boolean eliminat = false;  
String linia;  
while ((linia = dades.readLine()) != null) {  
String[] camps = linia.split(",");  
if (camps[0].equalsIgnoreCase(nom) &&  
camps[1].equalsIgnoreCase(cognom)) {  
eliminat = true;
```

En el cas de que la línia no tingui coincidències, no el marquem com eliminat i escrivim en una línia de l'arxiu temporal. I tanquem els arxius.

```
} else {  
dadesTemp.println(linia);  
}  
}  
dades.close();  
fitxer.close();  
dadesTemp.close();  
fitxerTemp.close();
```

Si després de recórrer totes les línies hem registrat algun usuari com eliminat vol dir q hi han hagut canvis en els documents. Obrim els arxius, eliminem l'arxiu original i el contingut del arxiu temporal el volquem en l'arxiu original.

```
if (eliminat) {  
File fitxerOriginal = new File("estudiants.csv");  
File fitxerNou = new File("estudiantsTemp.csv");  
fitxerOriginal.delete();  
fitxerNou.renameTo(fitxerOriginal);  
System.out.println("Estudiant eliminat correctament.");  
} else {  
System.out.println("No s'ha trobat l'estudiant.");  
}  
}
```

La cinquena opció del menú crida la funcionalitat per modificar la nota d'un estudiant. Demanem al usuari que introdueix-hi el nom, el cognom, el mòdul , la nota que vol modificar i si l'estudiant està aprovat o no. Guardem totes aquestes dades en les seves variables corresponents.

```
public static void modificarNota(Scanner scanner) throws IOException {  
System.out.println("Introdueix el nom del estudiant del qual vols modificar  
una nota:");  
String nom = scanner.nextLine();  
System.out.println("Introdueix el cognom de l'estudiant del qual vols  
modificar una nota:");  
String cognom = scanner.nextLine();  
System.out.println("Introdueix el mòdul del qual vols modificar la nota:");  
String modul = scanner.nextLine();  
System.out.println("Introdueix el nou valor de la nota:");  
double novaNota = scanner.nextDouble();  
scanner.nextLine();
```

```
System.out.println("Introdueix si ha aprovat (true/false):");  
boolean aprovat = scanner.nextBoolean();  
scanner.nextLine();
```

Obrim l'arxiu original i un de temporal i les funcionalitats per llegir el primer i escriure en el segon.

```
FileReader fitxer = new FileReader("estudiants.csv");  
BufferedReader dades = new BufferedReader(fitxer);  
FileWriter fitxerTemp = new FileWriter("estudiantsTemp.csv");  
PrintWriter dadesTemp = new PrintWriter(fitxerTemp);  
boolean modificat = false;  
String linia;
```

Amb un bucle while llegim cada línia mentres tinguin dades escrites, i comprovem que els camps de nom, cognom i matèria coincideixen.

```
while ((linia = dades.readLine()) != null) {  
String[] camps = linia.split(",");  
if (camps[0].equalsIgnoreCase(nom) &&  
camps[1].equalsIgnoreCase(cognom) &&  
camps[3].equalsIgnoreCase(modul)) {
```

Si hi ha coincidència imprimim les noves variables en una línia del document temporal. I marquem que s'ha modificat algun valor amb el booleà "modificat".

```
dadesTemp.println(nom + "," + cognom + "," + novaNota + "," + modul +  
"," + aprovat);  
modificat = true;
```

Si no hi ha coincidència de igual forma imprimim la línia original en l'arxiu temporal. Això vol dir que les línies mantindran el mateix ordre que l'arxiu original hagin sigut modificades o no. Tanquem els arxius.

```
} else {  
dadesTemp.println(linia);  
}  
}  
dades.close();  
fitxer.close();  
dadesTemp.close();  
fitxerTemp.close();
```

Si hi han hagut modificacions substituïm el contingut de l'arxiu temporal per l'original i informem per pantalla. Si no hi han modificacions no fem canvis entre arxius i també informem per pantalla.

```
if (modificat) {  
File fitxerOriginal = new File("estudiants.csv");  
File fitxerNou = new File("estudiantsTemp.csv");  
fitxerOriginal.delete();  
fitxerNou.renameTo(fitxerOriginal);  
System.out.println("La nota s'ha modificat correctament.");  
} else {  
System.out.println("No s'ha trobat l'estudiant o el mòdul indicat.");  
}  
}
```