

Fitxers 1

1. Projecte Gutenberg

Clase BookTxt

```
import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.io.OutputStream;
import java.util.StringTokenizer;

public class BookTxt{

    private String filePath;

    public BookTxt(String fp){
        this.filePath = fp;
    }

    public void showFile(){

        Reader fr = new Reader(this.filePath);
        String lf = fr.readLine();

        while (lf != null) {

            System.out.println(lf);

            lf = fr.readLine();

        }

        fr.close();
    }

    public int countLines(){

        Reader fr = new Reader(this.filePath);

        String lf = fr.readLine();
        int sum = 1;

        while (lf != null){
            lf = fr.readLine();
```

```

        sum++;
    }

    fr.close();

    return sum;
}

public int countWords(){

    Reader fr = new Reader(this.filePath);

    String lf = fr.readLine();
    int suma = 0;

    while (lf != null){

        StringTokenizer st = new StringTokenizer(lf);

        suma = suma + st.countTokens();

        lf = fr.readLine();
    }

    fr.close();

    return suma;
}

public int countChars(){

    Reader fr = new Reader(this.filePath);

    String lf = fr.readLine();
    int suma = 0;

    while (lf != null){

        if (!lf.isEmpty()){//No se tienen en cuenta las líneas vacías

            lf = lf.replace(" ", "");//Para no tener en cuenta los espacios como
caracteres

            suma = suma + lf.split("").length;
        }

        lf = fr.readLine();
    }
}

```

```

        fr.close();

        return suma;

    }

    public void copy(String outPut){

        File origen = new File(this.filePath);
        File destino = new File(outPut);

        try {
            // A partir del objeto File creamos el fichero fisicamente
            if (destino.createNewFile()){

                InputStream in = new FileInputStream(origen);
                OutputStream out = new FileOutputStream(destino);

                byte[] buf = new byte[1024];
                int len;

                while ((len = in.read(buf)) > 0) {

                    out.write(buf, 0, len);

                }

                in.close();
                out.close();

            } else{
                System.out.println("No ha podido hacer la copia del archivo");
            }
        } catch (IOException ioe) {
            ioe.printStackTrace();
        }

    }

    public int countAWord(String sTextoBuscado){

        Reader fr = new Reader(this.filePath);
        String lf = fr.readLine();
        int contador = 0;

        while (lf != null) {

            //Se distinguen entre mayúsculas y minúsculas
            while (lf.indexOf(sTextoBuscado) > -1) {

                lf =

```

```

lf.substring(lf.indexOf(sTextoBuscado)+sTextoBuscado.length(),lf.length());
        contador++;
    }

    lf = fr.readLine();

}

fr.close();

return contador;

}

public int countAChar(char c){

    Reader fr = new Reader(this.filePath);
    String lf = fr.readLine();
    int posicion, contador = 0;

    while (lf != null) {

        //se busca la primera vez que aparece
        posicion = lf.indexOf(c);

        while (posicion != -1) { //mientras se encuentre el caracter
            contador++;          //se cuenta
            //se sigue buscando a partir de la posición siguiente a la encontrada
            posicion = lf.indexOf(c, posicion + 1);
        }

        lf = fr.readLine();

    }

    fr.close();

    return contador;

}

public String getTitle(){

    Reader fr = new Reader(this.filePath);
    String title = "";

    String lf = fr.readLine();
    int contador = 1;

```

```

while (lf != null){

    lf = fr.readLine();

    contador++;

    if (contador == 9){

        String cadena = lf;
        String[] parts = cadena.split("Title: ");
        title = parts[1];

    }
}

fr.close();

return title;

}

public String getAuthor(){

    Reader fr = new Reader(this.filePath);
    String autor = "";

    String lf = fr.readLine();
    int contador = 1;

    while (lf != null){
        lf = fr.readLine();
        contador++;

        if (contador == 11){

            String cadena = lf;
            String[] parts = cadena.split("Author: ");
            autor = parts[1];

        }
    }

    fr.close();

    return autor;

}

public int countLinesBook(){

    //Usaremos estas variables para saber dónde comienza el contenido del libro
    String titulo = this.getTitle();

```

```

        String inicio_contenido = "*** START OF THIS PROJECT GUTENBERG EBOOK " + titulo
+ " ***";

        boolean empezado = false; //determinamos si ya ha empezado el contenido del libro

        Reader fr = new Reader(this.filePath);
        String lf = fr.readLine();

        int suma = 0;

        while (lf != null) {

            if (lf.equals(inicio_contenido)){
                empezado = true;
            }

            if (empezado && !(lf.equals(inicio_contenido))){
                suma++;
            }

            lf = fr.readLine();

        }

        fr.close();

        return suma;
    }
}

```

Clase Program (es la clase que contiene el método main)

```

public class Program {

    public static void main (String [] args){

        BookTxt libro = new BookTxt("archivo.txt");

        System.out.println(libro.getTitle());
        System.out.println(libro.getAuthor());
        libro.showFile();

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

        String palabra = "EBOOK";
        char character = 'z';
        System.out.println("Líneas del archivo: " + libro.countLines());
        System.out.println("Líneas del libro: " + libro.countLinesBook());
        System.out.println("Número de palabras: " + libro.countWords());
        System.err.println("Número de caracteres: " + libro.countChars());
        System.err.println("Veces que aparece la palabra " + palabra + ": " +
libro.countAWord(palabra));
    }
}

```

```

        System.err.println("Veces que aparece el caracter " + caracter + ": " +
libro.countAChar(caracter));
        libro.copy("copia.txt");
    }
}

```

Resultado de ejecución:

```

ronald@PC-ron:~/dam/m03/archivos/practical/gutenberg$ cd /home/ronald/dam/m03/archivos/practical/gutenberg_9
fig/Code/User/workspaceStorage/0dd754d7f1db9f62a90ec8728731860d/redhat.java/jdt_ws/gutenberg_9
El guardián entre el centeno
J.D. Salinger

rgerg

rgergt

zzz
etgetg

Title: El guardián entre el centeno
Author: J.D. Salinger

*** START OF THIS PROJECT GUTENBERG EBOOK El guardián entre el centeno ***

Este es el texto del libro yatusaes erferzferfereververververververververververververververververver
ververververververver
ververververververver
-----
Líneas del archivo: 19
Líneas del libro: 4
Número de palabras: 36
Número de caracteres: 258
Veces que aparece la palabra EBOOK: 1
Veces que aparece el caracter z: 4
ronald@PC-ron:~/dam/m03/archivos/practical/gutenberg$ █

```

```

copia.txt
1
2   rgerg
3
4   rgergt
5
6   zzz
7   etgetg
8
9   Title: El guardián entre el centeno
10
11  Author: J.D. Salinger
12
13
14  *** START OF THIS PROJECT GUTENBERG EBOOK El guardián entre el centeno ***
15
16  Este es el texto del libro yatusaes erferzferfereververververververververververververververververver
17  ververververververver
18  ververververververver
19

```

2. CSV Training

Clase Training01

```
import java.text.DateFormat;
import java.text.SimpleDateFormat;
import java.util.Date;
import java.util.HashSet;

public class Training01 {

    public Training01(){

    }

    //1. Oficines, ciutats on estan ubicades i vendes actuals.
    public void m1(){

        Reader lector = new Reader("oficinas.dat");
        Writer escritor = new Writer("output1.dat");

        //Leemos la primera linea
        String linea = lector.readLine();

        while (linea != null) {

            String campos[] = linea.split("\\s+");
            escritor.println(campos[0] + "\t" + campos[1] + "\t" + campos[5]);
            linea = lector.readLine();

        }

        escritor.close();

    }

    //2. Nom dels venedors, oficines on treballen, vendes actuals i quota de cada venedor.

    public void m2(){
        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output2.dat");

        //Leemos la primera linea
        String linea = lector.readLine();

        while (linea != null) {

            String campos[] = linea.split("\\s+");
            escritor.println(campos[1] + "\t" + campos[4] + "\t" + campos[9] + "\t" +
campos[8]);
        }
    }
}
```



```

        linea = lector.readLine();

    }

    escritor.close();
}

//3. Nom dels venedors, vendes actuals, quota assignada a cada venedor, quantitat de
desviació de la quota respecte a les vendes efectuades.

public void m3(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output3.dat");

    //Leemos la primera línea
    String linea = lector.readLine();
    String cabecera[] = linea.split("\\s+");
    escritor.println(cabecera[1] + "\t" + cabecera[9] + "\t" + cabecera[8] + "\t" +
"DESVIACIÓN");
    int desv, cuota, ventas;

    //Seguimos con la siguiente línea
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        ventas = 0;
        if (campos[9] != "\\N"){
            ventas = Integer.parseInt(campos[9]);
        }

        cuota = 0;

        if (!campos[8].equals("\\N")){
            cuota = Integer.parseInt(campos[8]);
        }

        desv = ventas-cuota;

        escritor.println(campos[1] + "\t" + campos[9] + "\t" + campos[8] + "\t" + desv);

        linea = lector.readLine();

    }

    escritor.close();
}

//4. Número de comanda, client, producte, quantitat, per comandes amb imports superiors a

```

```

un valor donat (per exemple 25000), ordenats per import.
public void m4(){

}

//5. Import mitjà de les comandes.
public void m5(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output5.dat");

    int contador = 0;
    int suma = 0;
    int importe;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Seguimos a la otra línea
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        importe = Integer.parseInt(campos[7]);

        contador++;
        suma = suma + importe;

        linea = lector.readLine();

    }

    escritor.print(suma/contador);

    escritor.close();
}

//6. Import mitjà de les comandes fetes per un client (per exemple el 2103).
public void m6(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output6.dat");

    int cliente = 2103;
    int cliecod;
    int contador = 0;
    int suma = 0;
    int importe;

    //Leemos la primera línea

```

```

String linea = lector.readLine();
//Seguimos a la otra línea
linea = lector.readLine();

while (linea != null) {

    String campos[] = linea.split("\\s+");

    importe = Integer.parseInt(campos[7]);

    cliecod = Integer.parseInt(campos[2]);

    if (cliecod == cliente){
        contador++;
        suma = suma + importe;
    }

    linea = lector.readLine();

}

if (contador > 0){
    escritor.print(suma/contador);
} else {
    escritor.print("No hay pedidos del cliente " + cliente );
}

escritor.close();
}

//7. Import total de les comandes enviades a cada client.
public void m7(){

    //Primero guardaremos la lista de clientes
    HashSet<Integer> clientes = new HashSet<Integer>();

    Reader lector = new Reader("pedidos.dat");
    int client;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la segunda línea
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        client = Integer.parseInt(campos[2]);

        clientes.add(client);

        linea = lector.readLine();
    }
}

```

```

    }

    //Recorremos el hashset
    Writer escritor = new Writer("output7.dat");

    for (int i : clientes){

        Reader l = new Reader("pedidos.dat");

        int suma = 0;
        int importe = 0;

        String line = l.readLine();
        //Pasamos a la segunda línea
        line = l.readLine();

        while (line != null) {

            String campos[] = line.split("\\s+");

            client = Integer.parseInt(campos[2]);
            importe = Integer.parseInt(campos[7]);

            if (client == i){
                suma = suma + importe;
            }

            line = l.readLine();

        }

        escritor.println("Importe total del cliente " + i + ": " + suma + "€");
    }

    escritor.close();

}

//8. Nom dels venedors i data contracte per contractes posteriors a una data donada (per
exemple el 14 de juny de 1989).

//Conversor de String a Date

public static Date stringToDate(String string){

    Date result = new Date();

    try{

```

```

        DateFormat newFormato = new SimpleDateFormat("yyyy-MM-dd");
        result = newFormato.parse(string);
    } catch (java.text.ParseException e){
        e.printStackTrace();
    }

    return result;
}

//Conversor de Date a String
public static String dateToString(Date date){

    String pattern = "yyyy-MM-dd";

    DateFormat df = new SimpleDateFormat(pattern);

    String result = df.format(date);

    return result;
}

public void m8(){

    String strf = "1989-06-14";
    Date fecha = stringToDate(strf);

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output8.dat");

    //Leemos la primera línea
    String linea = lector.readLine();//Es el encabezado;
    //Leemos la siguiente línea
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        Date f = stringToDate(campos[6]);

        if (f.after(fecha)){
            escritor.println(campos[1] + "\t" + campos[6]);
        }

        linea = lector.readLine();

    }

    escritor.close();

}

//9. Ciutats on hi ha oficines on l'objectiu supera un cert valor (per exemple 10000€) la

```

quota de vendes incrementada en un cert valor (per exemple un 10 %).

```
public void m9(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output9.dat");

    String linea = lector.readLine();
    linea = lector.readLine();
    int objetivo;
    int ventas;

    escritor.println("OFICINA" + "\t" + "OBJETIVO" + "\t" + "VENTAS" + "\t" + "VENTAS
+10%");

    while(linea != null){

        String campos[] = linea.split("\\s+");

        objetivo = Integer.parseInt(campos[4]);
        ventas = Integer.parseInt(campos[5]);

        if (objetivo > 10000){
            ventas = ventas + ((ventas * 10) / 100);
            escritor.println(campos[1] + "\t" + campos[4] + "\t" + campos[5] + "\t" +
ventas);

        }

        linea = lector.readLine();

    }

    escritor.close();

}
```

//10. Ciutats, objectius, vendes actuals i vendes actuals en % respecte dels objectius.

```
public void m10(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output10.dat");

    String linea = lector.readLine();
    linea = lector.readLine();
    int objetivo;
    int ventas;
    int porcentaje;

    escritor.println("CIUDAD" + "\t" + "OBJETIVO" + "\t" + "VENTAS" + "\t" + "% DE
VENTAS");
```

```

while(linea != null){

    String campos[] = linea.split("\\s+");

    objetivo = Integer.parseInt(campos[4]);
    ventas = Integer.parseInt(campos[5]);

    porcentaje = 100 * (ventas - objetivo) / ventas;

    escritor.println(campos[1] + "\t" + objetivo + "\t" + ventas + "\t" + porcentaje
+ "%");

    linea = lector.readLine();
}

escritor.close();
}
}

```

Clase Training02

```

public class Training02{

    //11. Ciutats, objectius i vendes d'una regió (per exemple "Oeste"), on les vendes
actuais superen els objectius, ordenats per ciutats.
    public void m11(){

        Reader lector = new Reader("oficinas.dat");
        Writer escritor = new Writer("output11.dat");
        String region = "Oeste";
        int ventas;
        int objetivo;

        //Leemos la primera línea y pasamos a la segunda
        String linea = lector.readLine();
        linea = lector.readLine();

        while (linea != null) {

            String campos[] = linea.split("\\s+");

            objetivo = Integer.parseInt(campos[4]);
            ventas = Integer.parseInt(campos[5]);

            if (campos[2].equals(region)){

                if (ventas > objetivo){

                    escritor.println(campos[1] + "\t" + objetivo + "\t" + ventas + "\t" +

```

```

campos[2]);
        }
    }

    linea = lector.readLine();

}

escritor.close();

}

//12. Valors mitjans dels objectius i vendes actuals d'una regió(per exemple "Oeste").
public void m12(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output12.dat");
    String region = "Oeste";
    int ventas;
    int objetivo;
    int sumaObj = 0;
    int contadorObj = 0;
    int sumaVent = 0;
    int contadorVent = 0;

    //Leemos la primera línea y pasamos a la segunda
    String linea = lector.readLine();
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        objetivo = Integer.parseInt(campos[4]);
        ventas = Integer.parseInt(campos[5]);

        if (campos[2].equals(region)){

            contadorObj++;
            sumaObj = sumaObj + objetivo;

            contadorVent++;
            sumaVent = sumaVent + ventas;

        }

        linea = lector.readLine();

    }

    escritor.println("Oficinas de la región " + region + ": ");
    escritor.println("MEDIA DE OBJETIVOS" + "\t" + "MEDIA DE VENTAS");
    escritor.println((sumaObj/contadorObj) + "\t" + (sumaVent/contadorVent) );
}

```



```

        escritor.close();
    }

    //13. Noms, oficines i dates de contracte de tots els venedors.
    public void m13(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output13.dat");

        //Escribimos el encabezado
        String linea = lector.readLine();
        String encabezados[] = linea.split("\\s+");
        escritor.println(encabezados[1] + "\t" + encabezados[4] + "\t" + encabezados[6]);

        //Pasamos a la siguiente línea
        linea = lector.readLine();

        while(linea != null){

            String campos[] = linea.split("\\s+");

            escritor.println(campos[1] + "\t" + campos[4] + "\t" + campos[6]);

            linea = lector.readLine();
        }

        escritor.close();
    }

    //14. Nom, quota i vendes actuals d'un empleat ( per exemple el 107).
    public void m14(){

        int empleado = 107;
        int id;

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output14.dat");

        //Escribimos el encabezado
        String linea = lector.readLine();
        String encabezados[] = linea.split("\\s+");
        escritor.println(encabezados[0] + "\t" + encabezados[8] + "\t" + encabezados[9]);

        //Pasamos a la siguiente línea
        linea = lector.readLine();

        while(linea != null){

            String campos[] = linea.split("\\s+");

```

```

        id = Integer.parseInt(campos[0]);

        if (id == empleado){
            escritor.println(campos[0] + "\t" + campos[8] + "\t" + campos[9]);
        }

        linea = lector.readLine();
    }

    escritor.close();

}

//15. Valor mitjà de totes les vendes.
public void m15(){

    //Se supondrà que se refereix a les vendes del arxivo repventas.dat
    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output15.dat");
    int suma = 0;
    int contador = 0;
    int ventas;

    //Escribimos el encabezado
    String linea = lector.readLine();
    //Pasamos a la siguiente línea
    linea = lector.readLine();

    while(linea != null){

        String campos[] = linea.split("\\s+");

        ventas = Integer.parseInt(campos[9]);

        suma = suma + ventas;
        contador++;

        linea = lector.readLine();
    }

    escritor.println(suma/contador);

    escritor.close();

}

//16. Noms i dates del contracte dels venedors tal que les seves vendes actuals superen
un valor (per exemple 500.000 €).
public void m16(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output16.dat");
    int valor = 500000;

```

```

int ventas;

//Escribimos el encabezado
String linea = lector.readLine();
String encabezados[] = linea.split("\\s+");
escritor.println(encabezados[1] + "\t" + encabezados[6] + "\t" + encabezados[9]);
//Pasamos a la siguiente linea
linea = lector.readLine();

while(linea != null){

    String campos[] = linea.split("\\s+");

    ventas = Integer.parseInt(campos[9]);

    if (ventas > valor){

        escritor.println(campos[1] + "\t" + campos[6] + "\t" + ventas);
    }

    linea = lector.readLine();
}

escritor.close();
}

//17. Valor de l'inventari de cada producte.
public void m17(){

    Reader lector = new Reader("productos.dat");
    Writer escritor = new Writer("output17.dat");
    int valor;
    int precio_ud;
    int stock;

    //Escribimos el encabezado
    String linea = lector.readLine();
    String encabezados[] = linea.split("\\s+");
    escritor.println(encabezados[1] + "\t" + "VALOR DE STOCK");
    //Pasamos a la siguiente linea
    linea = lector.readLine();

    while(linea != null){

        String campos[] = linea.split("\\s+");

        precio_ud = Integer.parseInt(campos[3]);
        stock = Integer.parseInt(campos[4]);

        valor = precio_ud * stock;

        escritor.println(campos[1] + "\t" + valor);
    }
}

```

```

        linea = lector.readLine();
    }

    escritor.close();
}

//18. Nom, quota i quota incrementada en un valor (per exemple un 3%) de les vendes
actuals.
public void m18(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output18.dat");
    int porcentaje = 3;
    int cuota;
    int incremento;

    //Escribimos el encabezado
    String linea = lector.readLine();
    String encabezados[] = linea.split("\\s+");
    escritor.println(encabezados[1] + "\t" + encabezados[8] + "\t" + "QUOTA
INCREMENTADA " + porcentaje + "%");
    //Pasamos a la siguiente línea
    linea = lector.readLine();

    while(linea != null){

        String campos[] = linea.split("\\s+");

        if (!campos[8].equals("\\N")){

            cuota = Integer.parseInt(campos[8]);
            incremento = cuota + ((cuota*porcentaje)/100);

            escritor.println(campos[1] + "\t" + campos[8] + "\t" + incremento);
        }

        linea = lector.readLine();
    }

    escritor.close();
}

//19. Ciutat, vendes actuals, objectius de les oficines on les vendes actuals són
inferiors a un valor (per exemple 80%) dels seus objectius.
public void m19(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output19.dat");
    int ventas;
    int objetivo;
    int porcentajeObj;
    int porcentaje = 80;

```

```

//Leemos la primera línea y pasamos a la segunda
String linea = lector.readLine();
String encabezados[] = linea.split("\\s+");
escritor.println("Ciudades con ventas inferiores al " + porcentaje + "% " + "de
su objetivo \n");
escritor.println(encabezados[1] + "\t" + encabezados[5] + "\t" + encabezados[4]);
linea = lector.readLine();

while (linea != null) {

    String campos[] = linea.split("\\s+");

    if (!campos[4].equals("\\N") && !campos[5].equals("\\N")){

        objetivo = Integer.parseInt(campos[4]);
        ventas = Integer.parseInt(campos[5]);

        porcentajeObj = (objetivo * porcentaje)/100;

        if (ventas < porcentajeObj){
            escritor.println(campos[1] + "\t" + campos[5] + "\t" + campos[4]);
        }

    }

    linea = lector.readLine();

}

escritor.close();
}

//20. Ciutats on hi ha oficines no dirigides per un empleat (per exemple el 108).
public void m20(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output20.dat");
    int empleado = 108;
    int director;

    //Leemos la primera línea y pasamos a la segunda
    String linea = lector.readLine();
    String encabezados[] = linea.split("\\s+");
    escritor.println("Ciudades no dirigidas por el empleado " + empleado + "\n");
    escritor.println(encabezados[1] + "\t" + encabezados[3]);
    linea = lector.readLine();

    while (linea != null) {

        String campos[] = linea.split("\\s+");

        director = Integer.parseInt(campos[3]);

```

```

        if (director != empleado){
            escritor.println(campos[1] + "\t" + campos[3]);
        }

        linea = lector.readLine();

    }

    escritor.close();
}
}

```

Clase Training03

```

public class Training03{

    //21. Nom de venedors que les seves vendes actuals són superiors a les seves quotes.
    public void m21(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output21.dat");
        int ventas;
        int cuotas;

        //Leemos la primera línea
        String linea = lector.readLine();
        //Leemos la siguiente línea
        linea = lector.readLine();

        while(linea != null){

            String campos[] = linea.split("\\s+");

            if (!campos[8].equals("\\N") && !campos[9].equals("\\N")){

                cuotas = Integer.parseInt(campos[8]);
                ventas = Integer.parseInt(campos[9]);

                if (ventas > cuotas){
                    escritor.println(campos[1]);
                }
            }

            linea = lector.readLine();

        }

        escritor.close();
    }
}

```

```

    }

    //22. Nombre de comanda i import on l'import de la comanda estigui entre dos valor
    donats (per exemple entre 30.000 i 40.000€).
    public void m22(){

        Reader lector = new Reader("pedidos.dat");
        Writer escritor = new Writer("output22.dat");
        int maximo = 40000;
        int minimo = 30000;
        int importe;

        //Leemos la primera línea
        String linea = lector.readLine();
        String encabezado[] = linea.split("\\s+");
        escritor.println(encabezado[0] + "\t" + encabezado[7]);
        //Leemos la siguiente línea
        linea = lector.readLine();

        while(linea != null){

            String campos[] = linea.split("\\s+");

            importe = Integer.parseInt(campos[7]);

            if (importe >= minimo && importe <= maximo){

                escritor.println(campos[0] + "\t" + campos[7]);
            }

            linea = lector.readLine();

        }

        escritor.close();
    }

    //23. Nom, vendes actuals, i quota dels venedors d'aquells venedors que les seves vendes
    no estan
    //entre dos valors donats (per exemple el 80% i el 120%) de la seva quota.
    public void m23(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output23.dat");
        int ventas;
        int cuotas;
        int minP = 80;
        int maxP = 120;
        int minC;
        int maxC;

        //Leemos la primera línea
        String linea = lector.readLine();

```

```

//Leemos la siguiente línea
línea = lector.readLine();

while(línea != null){

    String campos[] = línea.split("\\s+");

    if (!campos[8].equals("\\N") && !campos[9].equals("\\N")){

        cuotas = Integer.parseInt(campos[8]);
        ventas = Integer.parseInt(campos[9]);

        minC = (cuotas * minP ) / 100;
        maxC = (cuotas * maxP ) / 100;

        if (ventas >= minC && ventas <= maxC){

            escritor.println(campos[1] + "\t" + campos[8] + "\t" + campos[9]);
        }

    }

    línea = lector.readLine();

}

escritor.close();

}

//24. Nom dels venedors que treballen a tres ciutats (per exemple a New York, Atlanta o
Denver).
public void m24(){

    Reader lector = new Reader("repventas.dat");
    Reader lectorOficina = new Reader("oficinas.dat");
    Writer escritor = new Writer("output24.dat");
    String city1 = "NewYork";
    String city2 = "Atlanta";
    String city3 = "Denver";
    int oficinas[] = new int[3];
    int ofinum;
    int ofinumRepventas;
    int contador = 0;

    String l = lectorOficina.readLine();
    l = lectorOficina.readLine();

    //Este primer bucle nos servirá para saber el ID de las oficinas
    while(l != null){

```



```

        String c[] = l.split("\\s+");

        if (c[1].equals(city1) || c[1].equals(city2) || c[1].equals(city3)){

            ofinum = Integer.parseInt(c[0]);
            oficinas[contador] = ofinum;
            contador++;

        }

        l = lectorOficina.readLine();

        if (contador >= 3){//Condición de salida
            l = null;
        }

    }

    //Leemos la primera línea
    String linea = lector.readLine();
    //Leemos la siguiente línea
    linea = lector.readLine();

    while(linea != null){

        String campos[] = linea.split("\\s+");

        if (!campos[4].equals("\\N")){

            ofinumRepventas = Integer.parseInt(campos[4]);

            if (ofinumRepventas == oficinas[0] || ofinumRepventas == oficinas[1] ||
ofinumRepventas == oficinas[2] ){

                escritor.println(campos[1] + "\t" + campos[4]);
            }

        }

        linea = lector.readLine();

    }

    escritor.close();
}

//25. Número de comanda, data comanda, import de totes les comandes fetes un dijous del
mes de gener de 1990. (Eren els dies 4, 11,18 i 25).
public void m25(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output25.dat");

```

```

int dias[] = {4,11,18,25};
String fecha = "1990-01-";//Contendrá el mes y año pero no el día

//Leemos la primera línea
String linea = lector.readLine();
//Leemos la siguiente línea
linea = lector.readLine();

while(linea != null){

    String campos[] = linea.split("\\s+");

    for (int i = 0; i < dias.length; i++) {

        if (campos[1].equals( fecha + String.valueOf(dias[i]))){

            escritor.println(campos[0] + "\t" + campos[1] + "\t" + campos[7]);

        }

    }

    linea = lector.readLine();

}

escritor.close();

}

//26. Noms dels venedors sense oficines assignades.
public void m26(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output26.dat");

    //Leemos la primera línea
    String linea = lector.readLine();
    //Leemos la siguiente línea
    linea = lector.readLine();

    while(linea != null){

        String campos[] = linea.split("\\s+");

        if (campos[4].equals("\\N")){

            escritor.println(campos[1]);

        }

        linea = lector.readLine();

    }

}

```

```

        escritor.close();
    }

    //27. Nom venedors, quotes i vendes actuals dels venedors que tenen vendes per sota de
    la seva quota o vendes inferiors a un cert valor (per exemple 300.000 €.)
    public void m27(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output27.dat");
        int ventas;
        int cuotas;
        int valor = 300000;

        //Leemos la primera línea
        String linea = lector.readLine();
        //Leemos la siguiente línea
        linea = lector.readLine();

        while(linea != null){

            String campos[] = linea.split("\\s+");

            if (!campos[8].equals("\\N") && !campos[9].equals("\\N")){

                cuotas = Integer.parseInt(campos[8]);
                ventas = Integer.parseInt(campos[9]);

                if (ventas < cuotas || ventas < valor){
                    escritor.println(campos[1] + "\t" + campos[8] + "\t" + campos[9]);
                }

            }

            linea = lector.readLine();

        }

        escritor.close();
    }

    //28. Nom venedors, quotes i vendes actuals dels venedors que tenen vendes per sota de
    la seva quota i vendes inferiors a un cert valor (per exemple 300.000€).
    public void m28(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output28.dat");
        int ventas;
        int cuotas;
        int valor = 300000;

        //Leemos la primera línea
        String linea = lector.readLine();

```

```

//Leemos la siguiente línea
línea = lector.readLine();

while(línea != null){

    String campos[] = línea.split("\\s+");

    if (!campos[8].equals("\\N") && !campos[9].equals("\\N")){

        cuotas = Integer.parseInt(campos[8]);
        ventas = Integer.parseInt(campos[9]);

        if (ventas < cuotas && ventas < valor){
            escritor.println(campos[1] + "\t" + campos[8] + "\t" + campos[9]);
        }

    }

    línea = lector.readLine();

}

escritor.close();
}

//29. Nom venedors, quotes i vendes actuals dels venedors que tenen vendes per sota de
la seva quota
//però les seves vendes no són inferiors a un cert valor (per exemple 150.000 €).
public void m29(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output29.dat");
    int ventas;
    int cuotas;
    int valor = 150000;

    //Leemos la primera línea
    String línea = lector.readLine();
    //Leemos la siguiente línea
    línea = lector.readLine();

    while(línea != null){

        String campos[] = línea.split("\\s+");

        if (!campos[8].equals("\\N") && !campos[9].equals("\\N")){

            cuotas = Integer.parseInt(campos[8]);
            ventas = Integer.parseInt(campos[9]);

            if (ventas < cuotas && ventas >= valor){
                escritor.println(campos[1] + "\t" + campos[8] + "\t" + campos[9]);
            }

        }

    }

}

```

```

    }

    linea = lector.readLine();

}

escritor.close();
}

//30. Ciutat, regió i vendes ordenades per regió i ciutat.
public void m30(){

}

}

```

Clase Training04

```

public class Training04{

    //31. Ciutat i regió de les oficines ordenades pel millor rendiment en les vendes.
    public void m31(){

    }

    //32. Fabricant i producte de les comandes superiors a un cert valor (per exemple 30.000
    e).
    public void m32(){

        Reader lector = new Reader("pedidos.dat");
        Writer escritor = new Writer("output32.dat");
        int valor = 30000;
        int importe;

        //Leemos la primera línea
        String linea = lector.readLine();
        String encabezados [] = linea.split("\\s+");
        escritor.println(encabezados[4] + "\t" + encabezados[5] + "\t" + encabezados[7]);
        //Pasamos a la siguiente
        linea = lector.readLine();

        while(linea != null){

            String campos [] = linea.split("\\s+");

            importe = Integer.parseInt(campos[7]);

            if (importe > valor){

                escritor.println(campos[4] + "\t" + campos[5] + "\t" + importe);
            }

        }
    }
}

```

```

        linea = lector.readLine();

    }

    escritor.close();

}

//33. Número de comanda, import, empresa, límit de crèdit del client.
public void m33(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output33.dat");
    int cliente; //Se usará en pedidos.dat
    int idCliente; //Se usará en clientes.dat
    int limCredit = 0;
    String empresa = "";

    //Leemos la primera línea
    String linea = lector.readLine();
    String encabezados [] = linea.split("\\s+");
    escritor.println(encabezados[0] + "\t" + encabezados[7]);
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

        String campos [] = linea.split("\\s+");

        cliente = Integer.parseInt(campos[2]);

        Reader lectorClientes = new Reader("clientes.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaCliente = lectorClientes.readLine();
        lineaCliente = lectorClientes.readLine();
        boolean salir = false;

        while(salir == false){

            String c[] = lineaCliente.split("\\s+");

            idCliente = Integer.parseInt(c[0]);
            limCredit = Integer.parseInt(c[3]);
            empresa = c[1];

            lineaCliente = lectorClientes.readLine();

```

```

        //Condición de salida
        if (cliente == idCliente){
            salir = true;
        }
    }

    escritor.println(campos[0] + "\t" + campos[7] + "\t" + empresa + "\t" +
limCredit);

    linea = lector.readLine();

}

escritor.close();
}

//34. Nom dels venedors, ciutat i oficines on treballen.
public void m34(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output34.dat");
    int oficina; //Se usará en repventas.dat
    int idOficina; //Se usará en oficinas.dat
    String ciudad = "";

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

        String campos [] = linea.split("\\s+");

        if (!campos[4].equals("\\N")){
            oficina = Integer.parseInt(campos[4]);

            Reader lectorOficinas = new Reader("oficinas.dat");

            //Primera línea y pasamos a la siguiente;
            String lineaOficina = lectorOficinas.readLine();
            lineaOficina = lectorOficinas.readLine();
            boolean salir = false;

            while(salir == false){

                String c[] = lineaOficina.split("\\s+");

                idOficina = Integer.parseInt(c[0]);

```

```

        ciudad = c[1];

        lineaOficina = lectorOficinas.readLine();

        //Condición de salida
        if (idOficina == oficina){
            salir = true;
        }
    }

    escritor.println(campos[1] + "\t" + ciudad + "\t" + oficina);
}

    linea = lector.readLine();

}

    escritor.close();
}

//35. Ciutats, nombres dels directors i càrrecs de totes les oficines.
public void m35(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output35.dat");
    int director;
    int vendedor;
    String cargo = "";
    String nombre = "";

    //Leemos la primera línea
    String linea = lector.readLine();
    escritor.println("CIUDAD" + "\t" + "DIRECTOR" + "\t" + "Cargo");
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

        String campos [] = linea.split("\\s+");

        director = Integer.parseInt(campos[3]);

        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();
        boolean salir = false;

```



```

        while(salir == false){

            String c[] = lineaRep.split("\\s+");

            vendedor = Integer.parseInt(c[0]);
            cargo = c[5];
            nombre = c[1];

            lineaRep = lectorRep.readLine();

            //Condición de salida
            if (vendedor == director){
                salir = true;
            }
        }

        escritor.println(campos[1] + "\t" + nombre + "\t" + cargo);

        linea = lector.readLine();

    }

    escritor.close();
}

//36. Ciutats, nombres dels directors i càrrecs de les oficines amb objectius superiors
a un cert valor (per exemple 600.000 €).
public void m36(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output36.dat");
    int valor = 600000;
    int objetivo;
    int director;
    int vendedor;
    String cargo = "";
    String nombre = "";

    //Leemos la primera línea
    String linea = lector.readLine();
    escritor.println("CIUDAD" + "\t" + "DIRECTOR" + "\t" + "Cargo");
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

        String campos [] = linea.split("\\s+");

        if (!campos[4].equals("\\N")){

```

```

        director = Integer.parseInt(campos[3]);
        objetivo = Integer.parseInt(campos[4]);

        if (objetivo > valor){
            Reader lectorRep = new Reader("repventas.dat");

            //Primera línea y pasamos a la siguiente;
            String lineaRep = lectorRep.readLine();
            lineaRep = lectorRep.readLine();
            boolean salir = false;

            while(salir == false){

                String c[] = lineaRep.split("\\s+");

                vendedor = Integer.parseInt(c[0]);
                cargo = c[5];
                nombre = c[1];

                lineaRep = lectorRep.readLine();

                //Condición de salida
                if (vendedor == director){
                    salir = true;
                }
            }

            escritor.println(campos[1] + "\t" + nombre + "\t" + cargo);
        }

        linea = lector.readLine();
    }

    escritor.close();
}

//37. Número de comanda, import i descripció.
public void m37(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output37.dat");
    String producto; //Se usará en pedidos.dat
    String id; //Se usará en productos.dat
    String descripcion = "";

```

```

//Leemos la primera línea
String linea = lector.readLine();
escritor.println("Nº PEDIDO" + "\t" + "IMPORTE" + "\t" + "DESCRIPCION");
//Pasamos a la siguiente
linea = lector.readLine();

while(linea != null){

    String campos [] = linea.split("\\s+");

    producto = campos[5];

    Reader lectorProd = new Reader("productos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaProd = lectorProd.readLine();
    lineaProd = lectorProd.readLine();
    boolean salir = false;

    while(salir == false){

        String c[] = lineaProd.split("\\s+");

        id = c[1];
        descripcion = c[2];

        lineaProd = lectorProd.readLine();

        //Condición de salida
        if (producto.equals(id)){
            salir = true;
        }
    }

    escritor.println(campos[0] + "\t" + campos[7] + "€" + "\t" + descripcion);

    linea = lector.readLine();

}

escritor.close();
}

//38. Número de comanda, import, empresa, nom del venedor de les comandes amb import
superior a un cert valor (per exemple 25.000 €).
public void m38(){

    Reader lector = new Reader("pedidos.dat");

```

```

Writer escritor = new Writer("output38.dat");
int idCliente;
int vendedor;
int idVendedor;
int cliente;
int importe;
int valor = 25000;
String nombreEmpresa = "";
String nombreVendedor = "";

//Leemos la primera línea
String linea = lector.readLine();
escritor.println("N° PEDIDO" + "\t" + "IMPORTE" + "\t" + "EMPRESA" + "\t" +
"VENDEDOR");
//Pasamos a la siguiente
linea = lector.readLine();

while(linea != null){

    String campos [] = linea.split("\\s+");

    cliente = Integer.parseInt(campos[2]);
    vendedor = Integer.parseInt(campos[3]);
    importe = Integer.parseInt(campos[7]);

    if (importe > valor){

        //Datos del cliente
        Reader lectorClientes = new Reader("clientes.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaCliente = lectorClientes.readLine();
        lineaCliente = lectorClientes.readLine();
        boolean salir = false;

        while(salir == false){

            String c[] = lineaCliente.split("\\s+");

            idCliente = Integer.parseInt(c[0]);
            nombreEmpresa = c[1];

            lineaCliente = lectorClientes.readLine();

            //Condición de salida
            if (cliente == idCliente){
                salir = true;
            }
        }

        //Datos del vendedor

```

```

        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();
        boolean exit = false;

        while(exit == false){

            String c[] = lineaRep.split("\\s+");

            idVendedor = Integer.parseInt(c[0]);
            nombreVendedor = c[1];

            lineaRep = lectorRep.readLine();

            //Condición de salida
            if (vendedor == idVendedor){
                exit = true;
            }
        }

        escritor.println(campos[0] + "\t" + campos[7] + "€" + "\t" + nombreEmpresa
+ "\t" + nombreVendedor);
    }

    linea = lector.readLine();

}

escritor.close();
}

//39. Número de comanda, import, empresa, nom del venedor i ciutat de l'oficina del
venedor de les comandes amb import superior a un cert valor
// (per exemple 25.000 €).
public void m39(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output39.dat");
    int idCliente;
    int vendedor;
    int idVendedor;
    int oficina = 0;
    int idOficina;
    int cliente;
    int importe;
    int valor = 25000;
    String nombreEmpresa = "";
    String nombreVendedor = "";
    String ciudad = "";

```

```

//Leemos la primera línea
String linea = lector.readLine();
escritor.println("N° PEDIDO" + "\t" + "IMPORTE" + "\t" + "EMPRESA" + "\t" +
"VENDEDOR" + "\t" + "CIUDAD DE LA OFICINA");
//Pasamos a la siguiente
linea = lector.readLine();

while(linea != null){

    String campos [] = linea.split("\\s+");

    cliente = Integer.parseInt(campos[2]);
    vendedor = Integer.parseInt(campos[3]);
    importe = Integer.parseInt(campos[7]);

    if (importe > valor){

        //Datos del cliente
        Reader lectorClientes = new Reader("clientes.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaCliente = lectorClientes.readLine();
        lineaCliente = lectorClientes.readLine();
        boolean salir = false;

        while(salir == false){

            String c[] = lineaCliente.split("\\s+");

            idCliente = Integer.parseInt(c[0]);
            nombreEmpresa = c[1];

            lineaCliente = lectorClientes.readLine();

            //Condición de salida
            if (cliente == idCliente){
                salir = true;
            }
        }

        //Datos del vendedor
        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();
        boolean exit = false;

        while(exit == false){

```

```

String c[] = lineaRep.split("\\s+");

idVendedor = Integer.parseInt(c[0]);
nombreVendedor = c[1];

if (!c[4].equals("\\N")){

    oficina = Integer.parseInt(c[4]);

    //Datos de la oficina
    Reader lectorOficinas = new Reader("oficinas.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaOficina = lectorOficinas.readLine();
    lineaOficina = lectorOficinas.readLine();
    boolean salirOficina = false;

    while(salirOficina == false){

        String fields[] = lineaOficina.split("\\s+");

        idOficina = Integer.parseInt(fields[0]);

        ciudad = fields[1];

        lineaOficina = lectorOficinas.readLine();

        //Condición de salida
        if (idOficina == oficina){
            salirOficina = true;
        }
    }

} else {

    ciudad = "So oficina";
}

lineaRep = lectorRep.readLine();

//Condición de salida
if (vendedor == idVendedor){
    exit = true;
}

}

escritor.println(campos[0] + "\t" + campos[7] + "€" + "\t" + nombreEmpresa
+ "\t" + nombreVendedor + "\t" + ciudad);
}

```

```

        linea = lector.readLine();

    }

    escritor.close();
}

//40. Número de comanda, import, data de la comanda i venedor de les comandes fetes els
dies que algun venedor va ser contractat.
public void m40(){

    Reader lector = new Reader("pedidos.dat");
    Writer escritor = new Writer("output40.dat");
    int vendedor;
    int idVendedor;
    int importe;
    String nombreVendedor = "";
    String fecha = "";
    String fContrato;

    //Leemos la primera línea
    String linea = lector.readLine();
    escritor.println("Nº PEDIDO" + "\t" + "IMPORTE" + "\t" + "FECHA" + "\t" +
"VENDEDOR");
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

        String campos [] = linea.split("\\s+");

        vendedor = Integer.parseInt(campos[3]);
        importe = Integer.parseInt(campos[7]);
        fecha = campos[1];

        //Datos del vendedor
        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();
        boolean exit = false;

        while(exit == false){

            String c[] = lineaRep.split("\\s+");

            idVendedor = Integer.parseInt(c[0]);
            nombreVendedor = c[1];

            lineaRep = lectorRep.readLine();

```



```

        //Condición de salida
        if (vendedor == idVendedor){
            exit = true;
        }
    }

    //Otra vuelta para comparar fechas
    Reader lectorFecha = new Reader("repventas.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaFecha = lectorFecha.readLine();
    lineaFecha = lectorFecha.readLine();

    while(lineaFecha != null){

        String fields[] = lineaFecha.split("\\s+");

        fContrato = fields[6];

        lineaFecha = lectorFecha.readLine();

        if (fContrato.equals(fecha)){
            lineaFecha = null;
            escritor.println(campos[0] + "\t" + campos[7]+ "€" + "\t" + fecha +
"\t" + nombreVendedor);
        }

    }

    linea = lector.readLine();

}

escritor.close();
}
}

```

Clase Training05

```

import java.util.HashSet;

public class Training05{

    //41. Total de quotes i vendes actuals.
    public void m41(){

        //Se supondrá que se refiere a oficinas.dat
        Reader lector = new Reader("oficinas.dat");
        Writer escritor = new Writer("output41.dat");
        int sumaCuotas = 0;
    }
}

```

```

int sumaVentas = 0;
int cuota;
int ventas;

//Leemos la primera línea
String linea = lector.readLine();
//Pasamos a la siguiente
linea = lector.readLine();

escritor.println("TOTAL DE CUOTAS" + "\t" + "TOTAL DE VENTAS");

while(linea != null){

    String campos[] = linea.split("\\s+");

    cuota = Integer.parseInt(campos[4]);
    ventas = Integer.parseInt(campos[5]);

    sumaCuotas = sumaCuotas + cuota;
    sumaVentas = sumaVentas + ventas;

    linea = lector.readLine();
}

escritor.println(sumaCuotas + "\t" + sumaVentas);

escritor.close();
}

//42. Import total de les comandes d'un venedor (per exemple Bill Adams).
public void m42(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output42.dat");
    int suma = 0;
    int importe;
    int ventas;
    int idVendedor;
    int id = 0;
    String vendedor = "Bill Adams";
    String nombre;
    String apellido;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

    escritor.println("IMPORTE TOTAL DE LOS PEDIDOS DE " + vendedor);

    //Extraemos el id del vendedor
    while(linea != null){

```

```

        String campos[] = linea.split("\\s+");

        id = Integer.parseInt(campos[0]);
        nombre = campos[1];
        apellido = campos[2];

        linea = lector.readLine();

        if (vendedor.equals(nombre + " " + apellido)){
            linea = null;
        }
    }

    //Sumamos los importes
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idVendedor = Integer.parseInt(c[3]);
        importe = Integer.parseInt(c[7]);

        if (idVendedor == id){
            suma = suma + importe;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    escritor.println(suma);

    escritor.close();
}

//43. Preu mitjà dels productes d'un cert fabricant(per exemple ACI).
public void m43(){

    Reader lector = new Reader("productos.dat");
    Writer escritor = new Writer("output43.dat");
    int precio;
    int suma = 0;
    int contador = 0;
    int media;
    String fabricante = "aci";

    //Leemos la primera línea
    String linea = lector.readLine();

```

```

        //Pasamos a la siguiente
        linea = lector.readLine();

        escritor.println("PRECIO MEEDIO DE LOS PRODUCTOS DEL FABRICANTE" + "\t" +
fabricante);

        while(linea != null){

            String campos[] = linea.split("\\s+");

            precio = Integer.parseInt(campos[3]);

            if (campos[0].equals(fabricante)){
                suma = suma + precio;
                contador++;
            }

            linea = lector.readLine();
        }

        media = suma/contador;

        escritor.println(media + "€");

        escritor.close();
    }

```

```

//44. Import mitjà de les comandes d'una empresa (per exemple Acme Mfg).
public void m44(){

```

```

    Reader lector = new Reader("clientes.dat");
    Writer escritor = new Writer("output44.dat");
    int suma = 0;
    int contador = 0;
    int media;
    int importe;
    int ventas;
    int idCliente;
    int id = 0;
    String cliente = "AcmeMfg.";
    String nombre;

```

```

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

```

```

    escritor.println("IMPORTE MEDIO DE LOS PEDIDOS DE " + cliente);

```

```

    //Extraemos el id del vendedor
    while(linea != null){

```

```

        String campos[] = linea.split("\\s+");

        id = Integer.parseInt(campos[0]);
        nombre = campos[1];

        linea = lector.readLine();

        if (nombre.equals(cliente)){
            linea = null;
        }
    }

    //Sumamos los importes
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idCliente = Integer.parseInt(c[2]);
        importe = Integer.parseInt(c[7]);

        if (idCliente == id){
            suma = suma + importe;
            contador++;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    media = suma/contador;

    escritor.println(media + "€");

    escritor.close();
}

//45. Quotes màxima i mínima dels venedors.
public void m45(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output45.dat");
    int max;
    int min;
    int cuota;

    //Leemos la primera línea
    String linea = lector.readLine();

```

```

//Pasamos a la siguiente
linea = lector.readLine();

//Asignamos el mayor y el menor
String fields[] = linea.split("\\s+");

max = Integer.parseInt(fields[8]);
min = Integer.parseInt(fields[8]);

escritor.println("TOTAL DE CUOTAS DE LOS VENDEDORES");

while(linea != null){

    String campos[] = linea.split("\\s+");

    if(!campos[8].equals("\\N")){

        cuota = Integer.parseInt(campos[8]);

        if (cuota > max){
            max = cuota;
        } else if (cuota < min){
            min = cuota;
        }

    }

    linea = lector.readLine();
}

escritor.println("CUOTA MÁXIMA: " + max);
escritor.println("CUOTA MÍNIMA: " + min);

escritor.close();
}

//46. Millor rendiment de vendes de tots els venedors.
public void m46(){

    //Se va a suponer que se refiere al importe de los pedidos de cada vendedor
    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output46.dat");
    int suma = 0;
    int importe = 0;
    int ventas;
    int idVendedor;
    int id = 0;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

```

```

//Extraemos el id del vendedor
while(linea != null){

    String campos[] = linea.split("\\s+");

    suma = 0;
    id = Integer.parseInt(campos[0]);

    //Recorremos pedidos.dat
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idVendedor = Integer.parseInt(c[3]);
        importe = Integer.parseInt(c[7]);

        if (idVendedor == id){
            suma = suma + importe;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    escritor.println(id + "\t" + suma + "€");

    linea = lector.readLine();
}

escritor.close();

}

//47. Quants clients té l'empresa de recanvis.
public void m47(){
    //Se hace la suposición de que la empresa de recambios es la propietaria de los
    archivos .dat
    //Por lo tanto, hacemos un conteo de clientes.dat
    Reader lector = new Reader("clientes.dat");
    Writer escritor = new Writer("output47.dat");
    int contador = 0;

    //Leemos la primera línea
    String linea = lector.readLine();

```

```

//Pasamos a la siguiente
linea = lector.readLine();

escritor.println("CANTIDAD DE CLIENTES: ");

//Extraemos el id del vendedor
while(linea != null){

    contador++;

    linea = lector.readLine();
}

escritor.println(contador);

escritor.close();
}

//48. Quants venedors tenen vendes actuals que superen la seva quota?
public void m48(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output48.dat");
    int contador = 0;
    int ventas;
    int cuota;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

    escritor.println("CANTIDAD DE VENEDORES CON VENTAS MAYORES A SU CUOTA: ");

    //Extraemos el id del vendedor
    while(linea != null){

        String campos[] = linea.split("\\s+");

        if (!campos[8].equals("\\N")){

            cuota = Integer.parseInt(campos[8]);
            ventas = Integer.parseInt(campos[9]);

            if (ventas > cuota){
                contador++;
            }

        }

        linea = lector.readLine();
    }
}

```



```

        escritor.println(contador);

        escritor.close();
    }

    //49. Quantes comandes superen un cert valor (per exemple 25.000 €).
    public void m49(){

        Reader lector = new Reader("pedidos.dat");
        Writer escritor = new Writer("output49.dat");
        int contador = 0;
        int importe;
        int valor = 25000;

        //Leemos la primera linea
        String linea = lector.readLine();
        //Pasamos a la siguiente
        linea = lector.readLine();

        escritor.println("CANTIDAD DE PEDIDOS QUE SUPERAN LOS " + valor + "€");

        //Extraemos el id del vendedor
        while(linea != null){

            String campos[] = linea.split("\\s+");

            importe = Integer.parseInt(campos[7]);

            if (importe > valor){
                contador++;
            }

            linea = lector.readLine();
        }

        escritor.println(contador);

        escritor.close();
    }

    //50. Quantes categories tenen els empleats de l'empresa?
    public void m50(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output50.dat");
        int contador = 0;
        String categoria;
        HashSet <String> categorias = new HashSet<String>();

        //Leemos la primera linea
        String linea = lector.readLine();
        //Pasamos a la siguiente
        linea = lector.readLine();
    }

```

```

//Extraemos el id del vendedor
while(linea != null){

    String campos[] = linea.split("\\s+");

    categoria = campos[5];

    categorias.add(categoria);

    linea = lector.readLine();
}

escritor.println(categorias.size());

escritor.close();
}
}

```

Clase Training06

```

public class Training06{

    //51. Quantes oficines tenen venedors tal que les seves vendes actuals superin les seves
    quotes.

    public void m51(){

        Reader lector = new Reader("oficinas.dat");
        Writer escritor = new Writer("output51.dat");
        int contador = 0;
        int oficina;
        int id;
        int ventas;
        int cuota;

        //Leemos la primera línea
        String linea = lector.readLine();
        //Pasamos a la siguiente
        linea = lector.readLine();

        //Extraemos el id del vendedor
        while(linea != null){

            String campos[] = linea.split("\\s+");

            id = Integer.parseInt(campos[0]);

            //Leemos repventas.dat
            Reader lectorRep = new Reader("repventas.dat");

            //Primera línea y pasamos a la siguiente;

```

```

String lineaRep = lectorRep.readLine();
lineaRep = lectorRep.readLine();

while(lineaRep != null){

    String c[] = lineaRep.split("\\s+");

    if (!c[4].equals("\\N") || !c[8].equals("\\N")){

        oficina = Integer.parseInt(c[4]);
        cuota = Integer.parseInt(c[8]);
        ventas = Integer.parseInt(c[9]);

        if (oficina == id){

            if (ventas > cuota){
                contador++;
            }
        }
    }

    lineaRep = lectorRep.readLine();

}

linea = lector.readLine();
}

escritor.println(contador);

escritor.close();
}

//52. Import mitja de comandes per cada venedor.
public void m52(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output52.dat");
    int contador = 0;
    int suma = 0;
    int importe;
    int id;
    int idVendedor;
    int media;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

    escritor.println("IMPORTE MEDIO DE LOS PEDIDOS DE CADA VENDEDOR: ");

```

```

//Extraemos el id del vendedor
while(linea != null){

    String campos[] = linea.split("\\s+");

    id = Integer.parseInt(campos[0]);
    suma = 0;
    contador = 0;

    //Buscamos los importes
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idVendedor = Integer.parseInt(c[3]);
        importe = Integer.parseInt(c[7]);

        if (idVendedor == id){
            suma = suma + importe;
            contador++;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    if (contador == 0){
        media = 0;
    } else {
        media = suma/contador;
    }

    escritor.println(id + "\t" + media + "€");

    linea = lector.readLine();
}

escritor.close();
}

//53. Rang de quotes (min, max) de cada oficinas.
public void m53(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output53.dat");
    int max;

```

```

int min;
int cuota;

//Leemos la primera línea
String linea = lector.readLine();
//Pasamos a la siguiente
linea = lector.readLine();

//Asignamos el mayor y el menor
String fields[] = linea.split("\\s+");

max = Integer.parseInt(fields[4]);
min = Integer.parseInt(fields[4]);

while(linea != null){

    String campos[] = linea.split("\\s+");

    cuota = Integer.parseInt(campos[4]);

    if (cuota > max){
        max = cuota;
    } else if (cuota < min){
        min = cuota;
    }

    linea = lector.readLine();
}

escritor.println("RANGO DE CUOTAS DE LAS OFICINAS: " + min + "€ - " + max + "€");

escritor.close();

}

//54. Quants venedors estan assignats a cada oficina?
public void m54(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output54.dat");
    int contador = 0;
    int id;
    int oficina;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente
    linea = lector.readLine();

    while(linea != null){

```

```

        String campos[] = linea.split("\\s+");

        contador = 0;
        id = Integer.parseInt(campos[0]);

        //Leemos repventas.dat
        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();

        while(lineaRep != null){

            String c[] = lineaRep.split("\\s+");

            if (!c[4].equals("\\N")){

                oficina = Integer.parseInt(c[4]);

                if (oficina == id){
                    contador++;
                }

            }

            lineaRep = lectorRep.readLine();

        }

        //Podemos sumar 1 en todos los casos para incluir al director de la oficina
        escritor.println("N° de vendedores de la oficina " + id + ": " + contador);

        linea = lector.readLine();
    }

    escritor.close();
}

//55. Quants clients diferents són atesos per cada venedor?
public void m55(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output55.dat");
    int contador = 0;
    int id;
    int idVendedor;

    //Leemos la primera línea
    String linea = lector.readLine();
    //Pasamos a la siguiente

```

```

        linea = lector.readLine();

        //Extraemos el id del vendedor
        while(linea != null){

            String campos[] = linea.split("\\s+");

            contador = 0;
            id = Integer.parseInt(campos[0]);

            //Recorremos clientes.dat
            Reader lectorClientes = new Reader("clientes.dat");

            //Primera línea y pasamos a la siguiente;
            String lineaCliente = lectorClientes.readLine();
            lineaCliente = lectorClientes.readLine();
            boolean salir = false;

            while(lineaCliente != null){

                String c[] = lineaCliente.split("\\s+");

                idVendedor = Integer.parseInt(c[2]);

                if (idVendedor == id){
                    contador++;
                }

                lineaCliente = lectorClientes.readLine();
            }

            escritor.println(id + ": " + contador);

            linea = lector.readLine();
        }

        escritor.close();
    }

    //56. Identificador de vendedor, nom i total import de les comandes de cada vendedor.
    public void m56(){

        Reader lector = new Reader("repventas.dat");
        Writer escritor = new Writer("output56.dat");
        int suma = 0;
        int importe;
        int id;
        int idVendedor;
        String nombre;

        //Leemos la primera línea
        String linea = lector.readLine();
    }

```

```

//Pasamos a la siguiente
linea = lector.readLine();

escritor.println("ID" + "\t" + "NOMBRE" + "\t" + "IMPORTE TOTAL DE VENTAS");

//Extraemos el id del vendedor
while(linea != null){

    String campos[] = linea.split("\\s+");

    id = Integer.parseInt(campos[0]);
    nombre = campos[1];
    suma = 0;

    //Buscamos los importes
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idVendedor = Integer.parseInt(c[3]);
        importe = Integer.parseInt(c[7]);

        if (idVendedor == id){
            suma = suma + importe;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    escritor.println(id + "\t" + nombre + "\t" + suma + "€");

    linea = lector.readLine();
}

escritor.close();
}

//57. Quin és l'import mitjà de les comandes ateses per cada venedor que superen un
valor en vendes donat (per exemple 30.000 €).
public void m57(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output57.dat");
    int contador = 0;
    int suma = 0;
    int importe;

```



```

int id;
int idVendedor;
int media;
int valor = 30000;

//Leemos la primera línea
String linea = lector.readLine();
//Pasamos a la siguiente
linea = lector.readLine();

//Extraemos el id del vendedor
while(linea != null){

    String campos[] = linea.split("\\s+");

    id = Integer.parseInt(campos[0]);
    suma = 0;
    contador = 0;

    //Buscamos los importes
    Reader lectorPedidos = new Reader("pedidos.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaPedidos = lectorPedidos.readLine();
    lineaPedidos = lectorPedidos.readLine();

    while(lineaPedidos != null){

        String c[] = lineaPedidos.split("\\s+");

        idVendedor = Integer.parseInt(c[3]);
        importe = Integer.parseInt(c[7]);

        if (idVendedor == id){
            suma = suma + importe;
            contador++;
        }

        lineaPedidos = lectorPedidos.readLine();
    }

    if (contador == 0){
        media = 0;
    } else {
        media = suma/contador;
    }

    if (media > valor){
        escritor.println(id + "\t" + media + "€");
    }
}

```

```

        linea = lector.readLine();
    }

    escritor.close();
}

//58. Ciutat on es troben les oficines on el seus objectius superen a la suma de les
vendes actuals del seus venedors assignats.
public void m58(){

    //Se puede simplemente comparar objetivos con ventas de la oficina
    //Con esta forma calculamos las ventas nosotros
    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output58.dat");
    int oficina;
    int id;
    int suma = 0;
    int ventas;
    String ciudad;
    int objetivo;

    //Leemos la primera línea
    String linea = lector.readLine();
    escritor.println("CIUDAD" + "\t" + "OBJETIVO" + "\t" + "VENTAS DE EMPLEADOS");
    //Pasamos a la siguiente
    linea = lector.readLine();

    //Extraemos el id del vendedor
    while(linea != null){

        String campos[] = linea.split("\\s+");

        id = Integer.parseInt(campos[0]);
        objetivo = Integer.parseInt(campos[4]);
        ciudad = campos[1];
        suma = 0;

        //Leemos repventas.dat
        Reader lectorRep = new Reader("repventas.dat");

        //Primera línea y pasamos a la siguiente;
        String lineaRep = lectorRep.readLine();
        lineaRep = lectorRep.readLine();

        while(lineaRep != null){

            String c[] = lineaRep.split("\\s+");

            if (!c[4].equals("\\N") || !c[8].equals("\\N")){

                oficina = Integer.parseInt(c[4]);
            }
        }
    }
}

```

```

        ventas = Integer.parseInt(c[9]);

        if (oficina == id){

            suma = suma + ventas;

        }

    }

    lineaRep = lectorRep.readLine();

}

if (objetivo > suma){

    escritor.println(ciudad + "\t" + objetivo + "\t" + suma);

}

linea = lector.readLine();

}

escritor.close();

}

//59. Nom dels venedors que han venut per un import superior a l'objectiu d'una oficina
(per exemple la d'Atlanta).
public void m59(){

    Reader lector = new Reader("oficinas.dat");
    Writer escritor = new Writer("output59.dat");
    int objetivo = 0;
    int ventas;
    String ciudad = "Atlanta";
    String nombre;

    //Leemos la primera línea
    String linea = lector.readLine();
    escritor.println("NOMBRE" + "\t" + "VENTAS" + "\t" + "OBJETIVO DE OFICINA EN " +
ciudad);

    //Pasamos a la siguiente
    linea = lector.readLine();

    //Extraemos el objetivo de la ciudad
    while(linea != null){

        String campos[] = linea.split("\\s+");

        objetivo = Integer.parseInt(campos[4]);

        linea = lector.readLine();

        //Condición de salida
        if (campos[1].equals(ciudad)){

```

```

        linea = null;
    }

}

//Leemos repventas.dat
Reader lectorRep = new Reader("repventas.dat");

//Primera linea y pasamos a la siguiente;
String lineaRep = lectorRep.readLine();
lineaRep = lectorRep.readLine();

while(lineaRep != null){

    String c[] = lineaRep.split("\\s+");

    ventas = Integer.parseInt(c[9]);
    nombre = c[1];

    if (ventas > objetivo){

        escritor.println(nombre + "\t" + ventas + "\t" + objetivo);
    }

    lineaRep = lectorRep.readLine();
}

escritor.close();
}

//60. Nom de les empreses ateses per un venedor (per exemple Bill Adams)
public void m60(){

    Reader lector = new Reader("repventas.dat");
    Writer escritor = new Writer("output60.dat");
    String nombreEmpresa;
    int idVendedor;
    int id = 0;
    String nombreVendedor = "Bill Adams";

    //Leemos la primera linea
    String linea = lector.readLine();
    escritor.println("EMPRESAS ATENDIDAS POR " + nombreVendedor + ": ");
    //Pasamos a la siguiente
    linea = lector.readLine();

    //El id del vendedor
    while(linea != null){

        String campos[] = linea.split("\\s+");

        id = Integer.parseInt(campos[0]);
    }
}

```

```

        linea = lector.readLine();

        //Condición de salida
        if (nombreVendedor.equals(campos[1] + " " + campos[2])){
            linea = null;
        }

    }

    //Recorremos clientes.dat
    Reader lectorClientes = new Reader("clientes.dat");

    //Primera línea y pasamos a la siguiente;
    String lineaCliente = lectorClientes.readLine();
    lineaCliente = lectorClientes.readLine();
    boolean salir = false;

    while(lineaCliente != null){

        String c[] = lineaCliente.split("\\s+");

        idVendedor = Integer.parseInt(c[2]);
        nombreEmpresa = c[1];

        if (idVendedor == id){
            escritor.println(nombreEmpresa);
        }

        lineaCliente = lectorClientes.readLine();
    }

    escritor.close();
}
}

```

Clase Program

```

public class Program {

    public static void main(String[] args){

        Training01 t1 = new Training01();
        t1.m1();
        t1.m2();
        t1.m3();
        t1.m4();
        t1.m5();
        t1.m6();
        t1.m7();
        t1.m8();
    }
}

```

```
t1.m9();
t1.m10();

Training02 t2 = new Training02();
t2.m11();
t2.m12();
t2.m13();
t2.m14();
t2.m15();
t2.m16();
t2.m17();
t2.m18();
t2.m19();
t2.m20();

Training03 t3 = new Training03();
t3.m21();
t3.m22();
t3.m23();
t3.m24();
t3.m25();
t3.m26();
t3.m27();
t3.m28();
t3.m29();
t3.m30();

Training04 t4 = new Training04();
t4.m31();
t4.m32();
t4.m33();
t4.m34();
t4.m35();
t4.m36();
t4.m37();
t4.m38();
t4.m39();
t4.m40();

Training05 t5 = new Training05();
t5.m41();
t5.m42();
t5.m43();
t5.m44();
t5.m45();
t5.m46();
t5.m47();
t5.m48();
t5.m49();
t5.m50();

Training06 t6 = new Training06();
t6.m51();
```

```

t6.m52();
t6.m53();
t6.m54();
t6.m55();
t6.m56();
t6.m57();
t6.m58();
t6.m59();
t6.m60();

}
}

```

Resultados

```

output1.dat
1  IDOF    CIUTAT  VENDES
2  22  Denver  186042
3  11  NewYork 692637
4  12  Chicago  735042
5  13  Atlanta  367911
6  21  LosAngeles 835915
7

```

```

output2.dat
1  NOM IDOFI  VENDES  QUOTA
2  Bill    13  367911  350000
3  Mary    11  392725  300000
4  Sue 21  474050  350000
5  Sam 11  299912  275000
6  Bob 12  142594  200000
7  Dan 12  305673  300000
8  Tom \N  75985   \N
9  Larry  21  361865  350000
10 Paul   12  286775  275000
11 Nancy  22  186042  300000
12

```

```

output3.dat
1  NOM VENDES  QUOTA  DESVIACIÓN
2  Bill    367911  350000  17911
3  Mary    392725  300000  92725
4  Sue 474050  350000  124050
5  Sam 299912  275000  24912
6  Bob 142594  200000  -57406
7  Dan 305673  300000  5673
8  Tom 75985   \N  75985
9  Larry  361865  350000  11865
10 Paul   286775  275000  11775
11 Nancy  186042  300000  -113958
12

```

```

output5.dat
1  8256

```

```

output6.dat
1  8895

```

output7.dat

```
1  Importe total del cliente 2112: 47925€
2  Importe total del cliente 2113: 22500€
3  Importe total del cliente 2114: 22100€
4  Importe total del cliente 2117: 31500€
5  Importe total del cliente 2118: 3608€
6  Importe total del cliente 2120: 3750€
7  Importe total del cliente 2124: 3082€
8  Importe total del cliente 2101: 1458€
9  Importe total del cliente 2102: 3978€
10 Importe total del cliente 2103: 35582€
11 Importe total del cliente 2106: 4026€
12 Importe total del cliente 2107: 23132€
13 Importe total del cliente 2108: 7255€
14 Importe total del cliente 2109: 31350€
15 Importe total del cliente 2111: 6445€
16
```

output8.dat

```
1  Mary    1989-10-12
2  Tom 1990-01-13
3  Larry   1989-10-12
4
```

output9.dat

```
1  OFICINA OBJETIVO    VENTAS  VENTAS +10%
2  Denver  300000  186042  204646
3  NewYork 575000  692637  761900
4  Chicago 800000  735042  808546
5  Atlanta 350000  367911  404702
6  LosAngeles 725000  835915  919506
7
```

output10.dat

```
1  CIUDAD  OBJETIVO    VENTAS  % DE VENTAS
2  Denver  300000  186042  -61%
3  NewYork 575000  692637  16%
4  Chicago 800000  735042  -8%
5  Atlanta 350000  367911  4%
6  LosAngeles 725000  835915  13%
7
```

output11.dat

```
1  LosAngeles 725000  835915  Oeste
2
```



```
output12.dat
1 Oficinas de la región Oeste:
2 MEDIA DE OBJETIVOS MEDIA DE VENTAS
3 512500 510978
4
```

```
output13.dat
1 NOM IDOFI DATAINGRES
2 Bill 13 1988-02-12
3 Mary 11 1989-10-12
4 Sue 21 1986-12-10
5 Sam 11 1988-06-14
6 Bob 12 1987-05-19
7 Dan 12 1986-10-20
8 Tom \N 1990-01-13
9 Larry 21 1989-10-12
10 Paul 12 1987-03-01
11 Nancy 22 1988-11-14
12
```

```
output14.dat
1 IDREP QUOTA VENDES
2 107 300000 186042
3
```

```
output15.dat
1 289353
2
```

```
output16.dat
1 NOM DATAINGRES VENDES
2
```

output17.dat

	IDPRODUCTE	VALOR DE STOCK
1		
2	2a45c	16590
3	4100y	68750
4	xk47	13490
5	41672	0
6	779c	16875
7	41003	22149
8	41004	16263
9	41003	1956
10	887p	6000
11	xk48	27202
12	2a44l	54000
13	112	17020
14	887h	12042
15	41089	17550
16	41001	15235
17	775c	7125
18	4100z	70000
19	xk48a	4329
20	41002	12692
21	2a44r	54000
22	773c	27300
23	4100x	925
24	114	3645
25	887x	15200
26	2a44g	4900
27		

output18.dat

	NOM	QUOTA	QUOTA INCREMENTADA 3%
1			
2	Bill	350000	360500
3	Mary	300000	309000
4	Sue	350000	360500
5	Sam	275000	283250
6	Bob	200000	206000
7	Dan	300000	309000
8	Larry	350000	360500
9	Paul	275000	283250
10	Nancy	300000	309000
11			

```
output19.dat
1 Ciudades con ventas inferiores al 80% de su objetivo
2
3 CIUTAT VENDES OBJECTIU
4 Denver 186042 300000
5
```

```
output20.dat
1 Ciudades no dirigidas por el empleado 108
2
3 CIUTAT IDDIR
4 NewYork 106
5 Chicago 104
6 Atlanta 105
7
```

```
output21.dat
1 Bill
2 Mary
3 Sue
4 Sam
5 Dan
6 Larry
7 Paul
8
```

```
output22.dat
1 IDPEDIDO IMPORT
2 112961 31500
3 113069 31350
4
```

```
output23.dat
1 Bill 350000 367911
2 Sam 275000 299912
3 Dan 300000 305673
4 Larry 350000 361865
5 Paul 275000 286775
6
```

```
output24.dat
1 Bill 13
2 Mary 11
3 Sam 11
4 Nancy 22
5
```

```
output25.dat
1 113012 1990-01-11 3745
2 113003 1990-01-25 5625
3
```

output26.dat

```
1 Tom
2
```

output27.dat

```
1 Sam 275000 299912
2 Bob 200000 142594
3 Paul 275000 286775
4 Nancy 300000 186042
5
```

output28.dat

```
1 Bob 200000 142594
2 Nancy 300000 186042
3
```

output29.dat

```
1 Nancy 300000 186042
2
```

output32.dat

```
1 IDEFAB IDEPRO IMPORT
2 rei 2a44l 31500
3 rei 2a44r 45000
4 imm 775c 31350
5
```

output33.dat

	IDPEDIDO		IMPORT	
1	112961	31500	J.P.Sinclair	35000
2	113012	3745	JCPInc.	50000
3	112989	1458	JonesMfg.	65000
4	113051	1420	MidwestSystems	60000
5	112968	3978	FirstCorp.	65000
6	110036	22500	AceInternational	35000
7	113045	45000	Zetacorp	50000
8	112963	3276	AcmeMfg.	50000
9	113013	652	MidwestSystems	60000
10	113058	1480	Holm&Landis	55000
11	112997	652	PeterBrothers	40000
12	112983	702	AcmeMfg.	50000
13	113024	7100	OrionCorp	20000
14	113062	2430	PeterBrothers	40000
15	112979	15000	OrionCorp	20000
16	113027	4104	AcmeMfg.	50000
17	113007	2925	Zetacorp	50000
18	113069	31350	ChenAssociates	25000
19	113034	632	AceInternational	35000
20	112992	760	MidwestSystems	60000
21	112975	2100	JCPInc.	50000
22	113055	150	Holm&Landis	55000
23	113048	3750	RicoEnterprises	50000
24	112993	1896	FredLewisCorp.	65000
25	113065	2130	FredLewisCorp.	65000
26	113003	5625	Holm&Landis	55000
27	113049	776	MidwestSystems	60000
28	112987	27500	AcmeMfg.	50000
29	113057	600	JCPInc.	50000
30	113042	22500	Ian&Schmidt	20000
31				
32				

output34.dat

1	Bill	Atlanta	13
2	Mary	NewYork	11
3	Sue	LosAngeles	21
4	Sam	NewYork	11
5	Bob	Chicago	12
6	Dan	Chicago	12
7	Larry	LosAngeles	21
8	Paul	Chicago	12
9	Nancy	Denver	22
10			

output35.dat

```
1 CIUDAD DIRECTOR Cargo
2 Denver Larry DirVentas
3 NewYork Sam VPVentas
4 Chicago Bob DirVentas
5 Atlanta Bill RepVentas
6 LosAngeles Larry DirVentas
7
```

output36.dat

```
1 CIUDAD DIRECTOR Cargo
2 Chicago Bob DirVentas
3 LosAngeles Larry DirVentas
4
```


output37.dat

	Nº PEDIDO	IMPORTE	DESCRIPCION
1	112961	31500€	BisagraIzqda
2	113012	3745€	ArticuloTipo3
3	112989	1458€	BancadaMotor
4	113051	1420€	Reductor
5	112968	3978€	ArticuloTipo4
6	110036	22500€	Montador
7	113045	45000€	BisagraDcha
8	112963	3276€	ArticuloTipo4
9	113013	652€	ArticuloTipo3
10	113058	1480€	Cubierta
11	112997	652€	ArticuloTipo3
12	112983	702€	ArticuloTipo4
13	113024	7100€	Reductor
14	113062	2430€	BancadaMotor
15	112979	15000€	Montador
16	113027	4104€	ArticuloTipo2
17	113007	2925€	Riostra1/2-Tm
18	113069	31350€	Riostra1-Tm
19	113034	632€	VStagoTrinquete
20	112992	760€	ArticuloTipo2
21	112975	2100€	PasadorBisagra
22	113055	150€	Ajustador
23	113048	3750€	Riostra2-Tm
24	112993	1896€	VStagoTrinquete
25	113065	2130€	Reductor
26	113003	5625€	Riostra2-Tm
27	113049	776€	Reductor
28	112987	27500€	Extractor
29	113057	600€	Ajustador
30	113042	22500€	BisagraDcha
31			
32			

output38.dat

	Nº PEDIDO	IMPORTE	EMPRESA	VENDEDOR
1	112961	31500€	J.P.Sinclair	Sam
2	113045	45000€	Zetacorp	Larry
3	113069	31350€	ChenAssociates	Nancy
4	112987	27500€	AcmeMfg.	Bill
5				
6				

```

output39.dat
1  Nº PEDIDO    IMPORTE EMPRESA VENDEDOR    CIUDAD DE LA OFICINA
2  112961  31500€  J.P.Sinclair    Sam NewYork
3  113045  45000€  Zetacorp    Larry    LosAngeles
4  113069  31350€  ChenAssociates Nancy    Denver
5  112987  27500€  AcmeMfg.    Bill    Atlanta
6

```

```

output40.dat
1  Nº PEDIDO    IMPORTE FECHA    VENDEDOR
2  112968  3978€    1989-10-12    Dan
3  112979  15000€    1989-10-12    Sue
4

```

```

output41.dat
1  TOTAL DE CUOTAS TOTAL DE VENTAS
2  2750000 2817547
3

```

```

output42.dat
1  IMPORTE TOTAL DE LOS PEDIDOS DE Bill Adams
2  39327
3

```

```

output43.dat
1  PRECIO MEEDIO DE LOS PRODUCTOS DEL FABRICANTE    aci
2  804€
3

```

```

output44.dat
1  IMPORTE MEDIO DE LOS PEDIDOS DE AcmeMfg.
2  8895€
3

```

```

output45.dat
1  TOTAL DE CUOTAS DE LOS VENEDORES
2  CUOTA MÁXIMA: 350000
3  CUOTA MÍNIMA: 200000
4

```

```

output46.dat
1  105 39327€
2  109 7105€
3  102 22776€
4  106 32958€
5  104 0€
6  101 26628€
7  110 23132€
8  108 58633€
9  103 2700€
10 107 34432€
11

```


output47.dat

```
1 CANTIDAD DE CLIENTES:  
2 21  
3
```

output48.dat

```
1 CANTIDAD DE VENDEDORES CON VENTAS MAYORES A SU CUOTA:  
2 7  
3
```

output49.dat

```
1 CANTIDAD DE PEDIDOS QUE SUPERAN LOS 25000€  
2 4  
3
```

output50.dat

```
1 3  
2
```

output51.dat

```
1 7  
2
```

output52.dat

```
1 IMPORTE MEDIO DE LOS PEDIDOS DE CADA VENDEDOR:  
2 105 7865€  
3 109 3552€  
4 102 5694€  
5 106 16479€  
6 104 0€  
7 101 8876€  
8 110 11566€  
9 108 8376€  
10 103 1350€  
11 107 11477€  
12
```

output53.dat

```
1 RANGO DE CUOTAS DE LAS OFICINAS: 300000€ - 800000€  
2
```

output54.dat

```
1  Nº de vendedores de la oficina 22: 1
2  Nº de vendedores de la oficina 11: 2
3  Nº de vendedores de la oficina 12: 3
4  Nº de vendedores de la oficina 13: 1
5  Nº de vendedores de la oficina 21: 2
6
```

output55.dat

```
1  105: 2
2  109: 2
3  102: 4
4  106: 2
5  104: 1
6  101: 3
7  110: 1
8  108: 2
9  103: 3
10 107: 1
11
```

output56.dat

```
1  ID  NOMBRE  IMPORTE TOTAL DE VENTAS
2  105 Bill    39327€
3  109 Mary    7105€
4  102 Sue    22776€
5  106 Sam    32958€
6  104 Bob     0€
7  101 Dan    26628€
8  110 Tom    23132€
9  108 Larry   58633€
10 103 Paul    2700€
11 107 Nancy   34432€
12
```

output57.dat

```
1
```

output58.dat

```
1  CIUDAD  OBJETIVO  VENTAS DE EMPLEADOS
2  Denver  300000  186042
3  Chicago 800000  735042
4
```

output59.dat

```
1  NOMBRE  VENTAS  OBJETIVO DE OFICINA EN Atlanta
2  Bill    367911  350000
3  Mary    392725  350000
4  Sue    474050  350000
5  Larry    361865  350000
6
```

output60.dat

```
1  EMPRESAS ATENDIDAS POR Bill Adams:
2  AcmeMfg.
3  Three-WayLines
4
```

3. CRUD

Clase Crud

```
public class Crud{

    private String masterFile;
    private String createFile;
    private String deleteFile;
    private String updateFile;
    private String fieldDelimiter;

    public Crud(String mF, String cF, String dF, String uF, String fD){

        this.masterFile = mF;
        this.createFile = cF;
        this.deleteFile = dF;
        this.updateFile = uF;
        this.fieldDelimiter = fD;
    }

    public void create(){

        Reader lectorCreate = new Reader("create.dat");
        Reader lector = new Reader(this.masterFile);
        Writer escritor = new Writer("temporalFile.dat");
        String id;

        String line = lectorCreate.readLine();//Encabezado
        //Pasamos a la siguiente línea
        line = lectorCreate.readLine();

        String lineal = lector.readLine();
        String linea2 = lineal;

        //Escribimos el encabezado
        escritor.println(lineal);

        //Seguimos con las siguientes líneas
        lineal = lector.readLine();
        linea2 = lector.readLine();

        //Escribimos la primera línea
        escritor.println(lineal);

        while (line != null){

            String fields [] = line.split(":");
            id = fields[0];
```

```

        boolean salir = false;
        boolean insertado = false;

        while(salir == false){

            String campos1[] = linea1.split(":");
            String campos2[] = linea2.split(":");

            if (id.compareTo(campos1[0]) > 0 && id.compareTo(campos2[0]) < 0){
                //System.out.println(id + " va entre " + campos1[0] + " y " +
campos2[0]);

                escritor.println(line);

                insertado = true;

            }

            //Escribimos el resto de datos
            escritor.println(linea2);

            linea1 = linea2;
            linea2 = lector.readLine();

            if (insertado || linea2 == null){
                salir = true;
                //Si llega al final
                if (id.compareTo(campos2[0]) > 0 && insertado == false){
                    escritor.println(line);

                }

            }

        }

        line = lectorCreate.readLine();
    }

    escritor.close();
    lector.close();
    lectorCreate.close();

    //Eliminamos master.dat
    Files.delete(this.masterFile);

    //Renombramos el archivo temporal
    Files.rename("temporalFile.dat", "master.dat");
    this.masterFile = "master.dat";
}

public void delete(){

```

```

Reader lectorDelete = new Reader("delete.dat");
Reader lector = new Reader(this.masterFile);
Writer escritor = new Writer("temporalFile.dat");
String id;

String line = lectorDelete.readLine();//Encabezado
//Pasamos a la siguiente linea
line = lectorDelete.readLine();

String linea = lector.readLine();

//Escribimos el encabezado
escritor.println(linea);

//Seguimos con las siguientes lineas
linea = lector.readLine();

while (line != null || linea != null){

    id = line;

    boolean salir = false;
    boolean eliminado = false;

    while(salir == false){

        String campos[] = linea.split(":");

        if (campos[0].equals(id)){

            eliminado = true;

        } else {
            escritor.println(linea);
            //System.out.println(linea);
        }

        linea = lector.readLine();

        if (eliminado || linea == null){
            salir = true;
        }

    }

    line = lectorDelete.readLine();
}

escritor.close();
lector.close();
lectorDelete.close();

```

```

        //Eliminamos master.dat
        Files.delete(this.masterFile);

        //Renombramos el archivo temporal
        Files.rename("temporalFile.dat", "master.dat");
        this.masterFile = "master.dat";

    }

    public void update(){

        Reader lectorUpdate = new Reader("update.dat");
        Reader lector = new Reader(this.masterFile);
        Writer escritor = new Writer("temporalFile.dat");
        String id;

        String line = lectorUpdate.readLine(); //Encabezado
        //Pasamos a la siguiente línea
        line = lectorUpdate.readLine();

        String linea = lector.readLine();

        //Escribimos el encabezado
        escritor.println(linea);

        //Seguimos con las siguientes líneas
        linea = lector.readLine();

        while (line != null || linea != null){

            if (line == null){

                id = null;

            } else {

                String fields [] = line.split(":");

                id = fields[0];
            }

            boolean salir = false;
            boolean actualizado = false;

            while(salir == false){

                String campos[] = linea.split(":");

                if (campos[0].equals(id)){

                    escritor.println(line);
                    actualizado = true;

```

```

        } else {
            escritor.println(linea);
            //System.out.println(linea);
        }

        linea = lector.readLine();

        if (actualizado || linea == null){
            salir = true;
        }

    }

    line = lectorUpdate.readLine();
}

escritor.close();
lector.close();
lectorUpdate.close();

//Eliminamos master.dat
Files.delete(this.masterFile);

//Renombramos el archivo temporal
Files.rename("temporalFile.dat", "master.dat");
this.masterFile = "master.dat";

}
}

```

Clase Program

```

public class Program {

    public static void main(String[] args){

        Crud c = new Crud("master.dat", "create.dat", "delete.dat", "update.dat", ":");

        c.create();
        c.delete();
        c.update();

    }

}

```

Resultado:

```
master.dat
1 ID:TIPUS:TITOL:AUTOR:PREU:STOCK
2 DB02:Database:Easy DB:C. J. Wallis:49.95:10
3 DB04:Database:Postgres:P. Ronny:45.99:2
4 DB05:Database:Database Processing:D. Kroenke:136.65:12
5 DB07:Database:Access Database Design:S. Roman:34.95:25
6 DB09:Database:SQL Server 2005:P. Debetta:29.99:0
7 GR01:Graphics:Adobe Photoshop CS2:Adobe:29.99:4
8 GR04:Graphics:Macromedia Flash Professional:T. Green:44.99:17
9 GR07:Graphics:Digital Photographer Handbook:M. Freeman:24.95:22
10 GR10:Graphics:Creating Motion Graphics:T. Meyer:59.95:13
11 HW02:Hardware:How Computers Work:R. White:29.99:8
12 HW04:Hardware:Upgrading and Repairing PCs:S. Mueller:59.99:5
13 HW05:Hardware:USB System Architecture:D. Anderson:49.99:1
14 HW08:Hardware:GPU speeding:R. Weston:45.95:5
15 SW03:Software:Java How to Program:Deitel:98.59:9
16 SW04:Software:C Programming Language:B. Kernighan:44.25:12
17 SW09:Software:Programming PHP:R. J. Lerdorf:39.95:17
18 SW12:Software:Visual Basic.NET Programming:P. Vick:49.99:13
19 SW13:Software:Programming in Haskell:P. Forest:33.99:10
20 SY02:Systems:The UNIX Operating System:J. D. Peek:19.95:12
21 SY03:Systems:Windows Server 2003:W. R. Stanek:29.99:25
22 SY04:Systems:Linux in a Nutshell:S. Figgins:44.95:1
23 SY05:Systems:Mastering Active Directory:R. R. King:49.99:8
24 WB01:Web:Ajax in Action:D. Crane:22.67:14
25 WB02:Web:Professional ASP.NET 2.0:B. Evjen:32.99:21
26 WB05:Web:Cascading Style Sheets:E. A. Meyer:39.95:6
27 WB06:Web:Ajax in a nutshell:J. Pierce:43.99:12
28 WB08:Web:Java EE:D. Romero:39.99:7
29
```