

**Actividades de aplicación.** Realizar las siguientes 10.15, 10.16, 10.17, 10.18, 10.20, 10.22 y 10.23.

**10.15.** En el archivo *numeros.txt* disponemos de una serie de números (uno por cada línea). Diseña un programa que procese el fichero y nos muestre el menor y el mayor.

```
1 package com.mycompany.aal0_15;
2
3 import java.io.FileInputStream;
4 import java.util.Scanner;
5
6
7 public class Aal0_15 {
8
9     public static void main(String[] args) {
10         int numero, menor = 0, mayor = 0;
11         try{
12             FileInputStream flujo = new FileInputStream(name: "numeros.txt");
13             Scanner scanner = new Scanner(source: flujo);
14             while (scanner.hasNextInt()){
15                 numero = scanner.nextInt();
16                 menor = (menor > numero) ? numero : menor;
17                 mayor = (mayor < numero) ? numero : mayor;
18             }
19             System.out.println("El numero menor del fichero es: " + menor);
20             System.out.println("El numero mayor del fichero es: " + mayor);
21             flujo.close();
22         }catch(Exception e){
23             System.out.println(e);
24         }
25     }
26 }
27
```

numeros.txt

Archivo Editar Ver

10  
25  
9  
8  
17  
6  
0  
3

Ln 1, Col 1 18 caracteres.

Output - Run (Aal0\_15) x

```
--- resources:3.3.1:resources (default-resources) @ Aal0_15 ---
skip non existing resourceDirectory C:\Users\abrah\Documents\NetBeansProjects\Aal0_15\src\main\re
--- compiler:3.11.0:compile (default-compile) @ Aal0_15 ---
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes
--- exec:3.1.0:exec (default-cli) @ Aal0_15 ---
El numero menor del fichero es: 0
El numero mayor del fichero es: 25
BUILD SUCCESS
Total time: 1.374 s
Finished at: 2024-02-08T20:00:18Z
```

**10.16.** Un libro de firmas es útil para recoger los nombres de todas las personas que han pasado por un determinado lugar. Crea una aplicación que permita mostrar el libro de firmas e insertar un nuevo nombre (comprobando que no se encuentre repetido). Llamaremos al fichero *firmas.txt*.

```

1  package com.mycompany.aal0_16;
2
3  import java.io.*;
4  import java.util.Scanner;
5
6  public class Aal0_16 {
7      final static String FICHERO = "firmas.txt";
8
9      public static void main(String[] args) {
10         Scanner scanner = new Scanner(System.in);
11         String nombre = "";
12
13         try {
14             System.out.println("Estado actual del libro de firmas:\n"
15                 + leeLibro());
16             System.out.println("Introduza nuevo nombre a introducir");
17             nombre = scanner.nextLine();
18             inserta(nombre);
19             System.out.println("Estado final del libro de firmas:\n"
20                 + leeLibro());
21         } catch (Exception e) {
22             System.out.println(e);
23         }
24     }
25
26     static String leeLibro() throws FileNotFoundException, IOException {
27         BufferedReader in = new BufferedReader(new FileReader(fileName: FICHERO));
28         String firmas = "";
29         String linea = in.readLine();
30         while (linea != null) {
31             firmas += linea + "\n";
32             linea = in.readLine();
33         }
34         in.close();
35         return firmas;
36     }
37
38     static void inserta(String nombre) throws IOException{
39         BufferedWriter out = new BufferedWriter(new FileWriter(fileName: FICHERO, append:true));
40         if(leeLibro().contains(nombre)){
41             System.out.println("Error, el nombre ya existe en el fichero\n");
42         }else{
43             out.newLine();
44             out.write(nombre);
45         }
46         out.close();
47     }
48 }
49

```

```

--- exec:3.1.0:exec (default-cli) @ Aal0_16 ---
Estado actual del libro de firmas:
Abraham
Lucas
Paco
Luis
Antonio

Introduza nuevo nombre a introducir
Perro Sanchez
Estado final del libro de firmas:
Abraham
Lucas
Paco
Luis
Antonio
Perro Sanchez

BUILD SUCCESS

```

```

--- exec:3.1.0:exec (default-cli) @ Aal0_16 ---
Estado actual del libro de firmas:
Abraham
Lucas
Paco
Luis
Antonio
Perro Sanchez

Introduza nuevo nombre a introducir
Luis
Error, el nombre ya existe en el fichero

Estado final del libro de firmas:
Abraham
Lucas
Paco
Luis
Antonio
Perro Sanchez

BUILD SUCCESS

```

**10.17.** En Linux disponemos del comando *more*, al que se le pasa un fichero y lo muestra poco a poco: cada 24 líneas. Implementa un programa que funcione de forma similar.

```
1 package com.mycompany.aal0_17;
2
3 import java.io.BufferedReader;
4 import java.io.FileReader;
5 import java.io.IOException;
6 import java.util.Scanner;
7
8 public class Aal0_17 {
9
10     public static void main(String[] args) {
11         try {
12             final String FICHERO = "texto.txt";
13
14             BufferedReader in = new BufferedReader(new FileReader(fileName: FICHERO));
15             Scanner sc = new Scanner(source: System.in);
16             int contador = 1;
17             String texto = in.readLine();
18             while (texto != null) {
19                 if (contador < 25) {
20                     System.out.println(contador+ ": " + texto);
21                     texto = in.readLine();
22                     contador++;
23                 } else {
24                     contador = 1;
25                     System.out.println("--->Fin del tramo. Introduzca un caracter "
26                         + "\n pulse enter para continuar");
27                     sc.next();
28                 }
29             }
30         } catch (IOException ex) {
31             System.out.println(ex);
32         }
33     }
34 }
35
```

```
--- exec3.1.0-exec (default-cli) @ Aal0_17 ---
1: Una casa (del latín casa, choza) es una edificación destinada para ser habitada.
2: Puede organizarse en una o varias plantas, y normalmente, aunque no exclusivamente,
3: se refiere a un edificio destinado a vivienda unifamiliar.
4: Puede además disponer de un sótano, o un semisótano, y de una cubierta superior
5: transitable denominada azotea o terraza. Si dispone de terreno suficiente,
6: puede contar también con patio y jardín.
7:
8: Es el lugar en que históricamente se desarrollaron las circunstancias y
9: relaciones específicas de la vida social o familiar, desde el nacimiento a la
10: muerte de muchos de sus componentes. Sirve de refugio contra la lluvia, el viento
11: y demás agentes meteorológicos, y protege de posibles intrusos, humanos o animales.
12: Además es el lugar donde almacenar los enseres y propiedades de sus habitantes.
13:
14: Asimismo se llama casa a algunos edificios destinados a servicios comunitarios,
15: como la casa consistorial, la casa de beneficencia, la casa de empeños, la casa de maternidad, la casa de socorro; a sedes institucionales principales, como la Casa Blanca,
16: la Casa Azul, la Casa Rosada, o incluso a las de mala reputación, como las casas de camas,
17: casas de citas, etc.
18:
19: En contraposición a la «casas», se suele denominar «pisos»? («apartamento» o «departamentos»)
20: a la vivienda unifamiliar que forma parte de una edificación mayor, normalmente de varias
21: alturas.
22:
23: La unidad social que vive en una casa se conoce como hogar. Por lo general, un hogar es una
24: unidad familiar de algún tipo, aunque los hogares también pueden ser otros grupos sociales,
--->Fin del tramo. Introduzca un caracter y pulse enter para continuar
a
1: como compañeros de habitación o, en una pensión, personas no relacionadas. Algunas casas
2: solo tienen un espacio de vivienda para una familia o grupo de tamaño similar; las casas
3: más grandes llamadas casas adosadas pueden contener numerosas viviendas familiares en la
4: misma estructura. Una casa puede ir acompañada de dependencias, como un garaje para
5: vehículos o un cobertizo para equipos y herramientas de jardinería. Una casa puede tener un
6: patio trasero o un patio delantero o ambos, que sirven como áreas adicionales donde los
7: habitantes pueden relajarse o comer.
8:
9: Una casa típica de la burguesía, en Inglaterra.
10:
11: Casa de piedra de arquitectura inca (siglo xv), Ollantaytambo, Perú.
12:
13: Una casa tradicional de Siberia, en Novosibirsk, Rusia.
14:
15: Casas colgadas (siglo xv), Cuenca, España.
16: Etimología
17:
18: Hus, an Old English word
19: Casa es un vocablo que ya existía en latín, aunque con el significado de «choza» o
20: «cabaña», mientras que para denotar «casa», en la Roma Antigua usaban el término: domus.2?
21:
22:
23: La propia casa dio lugar a la letra 'B' a través de un temprano jeroglífico proto-semitica
24: que representaba, una casa. El símbolo fue llamado "bayt", "bet" o "beth" en varias lenguas
--->Fin del tramo. Introduzca un caracter y pulse enter para continuar
a
```

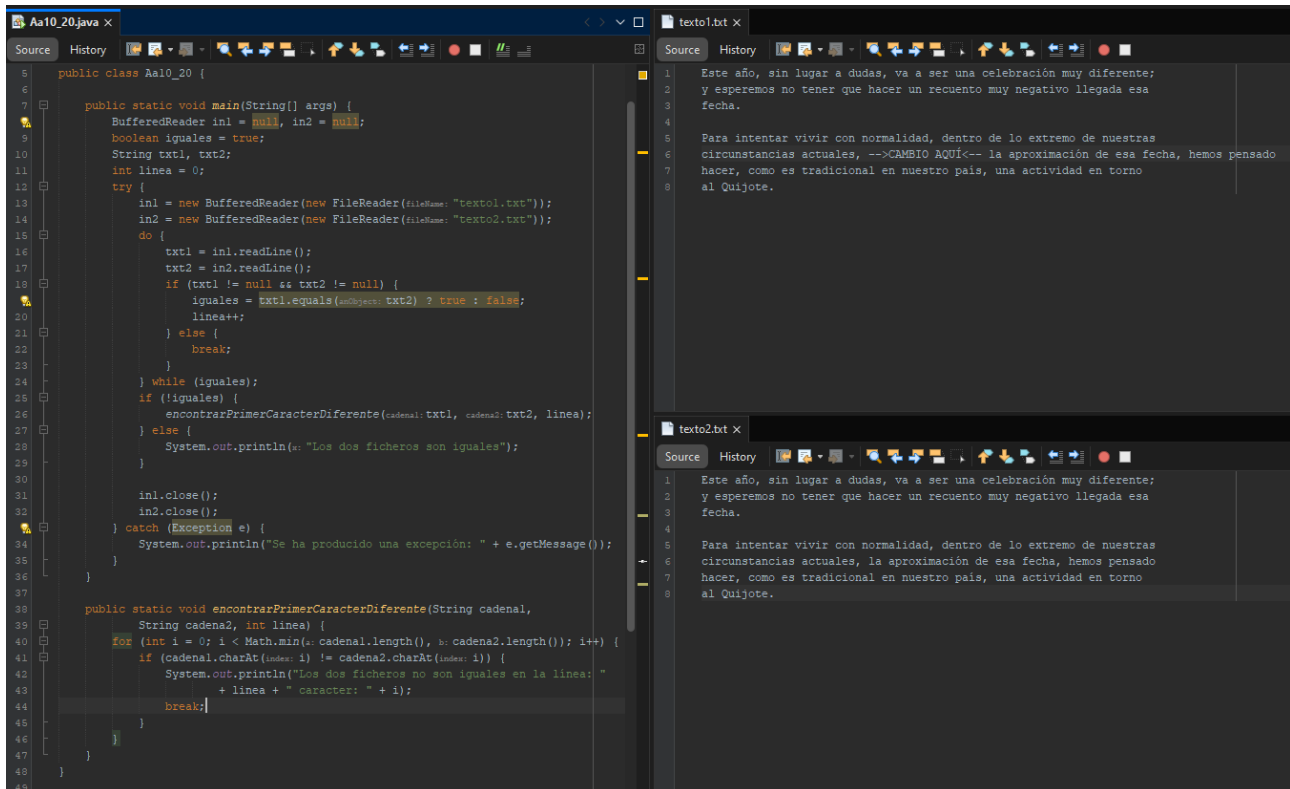
10.18. Escribe la función `Integer[] leerEnteros(String texto)`, al que se le pasa una cadena y devuelve una tabla con todos los enteros que aparecen en ella.

```
1 package com.mycompany.aal0_18;
2
3 import java.util.Arrays;
4 import java.util.Scanner;
5
6 public class Aal0_18 {
7
8     public static void main(String[] args) {
9         String cadena = "asd 15 asdlkas 24 aiowhqe 45 as,dmqoih 9 ñamsdk";
10        System.out.println( "x: Arrays.toString(x: leerEnteros(cadena))");
11    }
12
13    static Integer[] leerEnteros(String cadena){
14        Scanner sc = new Scanner(source:cadena);
15        Integer[] resultado = {};
16        int contador = 0;
17        while(sc.hasNext()){
18            if(sc.hasNextInt()){
19                Integer[] aux = new Integer[resultado.length + 1];
20                System.arraycopy(src:resultado, srcPos:0, dest:aux, destPos:0, length:resultado.length );
21                resultado = aux;
22                resultado[contador] = sc.nextInt();
23                contador ++;
24            }else{
25                sc.next();
26            }
27        }
28        sc.close();
29        return resultado;
30    }
31 }
32
```

Output X

```
Debugger Console X Run (Aal0_18) X
--- compiler:3.11.0:compile (default-compile) @ Aal0_18 ---
Changes detected - recompiling the module! :source
Compiling 1 source file with javac [debug target 21] to target\classes
--- exec:3.1.0:exec (default-cli) @ Aal0_18 ---
[15, 24, 45, 9]
-----
BUILD SUCCESS
-----
```

**10.20.** Algunos sistemas operativos disponen de la orden `comp`, que compara dos archivos y nos dice si son iguales o distintos. Diseña esta orden de forma que, además, nos diga en qué línea y carácter se encuentra la primera diferencia. Utiliza los ficheros `texto1.txt` y `texto2.txt`.



```
public class Aa10_20 {  
    public static void main(String[] args) {  
        BufferedReader in1 = null, in2 = null;  
        boolean iguales = true;  
        String txt1, txt2;  
        int linea = 0;  
        try {  
            in1 = new BufferedReader(new FileReader(fileName: "texto1.txt"));  
            in2 = new BufferedReader(new FileReader(fileName: "texto2.txt"));  
            do {  
                txt1 = in1.readLine();  
                txt2 = in2.readLine();  
                if (txt1 != null && txt2 != null) {  
                    iguales = txt1.equals(txt2) ? true : false;  
                    linea++;  
                } else {  
                    break;  
                }  
            } while (iguales);  
            if (!iguales) {  
                encontrarPrimerCaracterDiferente(cadena1: txt1, cadena2: txt2, linea);  
            } else {  
                System.out.println("Los dos ficheros son iguales");  
            }  
            in1.close();  
            in2.close();  
        } catch (Exception e) {  
            System.out.println("Se ha producido una excepción: " + e.getMessage());  
        }  
    }  
    public static void encontrarPrimerCaracterDiferente(String cadena1,  
        String cadena2, int linea) {  
        for (int i = 0; i < Math.min(a. cadena1.length(), b. cadena2.length()); i++) {  
            if (cadena1.charAt(index: i) != cadena2.charAt(index: i)) {  
                System.out.println("Los dos ficheros no son iguales en la línea: "  
                    + linea + " carácter: " + i);  
                break;  
            }  
        }  
    }  
}
```

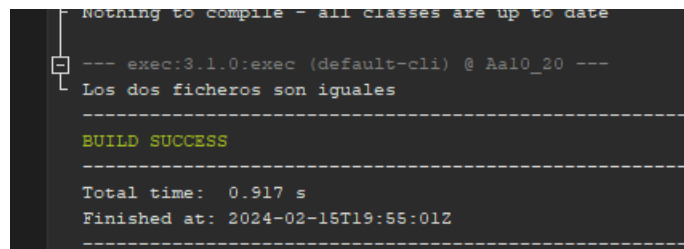
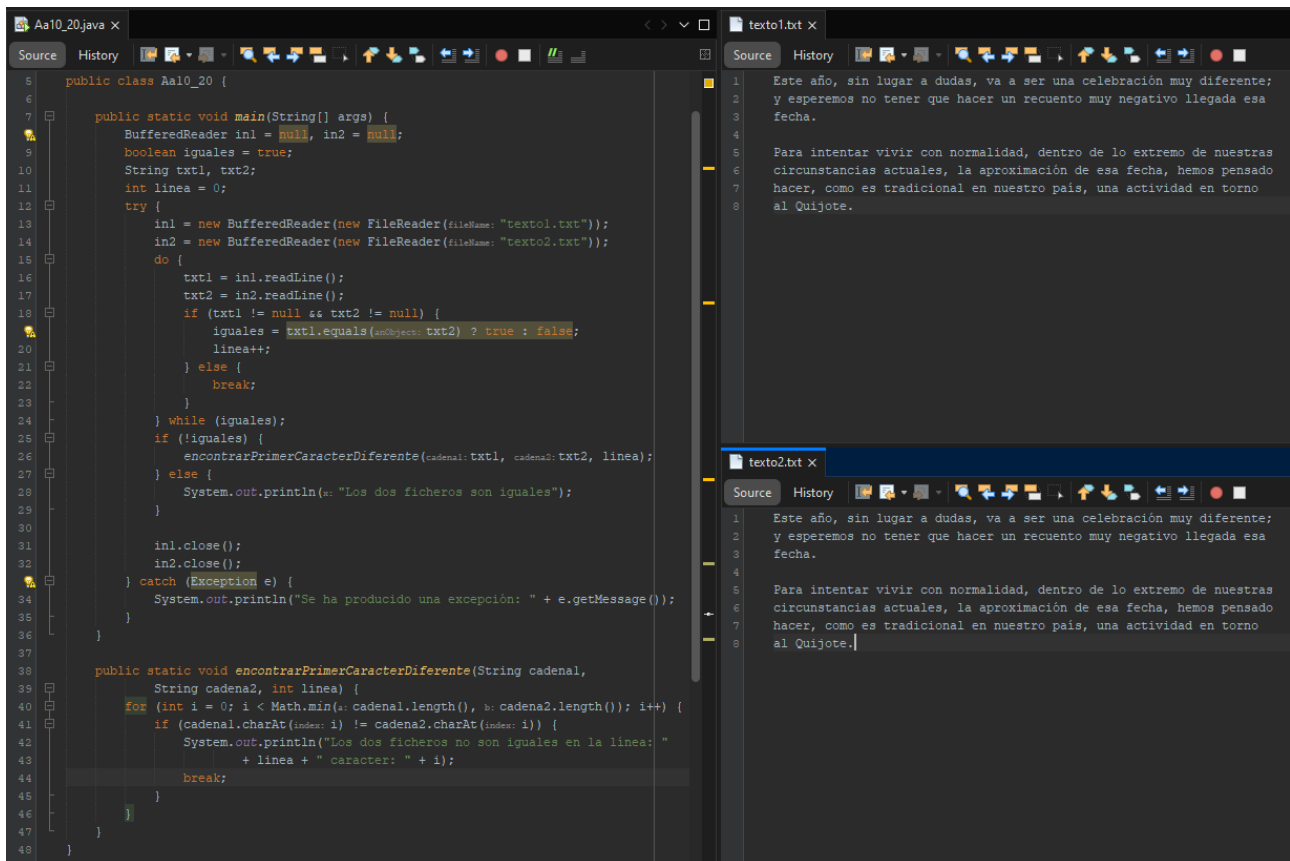
texto1.txt

```
1 Este año, sin lugar a dudas, va a ser una celebración muy diferente:  
2 y esperemos no tener que hacer un recuento muy negativo llegada esa  
3 fecha.  
4  
5 Para intentar vivir con normalidad, dentro de lo extremo de nuestras  
6 circunstancias actuales, --CAMBIO AQUÍ-- la aproximación de esa fecha, hemos pensado  
7 hacer, como es tradicional en nuestro país, una actividad en torno  
8 al Quijote.
```

texto2.txt

```
1 Este año, sin lugar a dudas, va a ser una celebración muy diferente:  
2 y esperemos no tener que hacer un recuento muy negativo llegada esa  
3 fecha.  
4  
5 Para intentar vivir con normalidad, dentro de lo extremo de nuestras  
6 circunstancias actuales, la aproximación de esa fecha, hemos pensado  
7 hacer, como es tradicional en nuestro país, una actividad en torno  
8 al Quijote.
```

```
--- exec:3.1.0:exec (default-cli) @ Aa10_20 ---  
Los dos ficheros no son iguales en la línea: 6 carácter: 25  
-----  
BUILD SUCCESS  
-----  
Total time: 1.482 s  
Finished at: 2024-02-15T19:54:04Z  
-----
```



**10.22.** Crea con un editor de texto el fichero *deportistas.txt*, donde se recogen los datos de un grupo de deportistas, uno en cada línea. Aparecerá el nombre completo, seguido de la edad, el peso y la estatura. La primera línea será el encabezamiento con los nombres de los campos. El documento tendrá la siguiente forma:

Nombre	Edad	Peso	Estatura
Juan Pedro Pérez Gómez	25	70,5	1,80
Ana Ruiz del Val	23	60	1,75

...

Implementa un programa donde se cree un flujo de texto de entrada, a partir del cual, usando un objeto *Scanner*, se leerán los datos de los deportistas, que se mostrarán por pantalla. Al final aparecerán los valores medios de la edad, el peso y la estatura.

The screenshot shows an IDE with two files open. The left pane shows the Java code for `Aa10_22.java`, and the right pane shows the content of `deportistas.txt`.

```
public class Aa10_22 {  
    public static void main(String[] args) {  
        int cant = 0, edadTotal = 0;  
        double pesoTotal = 0, estaturaTotal = 0;  
        BufferedReader in = null;  
        String linea = null;  
        Scanner sc;  
        try {  
            in = new BufferedReader(new FileReader("deportistas.txt"));  
            linea = in.readLine();  
            System.out.println(":" + linea);  
            while (linea != null) {  
                linea = in.readLine();  
                if (linea != null) {  
                    System.out.println(":" + linea);  
                    sc = new Scanner(linea);  
                    cant++;  
                    while (!sc.hasNextInt()) {  
                        sc.next();  
                    }  
                    while (sc.hasNext()) {  
                        edadTotal += sc.nextInt();  
                        pesoTotal += sc.nextDouble();  
                        estaturaTotal += sc.nextDouble();  
                    }  
                    sc.close();  
                }  
            }  
            in.close();  
            System.out.println("Media de edad: " + edadTotal / cant);  
            System.out.println("Media de peso: " +  
                Math.round(pesoTotal / cant * 100) / 100);  
            System.out.println("Media de estatura: " +  
                Math.round(estaturaTotal / cant * 100) / 100);  
        } catch (Exception e) {  
            System.out.println("Se ha producido una excepción: " + e.getMessage());  
        }  
    }  
}
```

The `deportistas.txt` file contains the following text:

```
Nombre          Edad    Peso    Estatura  
Juan Pedro Pérez Gómez  25      70,5    1,80  
Ana Ruiz del Val      23      60      1,75  
Abraham Quintana Micó 36      82      1,73
```

```
--- exec:3.1.0:exec (default-cli) @ Aa10_22 ---  
Nombre          Edad    Peso    Estatura  
Juan Pedro Pérez Gómez  25      70,5    1,80  
Ana Ruiz del Val      23      60      1,75  
Abraham Quintana Micó 36      82      1,73  
Media de edad: 28  
Media de peso: 70  
Media de estatura: 1  
-----  
BUILD SUCCESS  
-----  
Total time: 1.509 s  
Finished at: 2024-02-15T20:44:10Z
```

**10.23.** Con el fichero `deportistas.txt` de la Actividad de aplicación 10.22, implementa una aplicación que lea los datos de los deportistas y los guarde en otros tres ficheros, uno con los nombres y las edades, otro con los nombres y los pesos y el tercero con los nombres y las estaturas.

The screenshot shows an IDE with a Java file named `Aa10_23.java` and three output files: `deportistas.txt`, `Edades.txt`, `Pesos.txt`, and `Estaturas.txt`.

**deportistas.txt**

Nombre	Edad	Peso	Estatura
Juan Pedro Pérez Gómez	25	70,5	1,80
Ana Ruiz del Val	23	60	1,75
Abraham Quintana Micó	36	82	1,73

**Edades.txt**

Nombre	Edad
Juan Pedro Pérez Gómez	25
Ana Ruiz del Val	23
Abraham Quintana Micó	36

**Pesos.txt**

Nombre	Peso
Juan Pedro Pérez Gómez	70.5
Ana Ruiz del Val	60.0
Abraham Quintana Micó	82.0

**Estaturas.txt**

Nombre	Estatura
Juan Pedro Pérez Gómez	1.8
Ana Ruiz del Val	1.75
Abraham Quintana Micó	1.73

**Source Code (Aa10\_23.java)**

```
1 package com.mycompany.aal0_23;
2
3 import java.io.*;
4 import java.util.Scanner;
5
6 public class Aa10_23 {
7
8     public static void main(String[] args) {
9         int edad = 0;
10        double peso = 0, estatura = 0;
11        BufferedReader in;
12        BufferedWriter outEdad, outPeso, outEstatura;
13        String linea, nombre;
14        Scanner sc;
15        final String cabeceraEdad = "Nombre          Edad";
16        final String cabeceraPeso = "Nombre          Peso";
17        final String cabeceraEstatura = "Nombre          Estatura";
18        try {
19            in = new BufferedReader(new FileReader("deportistas.txt"));
20            outEdad = new BufferedWriter(new FileWriter("Edades.txt"));
21            outPeso = new BufferedWriter(new FileWriter("Pesos.txt"));
22            outEstatura = new BufferedWriter(new FileWriter("Estaturas.txt"));
23            outEdad.write((String) cabeceraEdad);
24            outPeso.write((String) cabeceraPeso);
25            outEstatura.write((String) cabeceraEstatura);
26            linea = in.readLine();
27            while (linea != null) {
28                linea = in.readLine();
29                if (linea != null) {
30                    nombre = "";
31                    sc = new Scanner((String) linea);
32                    while (!sc.hasNextInt()) {
33                        nombre += " " + sc.next();
34                    }
35                    while (sc.hasNext()) {
36                        edad = sc.nextInt();
37                        peso = sc.nextDouble();
38                        estatura = sc.nextDouble();
39                    }
40                    sc.close();
41                    outEdad.newLine();
42                    outEdad.append(nombre + "          " + edad);
43                    outPeso.newLine();
44                    outPeso.append(nombre + "          " + peso);
45                    outEstatura.newLine();
46                    outEstatura.append(nombre + "          " + estatura);
47                }
48            }
49        } catch (IOException e) {
50            e.printStackTrace();
51        }
52    }
53 }
```

```
--- exec:3.1.0:exec (default-cli) @ Aa10_23 ---
BUILD SUCCESS
Total time: 1.439 s
Finished at: 2024-02-15T21:13:09Z
```



**Actividades propuestas.** Realizar las siguientes 10.02, 10.03 y 10.04.

### Actividad propuesta 10.2

Fide por teclado el nombre, la edad (int) y la estatura en metros (double) de un deportista. Introduce los datos en una sola línea y léelos con un objeto `Scanner`. Muestra los resultados por pantalla.

```
1  package com.mycompany.ap10_2;
2
3  import java.util.Scanner;
4
5  public class Apl0_2 {
6      public static void main(String[] args) {
7          Scanner sc;
8          String linea;
9          try {
10             sc = new Scanner(source: System.in);
11             System.out.println(x: "Introduzca nombre: ");
12             linea = sc.nextLine();
13             System.out.println(x: "Introduzca edad: ");
14             linea += " " + sc.nextInt();
15             System.out.println(x: "Introduzca estatura en metros: ");
16             linea += " " + sc.nextDouble();
17             sc.close();
18             sc = new Scanner(source: linea);
19             System.out.println(x: "Nombre: ");
20             while (!sc.hasNextInt()) {
21                 System.out.print( sc.next() + " ");
22             }
23             System.out.print(s: "\nEdad: ");
24             System.out.println(x: sc.next());
25             System.out.print(s: "Estatura: ");
26             System.out.println( x: sc.next());
27             sc.close();
28         } catch (Exception e) {
29             System.out.println("Se ha producido una excepción: " + e);
30         }
31     }
32 }
```

Output ×

Debugger Console ×

Run (Ap10\_2) ×

--- exec:3.1.0:exec (default-cli) @ Apl0\_2 ---

Introduzca nombre:

Abraham Quintana

Introduzca edad:

36

Introduzca estatura en metros:

1,73

Nombre:

Abraham Quintana

Edad: 36

Estatura: 1.73

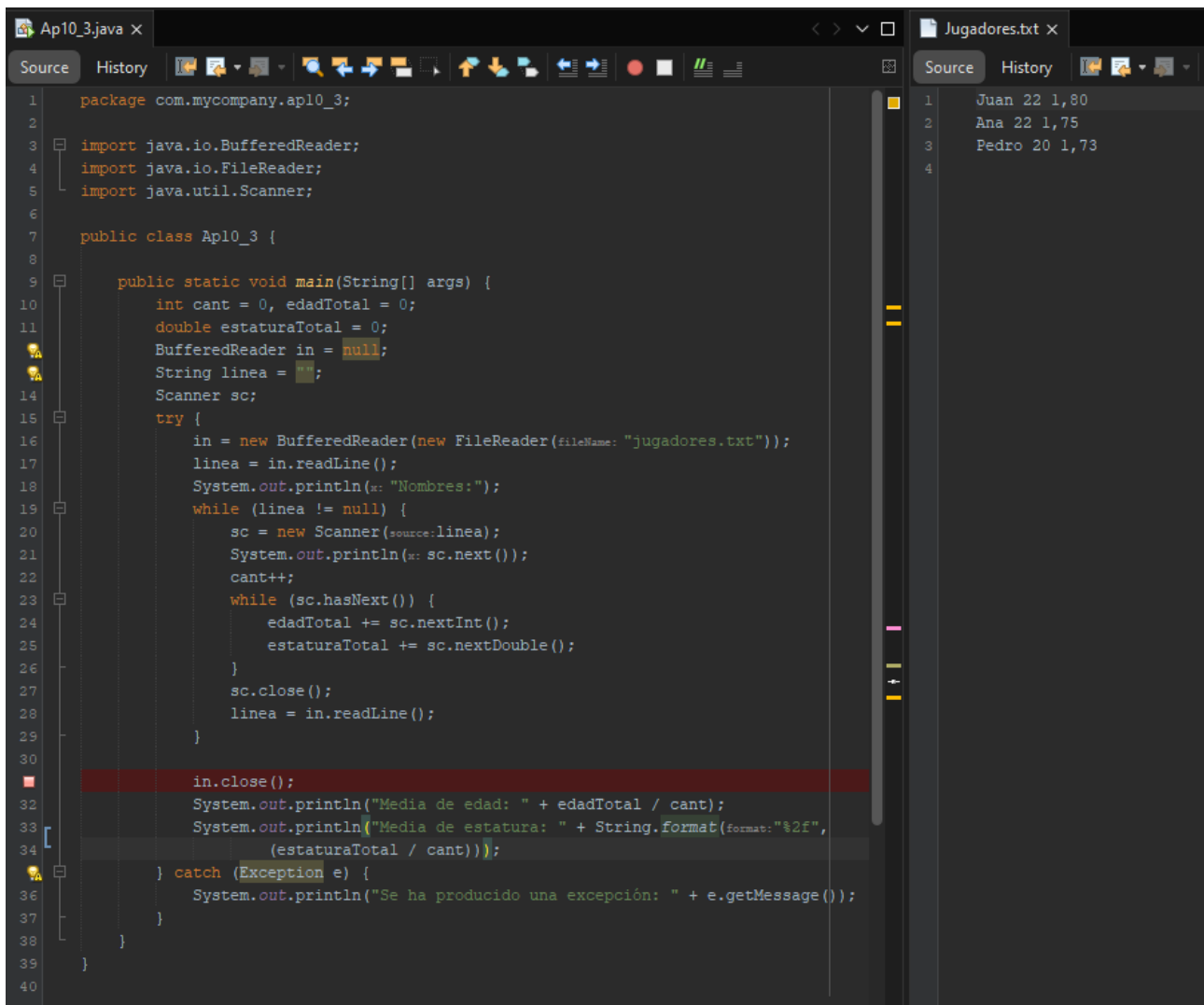
-----  
BUILD SUCCESS  
-----

### Actividad propuesta 10.3

Crea con un editor el fichero de texto *Jugadores.txt* en la carpeta del proyecto de NetBeans actual y escribe en él los nombres, edades y estaturas de los jugadores de un equipo, cada uno en una línea.

```
juan 22 1.77
luis 22 1.80
pedro 20 1.73
...
```

Implementa un programa que lea del fichero los datos, muestre los nombres y calcule la media de la edad y de las estaturas, mostrándolas por pantalla.



The screenshot shows the NetBeans IDE with two tabs: `Ap10_3.java` and `Jugadores.txt`. The `Ap10_3.java` tab is active, displaying the following Java code:

```
1 package com.mycompany.ap10_3;
2
3 import java.io.BufferedReader;
4 import java.io.FileReader;
5 import java.util.Scanner;
6
7 public class Ap10_3 {
8
9     public static void main(String[] args) {
10         int cant = 0, edadTotal = 0;
11         double estaturaTotal = 0;
12         BufferedReader in = null;
13         String linea = "";
14         Scanner sc;
15         try {
16             in = new BufferedReader(new FileReader(fileName: "jugadores.txt"));
17             linea = in.readLine();
18             System.out.println("Nombres:");
19             while (linea != null) {
20                 sc = new Scanner(source: linea);
21                 System.out.println(sc.next());
22                 cant++;
23                 while (sc.hasNext()) {
24                     edadTotal += sc.nextInt();
25                     estaturaTotal += sc.nextDouble();
26                 }
27                 sc.close();
28                 linea = in.readLine();
29             }
30             in.close();
31             System.out.println("Media de edad: " + edadTotal / cant);
32             System.out.println("Media de estatura: " + String.format(format: "%2f",
33                 (estaturaTotal / cant)));
34         } catch (Exception e) {
35             System.out.println("Se ha producido una excepción: " + e.getMessage());
36         }
37     }
38 }
39
40
```

The `Jugadores.txt` tab is also visible, showing the following content:

```
1 Juan 22 1,80
2 Ana 22 1,75
3 Pedro 20 1,73
4
```

```
--- exec:3.1.0:exec (default-cli) @ Ap10_3 ---
Nombres:
Juan
Ana
Pedro
Media de edad: 21
Media de estatura: 1,760000
-----
BUILD SUCCESS
-----
Total time: 1.547 s
Finished at: 2024-02-15T21:58:00Z
-----
```

## Actividad propuesta 10.4

Escribe un texto en un archivo de texto, línea a línea leídas del teclado, hasta que introduzca la cadena «fin».

The screenshot shows an IDE with two main windows. The left window, titled 'Ap10\_4.java', contains the following Java code:

```
2
3 import java.io.*;
4 import java.util.Scanner;
5
6 public class Ap10_4 {
7
8     public static void main(String[] args) {
9         File file = new File(pathname: "texto.txt");
10        final String FIN = "fin";
11        String linea;
12        boolean seguir;
13        BufferedWriter flujo;
14        Scanner sc;
15        try {
16            if (!file.exists()){
17                file.createNewFile();
18            }
19            flujo = new BufferedWriter(new FileWriter(fileName: file.getPath(), append:true));
20            sc = new Scanner(source: System.in);
21            do{
22                System.out.println("Introduzca texto a añadir, "
23                                + "para terminar infroduzca fin:");
24                linea = sc.nextLine();
25                seguir = !linea.equals(anObject: FIN);
26                if(seguir){
27                    flujo.write( str: linea);
28                    flujo.newLine();
29                }
30            }while(seguir);
31            sc.close();
32            flujo.close();
33        } catch (IOException ex) {
34            System.out.println(=: ex.getMessage());
35        }
36    }
37 }
```

The right window, titled 'texto.txt', shows the content of the file:

```
1  Hola esta es la 1ª linea
2  luego meto la segunda
3  y la tercera es la ultima antes de poner fin
4
```

At the bottom, the 'Output - Run (Ap10\_4)' window shows the program's execution:

```
*** exec: java.exe (default-ctx) @ Ap10_4 ***
Introduzca texto a añadir, para terminar infroduzca fin:
Hola esta es la 1ª linea
Introduzca texto a añadir, para terminar infroduzca fin:
luego meto la segunda
Introduzca texto a añadir, para terminar infroduzca fin:
y la tercera es la ultima antes de poner fin
Introduzca texto a añadir, para terminar infroduzca fin:
fin
-----
BUILD SUCCESS
-----
```