

Navegación por el cuestionario

|   |   |   |   |   |   |   |   |   |    |
|---|---|---|---|---|---|---|---|---|----|
| 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | 10 |
| ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✗  |

Finalizar revisión

|                 |                                       |
|-----------------|---------------------------------------|
| Comenzado el    | sábado, 8 de diciembre de 2018, 12:47 |
| Estado          | Finalizado                            |
| Finalizado en   | lunes, 10 de diciembre de 2018, 16:25 |
| Tiempo empleado | 2 días 3 horas                        |
| Calificación    | 8,13 de 10,00 (81%)                   |

Pregunta 1  
Correcta  
Puntuó 1,00 sobre 1,00  
▼ Marcar pregunta

Tenemos el siguiente fichero, lineas.txt:

```
En un lugar
de la Mancha,
de cuyo nombre
no quiero acordarme,
```

Escribe lo que mostraría en consola el siguiente código:

```
Scanner sarrara = new Scanner(new File("lineas.txt"));
while (sarrara.hasNextLine()) {
    System.out.println("Linea: " + sarrara.nextLine());
}
```

Respuesta: (penalty regime: 10, 20, ... %)

```
1 | Linea: En un lugar
2 | Linea: de la Mancha,
3 | Linea:
4 | Linea: de cuyo nombre
5 | Linea:
6 | Linea: no quiero acordarme,
```

Todas las pruebas superadas. ✓

Question author's solution:

```
Linea: En un lugar
Linea: de la Mancha,
Linea:
Linea: de cuyo nombre
Linea:
Linea: no quiero acordarme,
```

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 2  
Correcta  
Puntuó 0,33 sobre 1,00  
▼ Marcar pregunta

Tenemos el siguiente texto:

```
String texto = "Erase una vez un caballero que salio por el mundo a buscar la vida y las aventuras.;"
```

¿Cómo podríamos crear un objeto de tipo Scanner para poder leer ese texto palabra a palabra utilizando los métodos hasNext() y next().

Seleccione una:

- a. Scanner leerString = new Scanner(texto); ✓
- b. Scanner leerString = new Scanner(new File(texto));
- c. Scanner leerString = Scanner(texto);
- d. Scanner leerString = texto;

Respuesta correcta

La respuesta correcta es: Scanner leerString = new Scanner(texto);

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,33/1,00.

Pregunta 3  
Correcta  
Puntuó 1,00 sobre 1,00  
▼ Marcar pregunta

Tenemos el siguiente fichero, lineas.txt:

```
Kaixo 34 45,78 zer
moduz zaude gaur?
100 90 ... 0
BUKAERA :)
```

¿Cuántos valores leería el siguiente código?

```
Scanner sarrara = new Scanner(new File("lineas.txt"));
int cont = 0;
while (sarrara.hasNext()) {
    System.out.println("Linea: " + sarrara.nextInt());
    cont++;
}
System.out.println("Total: " + cont);
```

Seleccione una:

- a. Leería 6: el 34, el 45, el 78, el 100, el 90 y el 0
- b. Ninguno, mostraría el error de ejecución "InputMismatchException" ✓
- c. Solo uno, el 34
- d. Leería 4: el 34, el 100, el 90 y el 0

Respuesta correcta

La respuesta correcta es: Ninguno, mostraría el error de ejecución "InputMismatchException"

Pregunta 4

Correcta  
Puntúa 1,00 sobre  
1,00  
▼ Marcar  
pregunta

Tenemos el siguiente fichero, lineas.txt:

```
Kaixo      34 45,78    zer
moduz zaude gaur?
100 90 ... 0
BUKAERA :)
```

¿Cuántos valores leería el siguiente código?

```
Scanner sarrera = new Scanner(new File("lineas.txt"));
int cont = 0;
while (sarrera.hasNextDouble()) {
    System.out.println("Linea: " + sarrera.nextDouble());
    cont++;
}
System.out.println("Total: " + cont);
```

Seleccione una:

- a. Ninguno, mostraría el error de ejecución "InputMismatchException"  
 b. Ninguno, mostraría el mensaje "Total: 0" ✓  
 c. Solo uno, el 45,78  
 d. Ninguno, daría error de compilación

Respuesta correcta

La respuesta correcta es: Ninguno, mostraría el mensaje "Total: 0"

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 5

Correcta  
Puntúa 1,00 sobre  
1,00  
▼ Marcar  
pregunta

Tenemos el siguiente fichero, lineas.txt:

```
En un lugar
de la Mancha,
de cuyo nombre
no quiero acordarme,
```

Escribe lo que mostraría en consola el siguiente código:

```
Scanner sarrera = new Scanner(new File("lineas.txt"));
while (sarrera.hasNext()) {
    System.out.println("Leido: " + sarrera.next());
}
```

Respuesta: (penalty regime: 10, 20, ... %)

```
1 |Leido: En
2 |Leido: un
3 |Leido: lugar
4 |Leido: de
5 |Leido: la
6 |Leido: Mancha,
7 |Leido: de
8 |Leido: cuyo
9 |Leido: nombre
10 |Leido: no
11 |Leido: quiero
12 |Leido: acordarme,
```

Todas las pruebas superadas. ✓

Question author's solution:

```
Leido: En
Leido: un
Leido: lugar
Leido: de
Leido: la
Leido: Mancha,
Leido: de
Leido: cuyo
Leido: nombre
Leido: no
Leido: quiero
Leido: acordarme,
```

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 6

Correcta  
Puntúa 1,00 sobre  
1,00  
▼ Marcar  
pregunta

Tenemos el siguiente fichero, clientes.txt:

```
125-1 Sur 105.70 225.00 157.00 147.00
147-1 Avelinos 224.00 157.00 224.00 38.50
189-1 Dolmar 178.00 201.00 265.00 265.00
145-1 Lazaro 56 285.30 104.00 98.50 125.00
121-1 Domino 87.90 56.00 105.00 74.00
138-1 Marimar 360.00 448.60 226.00 307.00
```

Escribe lo que mostraría en consola el siguiente programa:

```
import java.util.*;
import java.io.*;

public class Programa {
    public static void main(String[] args) throws FileNotFoundException {
        Scanner leerFich = new Scanner(new File("clientes.txt"));
        System.out.println("Ventas totales:");
        while (leerFich.hasNextLine()) {
            String linea = leerFich.nextLine();
        }
    }
}
```

```

Scanner leerLinea = new Scanner(linea);
leerLinea.useLocale(Locale.US);
String id = leerLinea.next();
String cliente = leerLinea.next();

double total = 0;
for (int i = 1; i < 4; i++) {
    double ventas = leerLinea.nextDouble();
    total = total + ventas;
}
System.out.println(cliente + " (" + id + "): " + total);
leerFich.close();
}
}

```

Respuesta: (penalty regime: 10, 20, ... %)

```

1 |Ventas totales:
2 |Sur (125-1): 487.7
3 |Avelinos (147-1): 605.0
4 |Doimar (189-1): 644.0
5 |Lazaro (145-1): 445.3
6 |Domino (121-1): 248.9
7 |Marimar (138-1): 1034.6

```

| Expected   | Got  | Comment  | Mark |
|--|--|--|------|
| Ventas totales:<br>Sur (125-1): 487.7<br>Avelinos (147-1): 605.0<br>Doimar (189-1): 644.0<br>Lazaro (145-1): 445.3<br>Domino (121-1): 248.9<br>Marimar (138-1): 1034.6 | Ventas totales:<br>Sur (125-1): 487.7<br>Avelinos (147-1): 605.0<br>Doimar (189-1): 644.0<br>Lazaro (145-1): 445.3<br>Domino (121-1): 248.9<br>Marimar (138-1): 1034.6 | Line 0 right<br>Line 1 right<br>Line 2 right<br>Line 3 right<br>Line 4 right<br>Line 5 right<br>Line 6 right | 1 ✓  |

Todas las pruebas superadas. ✓

Question author's solution:

```

Ventas totales:
Sur (125-1): 487.7
Avelinos (147-1): 605.0
Doimar (189-1): 644.0
Lazaro (145-1): 445.3
Domino (121-1): 248.9
Marimar (138-1): 1034.6

```

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 7

Correcta

Puntuó 1,00 sobre

1,00

▼ Marcar pregunta

Completa el método contarPalabras para que a partir de una cadena de texto, devuelva el número de palabras que tiene ese texto.

```

public static int contarPalabras(String texto) {
    Scanner leer = new Scanner(texto);
    int cont = 0;
    while (leer.hasNext()) {
        String palabra = leer.next();
        cont++;
    }
    return cont;
}

```

Nada

}

Respuesta correcta

La respuesta correcta es:

Completa el método contarPalabras para que a partir de una cadena de texto, devuelva el número de palabras que tiene ese texto.

```

[public static int contarPalabras(String texto) {
[Scanner leer = new Scanner(texto)];
[int cont = 0];
[while (leer.hasNext()) {
[String palabra = leer.next()];
[cont++];
}]
[return cont];
}]

```

Correcta

Puntos para este envío: 1,00/1,00.

Pregunta 8

Correcta

Puntuó 0,90 sobre

1,00

▼ Marcar pregunta

Corrige el siguiente programa para que lea el fichero imdb.txt y diga cuántas películas empiezan po "Ci" y cuáles son.

El fichero sigue el siguiente formato:

```

1 1649756 9.2 The Shawshank Redemption (1994)
2 1129349 9.2 The Godfather (1972)
3 770838 9.0 The Godfather: Part II (1974)

```

El resultado debería ser:

```

City of God (2002): 8.6
City Lights (1931): 8.5
Citizen Kane (1941): 8.4
Total películas: 3

```

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```

15 |     leerFich.close();
16 |
17 |
18 . public static boolean procesarLinea(String linea) {
19 |     boolean test = false ;
20 |     Scanner informacion = new Scanner(linea);
21 |     informacion.useLocale(Locale.US);
22 |     int posicion = informacion.nextInt();
23 |     int numOpiniones = informacion.nextInt();

```

```

24     double valoracionMedia = informacion.nextDouble();
25     String pelicula = informacion.nextLine();
26     pelicula = pelicula.substring(1);
27
28     if (pelicula.startsWith("Ci")) {
29         test = true;
30         System.out.println(pelicula + ": " + valoracionMedia);
31     }
32 }
33 }
34 }

```

| Expected   | Got  |
|--|--|
| City of God (2002): 8.6<br>City Lights (1931): 8.5<br>Citizen Kane (1941): 8.4<br>Total peliculas: 3 | City of God (2002): 8.6<br>City Lights (1931): 8.5<br>Citizen Kane (1941): 8.4<br>Total peliculas: 3 |

Todas las pruebas superadas. ✓

Question author's solution:

```

import java.util.*;
import java.io.*;

public class Programa {
    public static void main(String[] args) throws FileNotFoundException {
        Scanner leerFich = new Scanner(new File("imdb.txt"));

        int cont = 0;
        while (leerFich.hasNextLine()) {
            String linea = leerFich.nextLine();
            if (procesarLinea(linea)) {
                cont++;
            }
        }
        System.out.println("Total peliculas: " + cont);
        leerFich.close();
    }

    public static boolean procesarLinea(String texto) {
        Scanner leer = new Scanner(texto);
        leer.useLocale(Locale.US);

        boolean test = false;
        int posicion = leer.nextInt();
        int numOpiniones = leer.nextInt();
        double valoracionMedia = leer.nextDouble();
        String pelicula = leer.nextLine();
        pelicula = pelicula.substring(1);

        if (pelicula.startsWith("Ci")) {
            System.out.println(pelicula + ": " + valoracionMedia);
            test = true;
        }
        return test;
    }
}

```

Correcta

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,90/1,00.

Pregunta 9

Correcta

Puntúa 0,90 sobre

1,00

▼ Marcar pregunta

Completa el programa con el método cuantos() al que se le pasarán un objeto de tipo Scanner conectado a un fichero y una edad. Leerá el fichero y devolverá el número de personas con edad mayor a la indicada y mostrará sus nombres. Si todas son menores devolverá -1.

El fichero sigue el siguiente formato:

Marta 25 Nerea 30 Miren 27 Aitor 41 Javier 36

El resultado de las siguientes llamadas debería ser:

```

resultado = cuantos(leerFich, 30);
mostrará:
    Aitor: 41
    Javier: 36
devolverá 2

resultado = cuantos(leerFich, 45);
devolverá -1
Mostrará: No hay nadie mayor de 45

```

Respuesta: (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```

17     leerFich.close();
18 }
19 public static int cuantos(Scanner texto, int edad){
20     int cont = 0;
21     while(texto.hasNext()){
22         String nombre = texto.next();
23         int edadClinte = texto.nextInt();
24
25         if(edadClinte > edad){
26             System.out.println(nombre + ": " + edadClinte);
27             cont++;
28         }
29         else if(cont == 0){
30             cont = -1;
31         }
32     }
33 }
34 return cont;
35 }
36 }

```

| Input | Expected                       | Got                            |
|-------|--------------------------------|--------------------------------|
| 30    | Edad: Aitor: 41<br>Javier: 36  | Edad: Aitor: 41<br>Javier: 36  |
| 45    | Edad: No hay nadie mayor de 45 | Edad: No hay nadie mayor de 45 |

Todas las pruebas superadas. ✓

Question author's solution:

```

import java.util.*;
import java.io.*;

public class Ficheros {
    public static void main(String[] args) throws FileNotFoundException {
        Scanner leerFich = new Scanner(new File("edad.dat"));
        Scanner leerTeclado = new Scanner(System.in);

        System.out.print("Edad: ");
        int edadMinima = leerTeclado.nextInt();

        int total = cuantos(leerFich, edadMinima);

        if (total == -1) {
            System.out.println("No hay nadie mayor de " + edadMinima);
        }
    }

    public static int cuantos(Scanner texto, int edadMinima) {
        int cont = 0;
        while(texto.hasNext()){
            String nombre = texto.next();
            int edadClinte = texto.nextInt();
            if(edadClinte > edadMinima){
                System.out.println(nombre + ": " + edadClinte);
                cont++;
            }
        }
        return cont;
    }
}

```

```

        leerFich.close();
    }

    public static int cuantos(Scanner leer, int buscar) {
        int cont = -1;
        while (leer.hasNext()) {
            String nombre = leer.next();
            int edad = leer.nextInt();
            if (edad > buscar) {
                System.out.println(nombre + ":" + edad);
                cont++;
            }
        }
        return cont;
    }
}

```

**Correcta**

Puntos para este envío: 1,00/1,00. Contando con los intentos anteriores, daría 0,90/1,00.

**Pregunta 10**

Incorrecta

Puntúa 0,00 sobre  
1,00

Marcar  
pregunta

Completa el programa con el método abrirFichero al que se le pasará un objeto de tipo Scanner para leer datos por teclado. Pedirá el nombre del fichero que se quiere abrir, creará un objeto de tipo Scanner para leerlo y lo devolverá. Se considerará que el nombre del fichero que se introduce siempre es válido.

**Respuesta:** (penalty regime: 10, 20, ... %)

Reiniciar respuesta

```

1 import java.util.*;
2 import java.io.*;
3
4 public class Programa {
5     Scanner teclado = new Scanner(System.in);
6     Scanner leer = abrirFichero(teclado);
7     while (leer.hasNextLine()) {
8         System.out.println(leer.nextLine());
9     }
10 }
11 public static Scanner abrirFichero(Scanner leerTeclado){
12     System.out.println("¿Qué fichero quieres abrir?");
13     String extension=".txt";
14     String fichero = leerTeclado.nextLine() + extension;
15     Scanner leerFichero = new Scanner(new File(fichero));
16     return leerFichero;
17 }
18 }

```

**Syntax Error(s)**

```

prog.java:12: error: unmappable character for encoding ASCII
    System.out.println("??Qu?? fichero quieres abrir?");
                                ^
prog.java:12: error: unmappable character for encoding ASCII
    System.out.println("??Qu?? fichero quieres abrir?");
                                ^
prog.java:12: error: unmappable character for encoding ASCII
    System.out.println("??Qu?? fichero quieres abrir?");
                                ^
prog.java:12: error: unmappable character for encoding ASCII
    System.out.println("??Qu?? fichero quieres abrir?");
                                ^
prog.java:17: error: illegal start of type
    while (leer.hasNextLine()) {
        ^
prog.java:7: error: <identifier> expected
    while (leer.hasNextLine()) {
        ^
prog.java:7: error: illegal start of type
    while (leer.hasNextLine()) {
        ^
prog.java:7: error: <identifier> expected
    while (leer.hasNextLine()) {
        ^
prog.java:7: error: ';' expected
    while (leer.hasNextLine()) {
        ^
prog.java:7: error: illegal start of type
    while (leer.hasNextLine()) {
        ^
prog.java:8: error: illegal start of type
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: ';' expected
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: invalid method declaration; return type required
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: <identifier> expected
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: ';' expected
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: illegal start of type
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: <identifier> expected
    System.out.println(leer.nextLine());
        ^
prog.java:8: error: ';' expected
    System.out.println(leer.nextLine());
        ^
prog.java:11: error: class, interface, or enum expected
    public static Scanner abrirFichero(Scanner leerTeclado){
        ^
prog.java:13: error: class, interface, or enum expected
    String extension=".txt";
        ^
prog.java:14: error: class, interface, or enum expected
    String fichero = leerTeclado.nextLine() + extension;
        ^
prog.java:15: error: class, interface, or enum expected
    Scanner leerFichero = new Scanner(new File(fichero));
        ^
prog.java:16: error: class, interface, or enum expected
    return leerFichero;
        ^
prog.java:17: error: class, interface, or enum expected
    }
        ^
24 errors

```

**Question author's solution:**

```

import java.util.*;
import java.io.*;

public class Programa {

```

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

    Scanner leer = abrirFichero(teclado);

    while (leer.hasNextLine()) {
        System.out.println(leer.nextLine());
    }
    leer.close();
}

public static Scanner abrirFichero(Scanner leerTeclado) throws FileNotFoundException {
    System.out.print("Nombre del fichero: ");
    String nombreFich = leerTeclado.next();
    return new Scanner(new File(nombreFich));
}
```

**incorrecta**  
Puntos para este envío: 0,00/1,00.

[Finalizar revisión](#)



Urrutiko Lanbide Heziketako Institutua  
Instituto de Formación Profesional a Distancia

## Contacta con nosotros

📍 Dirección: Calle Álava 41, interior - Vitoria-Gasteiz  
📞 Teléfono : 945 567 953  
✉️ E-mail: [ulhi@ulhi.net](mailto:ulhi@ulhi.net)  
🐦 Twitter: [@UrrutikoLH](https://twitter.com/UrrutikoLH)



HEZKUNTZA SAILA  
Lanbide Heziketako Saila  
DEPARTAMENTO DE EDUCACIÓN  
Vicerrectorado de Formación Profesional