

ILERNA

Online

Videotutoría 11 (UF3): Ficheros(II)

Módulo 03A: Programación



Resumen

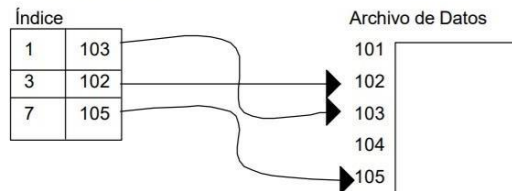
Podemos ver los **ficheros** como una parte de un **dispositivo no volátil** a la que se le asigna un nombre, y que puede contener una cantidad de datos que va a estar limitada, o por la cantidad de espacio del que disponga el dispositivo o por las características del sistema operativo.

- La información es un conjunto de 0 y 1.
- Al agrupar los bits se forman bytes o palabras.
- Al agrupar los campos se crean los registros de información.
- Un fichero es creado por un conjunto de registros de manera que todos tienen en común la misma estructura.
- Los directorios tienen la función de agrupar distintos ficheros siguiendo unas condiciones determinadas dadas por el sistema operativo o por el programador.

Secuencial: en los ficheros secuenciales los registros se van almacenando en posiciones consecutivas de manera que cada vez que queramos acceder a ellos tendremos que empezar desde el primero e ir recorriéndolos de uno en uno.



Esta tabla actúa como un índice



```
//Declaración de la variable del fichero  
fichero f1;
```

```
//Abrimos el fichero para leerlo  
f1.abrir(lectura);
```

```
Mientras no final de fichero hacer|  
    f1.leer(registro);  
    operaciones con registro leído;  
finMientras
```

```
//Cerramos el fichero  
f1.cerrar();
```

Resumen

```
//Declaración de la variable del fichero  
fichero f1;
```

```
//Abrimos el fichero para leerlo  
f1.abrir(escritura);
```

```
Mientras tengo información el fichero hacer  
    Configurar registros a partir de unos datos  
    f1. escribir(registro);
```

```
finMientras
```

```
//Cerramos el fichero  
f1.cerrar();
```

Si no cerramos la variable fichero nos daría un error.

Los flujos (también llamado Stream) de datos son las estructuras o pasarelas que tenemos para acceder a los datos de un fichero, de una forma consistente y fiable, desde un código fuente en un cualquier lenguaje de programación.

En el flujo de Entrada de datos (**Lectura**) solo podemos realizar la operación de lectura de un fichero, es decir, existe una comunicación unilateral desde el fichero al programa.

En el flujo de Salida (**Escritura**) también es en sentido unidireccional, ya que solo podemos realizar la operación de escritura en el fichero.

El tipo de stream lo vamos a **definir al principio** de trabajar con ficheros y no se podrá cambiar una vez abierto.

Ejemplo

En VisualStudio abrimos un proyecto como siempre. Tenemos que importar la biblioteca de entrada y salida el "using System.IO;"

FileStream fichero = new FileStream("ejemplo.txt", FileMode.Open, Open para abrir datos que ya existen sin borrar los anteriores.

FileAccess.Read); Lo vamos a leer. Por lo tanto este fichero ya tiene que estar creado. Por lo tanto hay que ir a la carpeta donde tenemos guardado nuestros proyectos. Vamos a la carpeta bin > Debug y creamos nuestro fichero ".ejemplo.txt".

StreamReader fs = new StreamReader(fichero);

Para conectar nuestra variable (pasarela) fichero con nuestro fichero físico.

string line;

while((line = fs.ReadLine()) != null) { Con using no se usa el while.

System.Console.WriteLine(line);

} Va a escribir por pantalla mientras no se llegue al final del fichero (leerlo que hay escrito en el fichero).

fs.close(); - Abrimos fichero.

- Abrimos pasarela.

fichero.Close(); - Cerramos pasarela.

- Cerramos fichero.

- En este ejemplo se lee el contenido de un archivo de texto línea a línea en una cadena mediante el método ReadLine de la clase StreamReader.
- Cada línea de texto se almacena en la cadena line y se muestra en la pantalla.

Directiva using

Utilizando using no tenemos que cerrar nosotros mismos la pasarela.

- Es simplemente una forma de indicar que un objeto se va a utilizar únicamente en un bloque de código y que una vez finalizado éste debe ser "destruido".

Revisar ruta relativa y ruta absoluta.

- Esto se puede aplicar al trabajo con el flujo de los ficheros

```
0 referencias
static void Main(string[] args)
{
    string fileName = @"mytest.txt"; @ Ruta relativa dónde nos va a gradar nuestro fichero.

    {
        Console.WriteLine("\n\n Create a file named mytest.txt in the disk :\n");
        Console.WriteLine("-----\n");
        // Create the file.
        using (FileStream fileStr = File.Create(fileName)) Creamos nuestra pasarela donde vamos a crear nuestro fichero.
        {
            Console.WriteLine(" A file created with name mytest.txt\n\n");
        }
        No trabajamos con ninguna pasarela que sea de escritura o lectura.
    }
}
```

Clase FileStream

- **Ficheros de texto:**
- Los datos pueden ser escritos o leídos de un fichero carácter a carácter las clases **StreamWriter** y **StreamReader**.
- **Ficheros binarios:**
- Para estos casos, el espacio de nombres System.IO proporciona las clases **BinaryReader** y **BinaryWriter**, las cuales permiten leer y escribir, respectivamente, datos de cualquier tipo primitivo en formato binario.

Ejemplo

0 referencias

```
static void Main(string[] args)
{
    string path = @".\escritura.txt";
    if (!File.Exists(path)) El if es para ver comprobar si el fichero existe, sin el if daría error si no existe.
    {
        // Create a file to write to.
        using (StreamWriter sw = File.CreateText(path))
        {
            sw.WriteLine("Hello"); Tenemos que hacer la escritura en la variable que utilizamos como pasarela.
            sw.WriteLine("And");
            sw.WriteLine("Welcome");
        }
    }
}
```

Preguntas UF3

Una ruta absoluta indica el camino de directorio desde la posición actual

- a) Verdadero
- b) Falso

Un fichero es un conjunto de registros almacenados

- a) Verdadero
- b) Falso

Preguntas UF3

Una ruta absoluta indica el camino de directorio desde la posición actual

a) Verdadero

b) Falso Es desde la raíz hasta donde tengamos el fichero.

Un fichero es un conjunto de registros almacenados

a) Verdadero

b) Falso

Preguntas UF3

¿Con qué clase trabajamos para la escritura de ficheros de texto?

- a) StreamWriter
- b) BinaryWriter
- c) BinaryReader
- d) System.io

En los ficheros secuenciales indexados, los registros poseen un campo clave para ser identificados

- a) Verdadero
- b) Falso

Preguntas UF3

¿Con qué clase trabajamos para la escritura de ficheros de texto?

- a) StreamWriter
- b) BinaryWriter
- c) BinaryReader
- d) System.io

En los ficheros secuenciales indexados, los registros poseen un campo clave para ser identificados

- a) Verdadero
- b) Falso

Preguntas UF3

Para un fichero binario, ¿cómo creamos un flujo de datos de escritura?

- a) `BinaryWriter` Es exactamente lo mismo que para ficheros de texto (stream). Por ejemplo un fichero .dat, ddl...
- b) `dataWriter`
- c) `BinaryReader`
- d) `dataWriteline`

Preguntas UF3

Para un fichero binario, ¿cómo creamos un flujo de datos de escritura?

- a) BinaryWriter
- b) dataWriter
- c) BinaryReader
- d) dataWriteline

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);
    fichero.Close();
}
```

¿De qué tipo es el fichero que estamos creando?

- a) texto
- b) binario
- c) Un fichero de procesamiento de lotes
- d) Fichero de datos

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);

    fichero.Close();
}
```

¿De qué tipo es el fichero que estamos creando?

a) texto

b) binario

c) Un fichero de procesamiento de lotes

d) Fichero de datos

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);
    fichero.Close();
}
```

¿Qué código falta en la línea 26?

- a) fs.close
- b) fs.open
- c) file.close
- d) file.open

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);
    fichero.Close();
}
```

¿Qué código falta en la línea 26?

- a) `fs.close` cerramos el flujo.
- b) `fs.open`
- c) `file.close`
- d) `file.open`

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);
    fichero.Close();
}
```

**¿Qué realiza el código del ejercicio?
Elige la respuesta más completa**

- a) Escribe dos números en un fichero que leen por teclado
- b) Escribe dos números en un fichero ordenador de mayor a menor
- c) Escribe dos números en un fichero ordenador de menor a mayor hasta que uno de los dos sea 0
- d) Escribe dos ceros en el fichero

Preguntas UF3

```
0 referencias
static void Main(string[] args)
{
    FileStream fichero = new FileStream("resultado.txt", FileMode.Create, FileAccess.Write);
    StreamWriter fs = new StreamWriter(fichero);
    int num1, num2, aux;

    do
    {
        Console.WriteLine("Introduce un número1:");
        num1 = Convert.ToInt32(Console.ReadLine());
        Console.WriteLine("Introduce un número2:");
        num2 = Convert.ToInt32(Console.ReadLine());

        if (num1 != 0 && num2 != 0)
        {
            if (num1 > num2)
            {
                aux = num1;
                num1 = num2;
                num2 = aux;
            }

            fs.WriteLine("{0}", "{1}", num1, num2);
        }
    } while (num1 != 0 || num2 != 0);
    fichero.Close();
}
```

**¿Qué realiza el código del ejercicio?
Elige la respuesta más completa**

- a) Escribe dos números en un fichero que leen por teclado
- b) Escribe dos números en un fichero ordenador de mayor a menor
- c) Escribe dos números en un fichero ordenador de menor a mayor hasta que uno de los dos sea 0**
- d) Escribe dos ceros en el fichero

Preguntas UF3

La operación Open:

- a) Abre el fichero para añadir más información a partir de lo escrito
- b) Abre el fichero en blanco
- c) Abre un fichero ya existente
- d) Abre un fichero en caso de que exista previamente en una carpeta o lo crea en caso contrario

Preguntas UF3

La operación Open:

- a) Abre el fichero para añadir más información a partir de lo escrito
- b) Abre el fichero en blanco
- c) Abre un fichero ya existente
- d) Abre un fichero en caso de que exista previamente en una carpeta o lo crea en caso contrario

