|  |  |
| --- | --- |
| **Ficheros de textos**  *• Abrir el fichero.*  *• Leer datos de él o escribir datos en él.*  *• Cerrar el fichero.*  **using** System**.**IO**;** | |
| **StreamWriter(“ nombreFichero”, true, System.Text.Encoding.Default)** | Clase escritora  Inicializa una nueva instancia de la clase StreamWriter para el archivo especificado usando la codificación y tamaño de búfer especificados. Si el archivo ya existe, es posible sobrescribir o anexar datos al mismo. Si el archivo no existe, este constructor crea uno nuevo. |
| **SreamReader(“ nombreFichero”, Boolean,Encoding.Default )** | Clase lectora |
| **Clase File** | fichero = File.CreateText("prueba.txt"); crear fichero |
| fichero = File.OpenText("prueba.txt");🡪 abrir para leer |
| fichero = File.AppendText("prueba2.txt");🡪abrir para añaddir |
| File.Exists(nombre)🡪 determina si existe el fichero |
|  |
| **EndOfStream;** | Devuelve true si esta al final del fichero. |
| **Método ReadLine()** | Lee línea a línea del fichero . 🡪fichero.ReadLine() |
| **Metoto ReadToEnd()** | Lee todos los caracteres desde la posición actual hasta el final de la secuencia. |
| **Método WriteLine()** | Escribe una línea en el fichero🡪fichero.Writeline(string) |
| **Metodo close()** | Cerrar fichero |
| **Excepciones** | •El fichero existe y es de sólo lectura (se lanzará una excepción del tipo "IOException").  •La ruta del fichero es demasiado larga (excepción de tipo "PathTooLongException").  •El disco puede estar lleno (IOException). |
| Control de formulario openFileDialog | |

**Ejercicios propuestos**:

Crear la clase fichero con los métodos:

* abrirFichero recibirá como parámetro el nombre del fichero. Utilizar excepciones para controlar la no existencia del fichero
* LeerLineaAlinea devolviendo un array con las líneas del fichero
* leeHastaFinal lee todas las líneas desde el punto en que se encuentra
* añadeLinea escribe una línea al final del fichero recibe como parámetro un string

2)

Crear un programa que gestione un fichero de frases célebres “frases.txt”. utilizando la clase fichero

Se programaran las siguientes opciones

* 1. Leer el fichero suministrado “frases.txt” , cargar cada frase en un ArrayList y mostrar todas las frases,
  2. Añadir nuevas frases al fichero. Cada frase tendrá la estructura: *“ frase célebre – autor”*
  3. Buscar frases por autor indicando el autor. Utilizar la función ***Split*** para separar frase de autor
  4. mostrar de 10 en 10 frases (crear un evento asociado a un botón para mostar las 10 siguientes cada vez que se pulse)
  5. Convertir a mayúsculas todas las frases y renombrar el fichero

