

FAMILIA PROFESIONAL:

CICLOS FORMATIVOS:

MÓDULO:

Informática y Comunicaciones

Desarrollo de Aplicaciones Multiplataforma,

Desarrollo de Aplicaciones Web

Programación

UNIDAD 9: FICHEROS

ACTIVIDADES



AUTORES: **Fernando Rodríguez Alonso**
Sonia Pasamar Franco

Este documento está bajo licencia Creative Commons Reconocimiento-NoComercial-SinObraDerivada 4.0 Internacional License.

Para ver una copia de esta licencia, visite <http://creativecommons.org/licenses/by-nc-nd/4.0/>.

Usted es libre de:

- **Compartir** — copiar y redistribuir el material en cualquier medio o formato.

El licenciante no puede revocar estas libertades en tanto usted siga los términos de la licencia.

Bajo los siguientes términos:

- **Atribución** — Usted debe dar crédito de manera adecuada, brindar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que usted o su uso tienen el apoyo de la licenciante.
- **NoComercial** — Usted no puede hacer uso del material con propósitos comerciales.
- **SinDerivadas** — Si remezcla, transforma o crea a partir del material, no podrá distribuir el material modificado.

No hay restricciones adicionales — No puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otras a hacer cualquier uso permitido por la licencia.

FICHEROS DE TEXTO

ACTIVIDAD 1x01

Codifica un programa principal **main** que:

- Lea un fichero de texto denominado `entrada.txt`.
- Cree otro fichero de texto denominado `salida.txt` y escriba en éste las mismas líneas del fichero de entrada pero en orden inverso.

Un ejemplo de ejecución del programa podría ser:

Fichero `entrada.txt`:

```
Este es el principio
Hola
Este es un fichero de texto
Adiós
Este es el fin
```

Fichero `salida.txt`:

```
Este es el fin
Adiós
Este es un fichero de texto
Hola
Este es el principio
```

ACTIVIDAD 1x02

Codifica un programa principal **main** que:

- Lea por teclado un nombre de fichero de texto.
- Si el nombre no se corresponde con ningún archivo (fichero, directorio o enlace) del sistema de archivos, visualizará en consola el mensaje:
El archivo no existe en el sistema de archivos.
- Si el nombre se corresponde con un archivo del sistema de archivos pero no es un fichero, visualizará en consola el mensaje:
El archivo no es un fichero.
- Si el nombre se corresponde con un fichero del sistema de archivos, leerá todas las líneas del fichero de texto indicado y creará un fichero de texto denominado `estadisticas.txt` que contenga lo siguiente:
 - El número de caracteres y el número de palabras de cada línea del fichero de texto.
 - El número total de líneas, el número total de caracteres y el número total de palabras del fichero de texto.

Un ejemplo de ejecución del programa podría ser:

Fichero `lineas.txt`:

```
Dos más tres son cinco.
Once menos uno son diez.
Siete por dos son catorce.
Quince entre dos son siete y medio.
```

Fichero estadisticas.txt:

Línea 1: 23 caracteres y 5 palabras

Línea 2: 24 caracteres y 5 palabras

Línea 3: 26 caracteres y 5 palabras

Línea 4: 35 caracteres y 7 palabras

Total: 4 líneas, 108 caracteres y 22 palabras

ACTIVIDAD 1x03

Codifica un programa principal **main** que gestione un fichero de texto denominado **frases.txt** mediante el siguiente menú de opciones:

0) Salir del programa.

1) Escribir varias frases en el fichero de texto.

Leerá por teclado de forma repetitiva frases y las escribirá en el fichero de texto de forma que su contenido previo no se sobrescriba. Este proceso deberá repetirse hasta que se introduzca la cadena *******, la cual no deberá escribirse en el fichero de texto.

2) Leer todas las frases del fichero de texto.

Leerá todas las frases del fichero de texto y las visualizará en consola en el mismo orden.

3) Consultar el número de líneas y el número de palabras del fichero de texto.

Calculará y visualizará en consola el número de líneas del fichero de texto.

Calculará y visualizará en consola el número de palabras del fichero de texto.

4) Buscar una palabra en el fichero de texto.

Leerá por teclado una palabra (que no tenga ningún espacio en blanco). Si la palabra no cumple la condición requerida, visualizará en consola un mensaje de error y leerá por teclado otra palabra. Este proceso se repetirá hasta que se introduzca una palabra válida.

Leerá las frases del fichero de texto para buscar la palabra en ellas, con dos situaciones posibles:

- Si la palabra está contenida en alguna de las frases del fichero de texto, visualizará en consola un mensaje indicando la línea en la que aparece por primera vez en el fichero de texto:

La palabra está en la línea 5 del fichero de texto.

- Si la palabra no está contenida en ninguna de las frases del fichero de texto, visualizará en consola el mensaje:

La palabra no está en el fichero de texto.

5) Crear otro fichero de texto con las frases del fichero de texto convertidas a mayúsculas.

Crearé un fichero de texto denominado **mayusculas.txt**.

Leerá cada frase del fichero de texto **frases.txt**, la convertirá a letras mayúsculas y escribirá esta frase convertida en el fichero **mayusculas.txt**. La escritura de estas frases convertidas en el fichero **mayusculas.txt** deberá empezar por el principio del fichero de texto, de forma que se sobrescriba su contenido previo.

6) Crear otro fichero de texto con las frases del fichero de texto convertidas a minúsculas.

Crearé un fichero de texto denominado `minusculas.txt`.

Leeré cada frase del fichero de texto `frases.txt`, la convertirá a letras minúsculas y escribiré esta frase convertida en el fichero `minusculas.txt`. La escritura de estas frases convertidas en el fichero `minusculas.txt` deberá empezar por el principio del fichero de texto, de forma que se sobrescriba su contenido previo.

Este programa principal deberá validar que la opción de menú elegida sea válida (comprendida entre 0 y 6). Si no lo es, visualizará en consola el mensaje:

La opción de menú debe estar comprendida entre 0 y 6.

ACTIVIDAD 1x04

Codifica una clase **Empleado** para tratar la información de los diferentes empleados de una empresa. De cada uno de ellos se desea guardar: un código, el nombre, la fecha de nacimiento, el departamento y el salario en euros.

El código de cada empleado será un número entero positivo identificativo. Cada empleado tendrá su propio código, es decir, no podrá haber dos o más empleados con el mismo código en la empresa.

Para esta clase **Empleado**:

- Codifica un constructor que reciba como parámetros el código, el nombre, la fecha de nacimiento, el departamento y el salario.
- Codifica otro constructor que reciba como parámetro una línea de texto, que extraiga varias subcadenas de esta línea de texto utilizando el carácter ';' como separador y que inicialice los atributos del objeto con estas subcadenas generadas. Según el atributo, además podría ser necesario convertir una subcadena a número entero o real.
- Codifica un método de objeto que devuelva una cadena de texto con el nombre de la clase (Empleado) y un resumen con los nombres y valores de todos los atributos del objeto. Para ello, sobrescribe el método **toString** que procede de la clase **Object**. El salario se deberá indicar con 2 dígitos decimales. Este método se utilizará para visualizar un empleado en consola.
- Codifica otro método de objeto **toStringWithSeparators** que devuelva una cadena de texto con los valores de todos los atributos del objeto, separados entre sí por el carácter ';'. El salario se deberá indicar con dos dígitos decimales y utilizando el carácter ',' para separar la parte entera de la parte fraccionaria. Este método se utilizará para guardar un empleado como una línea en un fichero de texto.

Codifica una clase **Actividad_1x04** que incluya un programa principal **main**. Este programa realizará operaciones de mantenimiento sobre un fichero de texto denominado `empleados.txt`. Este fichero almacenará los datos de cada empleado como una línea en el fichero de texto. Un ejemplo del contenido de este fichero, con 4 empleados, podría ser:

```
1;Juan Serrano;30/05/1985;Contabilidad;3025.87
2;Isabel Martínez;03/12/1992;Compras;2105.12
3;Jorge Gil;17/02/1998;Ventas;2003.98
4;Elena Gracia;24/08/2002;Recursos Humanos;1853.35
```

Codifica un programa principal **main** que gestione el fichero de texto `empleados.txt` mediante el siguiente menú de opciones:

0) Salir del programa.

1) Insertar un empleado en el fichero de texto.

Leerá por teclado el código, el nombre, la fecha de nacimiento, el departamento y el salario del empleado a insertar.

Si el fichero de texto contiene un empleado con el código dado, visualizará en consola el mensaje:

Ya existe otro empleado con ese código en el fichero de texto.

Si el fichero de texto no contiene ningún empleado con el código dado, insertará el empleado al final del fichero de texto y visualizará en consola el mensaje:

Se ha insertado un empleado en el fichero de texto.

2) Consultar todos los empleados del fichero de texto.

Si el fichero de texto no contiene ningún empleado, visualizará en consola el mensaje:

El fichero de texto está vacío.

Si el fichero de texto contiene uno o más empleados:

- Leerá todos los empleados del fichero de texto.
- Visualizará en consola un listado de todos los empleados almacenados en el fichero de texto, indicando para cada empleado, un resumen con los nombres y valores de todos los atributos de dicho empleado.
- Visualizará en consola el número de empleados consultados del fichero de texto.

Por ejemplo:

```
Empleado [Código = 1, Nombre = Juan Serrano,  
          FechaNacimiento = 30/05/1985,  
          Departamento = Contabilidad, Salario = 3025,87]  
Empleado [Código = 2, Nombre = Isabel Martínez,  
          FechaNacimiento = 03/12/1992,  
          Departamento = Compras, Salario = 2105,12]  
Empleado [Código = 3, Nombre = Jorge Gil,  
          FechaNacimiento = 17/02/1998,  
          Departamento = Ventas, Salario = 2003,98]  
Empleado [Código = 4, Nombre = Elena Gracia,  
          FechaNacimiento = 24/08/2002,  
          Departamento = Recursos Humanos, Salario = 1853,35]  
Se han consultado 4 empleados del fichero de texto.
```

3) Consultar un empleado, por código, del fichero de texto.

Leerá por teclado el código del empleado a consultar.

Si el fichero de texto no contiene ningún empleado con el código dado, visualizará en consola el mensaje:

No existe ningún empleado con ese código en el fichero de texto.

Si el fichero de texto contiene un empleado con el código dado, consultará el empleado del fichero de texto y visualizará en consola un resumen con los nombres y valores de todos los atributos de dicho empleado.

El siguiente ejemplo muestra el empleado cuyo código es 2:

```
Empleado [Código = 2, Nombre = Isabel Martínez,  
          FechaNacimiento = 03/12/1992,  
          Departamento = Compras, Salario = 2105,12]
```

4) Actualizar un empleado, por código, del fichero de texto.

Leerá por teclado el código, el nuevo departamento y el nuevo salario del empleado a actualizar.

Si el fichero de texto no contiene ningún empleado con el código dado, visualizará en consola el mensaje:

No existe ningún empleado con ese código en el fichero de texto.

Si el fichero de texto contiene un empleado con el código dado, actualizará el empleado del fichero de texto con el nuevo departamento y el nuevo salario y visualizará en consola el mensaje:

Se ha actualizado un empleado del fichero de texto.

Para realizar esta operación de actualización, será necesario utilizar una colección auxiliar o un fichero de texto auxiliar.

5) Eliminar un empleado, por código, del fichero de texto.

Leerá por teclado el código del empleado a eliminar.

Si el fichero de texto no contiene ningún empleado con el código dado, visualizará en consola el mensaje:

No existe ningún empleado con ese código en el fichero de texto.

Si el fichero de texto contiene un empleado con el código dado, eliminará el empleado del fichero de texto y visualizará en consola el mensaje:

Se ha eliminado un empleado del fichero de texto.

Para realizar esta operación de eliminación, será necesario utilizar una colección auxiliar o un fichero de texto auxiliar.

Este programa principal deberá validar que la opción de menú elegida sea válida (comprendida entre 0 y 5). Si no lo es, visualizará en consola el mensaje:

La opción de menú debe estar comprendida entre 0 y 5.