Ejercicios – **Ficheros de texto**

Utiliza tu **IDE** preferido: Apache NetBeans, Eclipse o IntelliJ IDEA.

Crea un Nuevo **proyecto** Java llamado **Ficheros-Texto**.
Este proyecto contendrá las aplicaciones propuestas a continuación.

Utiliza siempre que sea posible las clases de **java.io** para **lectura y escritura con buffer**, y la sentencia **try-with-resources**.

Algunas de estas aplicaciones se podrán utilizar como **comandos** en la linea de órdenes; ese es el motivo de que su nombre aparezca en minúsculas.

1) Aplicación jtype

Crea una Aplicación Java que visualice en consola el contenido de un fichero de texto dado (similar al comando **type** de Windows o **cat** de Unix).

- El nombre del fichero se pedirá por teclado.

2) Aplicación jcopytext

Crea una Aplicación Java que realice una copia exacta de un fichero de texto dado (similar al comando **copy** de Windows o **cp** de Unix).

- El nombre del fichero a copiar se pedirá por teclado.
- El nombre del fichero copia, se podrá pedir también o, simplemente, agregar la palabra "_copy" al nombre.

3) Aplicación jwc

Crea una Aplicación Java que muestre el número de caracteres y el número de líneas que tiene un fichero de texto dado (similar al comando **wc** de Unix).

- El nombre del fichero se pedirá por teclado.

4) Uso de argumentos en la línea de órdenes

Convierte las aplicaciones jtype, jcopytext y jwc para que soporten argumentos en la línea de órdenes, de modo que si se indica en la linea de órdenes, no se pedirá por teclado.

La sintaxis será la siguiente:

a) jtype <*input_filename*>

Ejemplo:
java **jtype** d:/pruebas/fichero.txt

b) jcopytext <*input_filename*> [<*output_filename*>]

Ejemplos:
java **jcopytext** Main.java
Fichero copiado con nombre Main_copy.java

java **jcopytext** d:/pruebas/fichero.txt d:/pruebas/copia.txt
Fichero copiado con nombre copia.txt

c) jwc [*opción*] <*input_filename*>

opción podrá ser:
- -l muestra número de líneas
- -c muestra número de caracteres
- -w muestra número de palabras

Si no se indica opción, mostrará los tres valores.
Comando wc: https://es.wikipedia.org/wiki/Wc_(Unix)

Ejemplos:

java **jwc -l** fichero.txt
3 líneas

java **jwc -c** fichero.txt
45 caracteres

java **jwc -w** fichero.txt
12 palabras

java **jwc** fichero.txt
3 líneas, 45 caracteres, 12 palabras

5) Aplicación CalcTextFile

Crea una Aplicación Java que realice el cálculo indicado en un fichero de texto.

- El nombre del fichero se pedirá por teclado.
- El contenido del fichero será: en la primera línea aparecerá la operación aritmética (+, -, x). Después aparecerá un número entero en cada linea (sin límite de líneas).
- La aplicación deberá aplicar la operación a todos los números y **escribir en consola** el resultado obtenido.

Tres ejemplos:

	+	**x**	**-**
	5	404	1
	133	100	1
	21		
En consola:	**159**	40400	**0**

6) Aplicación CryptTextFile

Crea una Aplicación Java que permita obtener una versión encriptada de un fichero de texto existente.

- El nombre del fichero se pedirá por teclado.
- El nombre del fichero encriptado se podrá pedir también o, simplemente, agregar la palabra "_crypt" al nombre.
- Un método sencillo podría ser el de la sustitución de cada carácter por otro, usando un desplazamiento en el valor del carácter.

  Por ejemplo, usando desplazamiento 3, si el carácter es A (65), se sustituiría por D (68).

7) Aplicación CensureTextFile

Crea una Aplicación Java que genere una nueva versión de un fichero de texto dado, en el que determinadas palabras no aparezcan.

- El nombre del fichero de entrada se pedirá por teclado (y/o por la linea de órdenes).

- El nombre del fichero de salida podrá ser el mismo nombre del de entrada, añadiendo la palabra "_censored".

- Las palabras "prohibidas" se darán en un fichero de texto, donde cada palabra aparecerá en una linea.

Primero:
La aplicación debe **cargar las palabras prohibidas**.

- Utiliza una Estructura de Datos Dinámica puesto que no sabemos el número de palabras que habrá en el fichero.

Segundo:
La aplicación debe **procesar el fichero de entrada**, generando un nuevo fichero donde aparezca una cadena de asteriscos (*) en lugar de la palabra prohibida.

- Debe leer el fichero linea por linea, extrayendo las palabras de la linea (puedes usar el método split) y buscando si existe alguna prohibida.

- La palabra prohibida se sustituye por una cadena de '*' con la misma longitud que dicha palabra.