

# Manejo de archivos de texto

Emmanuel Cruz Hernández

`emmanuel_cruzh@ciencias.unam.mx`

21 de enero de 2020

# Contenido

- 1 Introducción
- 2 Lectura de ficheros
- 3 Escritura de ficheros
- 4 Bibliografía

Un problema que hemos tenido hasta ahora es la pérdida de información al ejecutar un programa. Cuando ejecutamos un programa podemos realizar ciertas operaciones que no se guardan en memoria, así que el estado de un objeto está limitado al tiempo de ejecución.

La posibilidad de tener entrada y salida para un programa permite el almacenamiento de información.

La información guardada representa los datos de un estado particular de un objeto.

El almacenamiento de información de un programa estará representado por la lectura y escritura de ficheros o archivos. [1]

La lectura y escritura de un archivo puede lanzar excepciones, tales como errores de lectura, escritura o no encontrar el archivo a leer. Esto quiere decir que la lectura de un archivo se realiza dentro de un bloque **try... catch**.

La lectura de un fichero permitirá recuperar la información de un estado de las variables en un momento específico. De esta forma, se recupera la información para que un programa tome los datos, los pueda abstraer y los pueda utilizar.

# Leer un fichero en Java

Para leer un fichero en Java se necesita de la importación de las clases *Reader*, *FileReader* y *BufferedReader*. Estas clases nos van a servir para tener un lector de datos, para leer un archivo y para tener un buffer en el cuál se estará leyendo la entrada, respectivamente.

Todas estas clases se importan de `java.io` que se refiere a entrada (in) y salida (out).

# Forma de declarar la lectura de un fichero

```
try{  
    Reader archivo = new FileReader("NombreArchivo.txt");  
    BufferedReader lector = new BufferedReader(archivo);  
    ...  
} catch(Exception e){  
    ...  
}
```



La escritura de información en un fichero permitirá el almacenamiento del estado del objeto de un programa.

En general, la lectura de un archivo puede guardar cualquier dato generado durante la ejecución de un programa.

# Escribir un fichero en Java

Para escribir un fichero en Java se necesita de la importación de las clases *Writer*, *FileWriter* y *BufferedWriter*. Estas clases nos van a servir para tener un escritor de datos, para escribir sobre un archivo y para tener un buffer en el cuál se estará almacenando la información, respectivamente. Todas estas clases también se importan de `java.io`.

# Forma de declarar la escritura de un fichero

```
try{
    Writer archivo = new FileWriter("NombreArchivo.txt");
    BufferedWriter lector = new BufferedWriter(archivo);
    ...
} catch(Exception e){
    ...
}
```



Elisa Viso G. Canek Peláez V.

*Introducción a las Ciencias de la Computación con Java.*

Segunda edition, 2012.