



**UNIVERSIDAD AUTONOMA
DE CAMPECHE**



Facultad de ingeniería

**Ingeniería en sistemas
computacionales**

Retroalimentación practica 7 Filabs

Materia:

Lenguaje de programación II

Alumno:

Néstor Damián Hernández Medina

3er.Semestre Grupo: "A"

Profesor:

Guadalupe M Estrada Segovia

Fecha de entrega:

21/11/2023

Investigación

Retroalimentación práctica 7

Investiga las características de los tres modos de trabajar con archivo que hay en Java

(texto, binario y objeto).

Texto:

Está formado por secuencias de caracteres, organizados en líneas de igual o distinta

longitud. Todos los datos que aparecen están formados por caracteres.

“File”: Mediante la cual indicamos la ruta del fichero.

“FileOutputStream”: Mediante la cual indicamos con el valor «true» que vamos a añadir

texto.

“PrintWriter”: La usamos para añadir el texto.

Binario:

Un fichero binario o de datos está formado por secuencias de bytes. Estos archivos pueden

contener datos de tipo básico (int, float, char, etc) y objetos. Para poder leer el contenido de

un fichero binario debemos conocer la estructura interna del fichero, es decir, debemos

saber cómo se han escrito: si hay enteros, long, etc. y en qué orden están escritos en el

fichero. Si no se conoce su estructura podemos leerlo byte a byte.

La clase “File” para indicar la ruta del fichero donde vamos a escribir.

La clase “FileOutputStream”. Se trata de una clase intermedia.

La clase “DataOutputStream”. Mediante esta clase escribiremos los datos binarios.

Objeto:

Una de las características más potentes del lenguaje para persistir el estado de

nuestros programas. A esto se le llama serialización, y funciona de la siguiente manera.

Para serializar un objeto debemos utilizar 3 clases:

La clase “File” que apuntará al fichero donde se guardarán nuestros datos.

La clase “FileOutputStream” que se trata de una clase intermedia.

La clase “ObjectOutputStream” que será la clase que contiene el método necesario para

serializar la clase.

- Investiga en qué situaciones es mejor utilizar un tipo de archivo u otro

Los formatos textuales trabajan mucho mejor con sistemas de control de versiones, los

cuales únicamente pueden comprobar cambios significativos en archivos de texto. Además,

los formatos en texto pueden ser más duraderos; si existe una corrupción en algún sitio en

el archivo, puede ser fácilmente arreglado abriendo un editor de texto.

Por otro lado, también es difícil su legibilidad. Así pues, en ocasiones es preferible tener

datos almacenados en un archivo CSV y tener por otro lado un archivo de código que

especifique los metadatos. El complemento al formato textual es el binario, el cual es

necesario en ocasiones por eficiencia, o porque no existe manera de representación en

formato textual. También, con datos numéricos, podemos perder precisión cuando los

convertimos a un formato textual, es entonces cuando es apropiado utilizar el formato

binario.