

¿Qué es un archivo?

Un archivo no es más que una estructura que almacena información en el disco y se puede recuperar con el tiempo.

Las operaciones más básicas para la gestión de archivos, tanto de lectura como de escritura, se realizan a través de un flujo que no es más que un canal por el que se envía información al archivo.

En Java existen varias clases que implementan métodos para la lectura y escritura de archivos, y que lo hacen a niveles muy bajos que son bytes, y a niveles más altos como son caracteres, cadenas y hasta objetos.

Clases que se utilizan para leer y escribir en archivos

ESCRIBIR

❖ FileWriter

Esta clase permite la escritura de flujos de caracteres, es decir una cadena por ejemplo, tiene varios constructores, pero dejo los 2 más utilizados.

```

1 import java.io.*;
2 import java.util.*;
3
4 public class Archivo
5 {
6     public static void main(String[] args) throws IOException
7     {
8         File file = new File("Hola.txt");
9         file.createNewFile();
10        FileWriter writer = new FileWriter(file);
11        writer.write("Ejemplo de archivo");
12        writer.flush();
13        writer.close();
14    }
15 }

```

❖ BufferedWriter

Se puede escribir archivos con `FileWriter` tranquilamente pero es muy costoso.

Por ejemplo, si quieres escribir 10 líneas de texto, con `FileWriter` se tiene que hacer 10 accesos al disco, uno por cada línea, debido a que el buffer es más pequeño.

Esto se soluciona con `BufferedWriter`, ya que guarda todo el texto que quieres escribir en un buffer intermedio y de ahí lo envía al archivo y no hacer los 10 accesos al disco por cada línea.

```

//Sample 01: Open the FileWriter, Buffered Writer
FileWriter fw = new FileWriter( fileName: "C:\\Temp\\TestFile.Txt");
BufferedWriter WriteFileBuffer = new BufferedWriter(fw);

//Sample 02: Write Some Text to File Using Buffered Writer
WriteFileBuffer.write( str: "First Line");
WriteFileBuffer.newLine();
WriteFileBuffer.write( str: "Second Line");
WriteFileBuffer.newLine();
WriteFileBuffer.write( str: "Third Line");
WriteFileBuffer.newLine();

//Sample 03: Close both the Writers
WriteFileBuffer.close();

```

The diagram includes five numbered callouts pointing to specific lines of code:

- 1 points to the `new BufferedWriter(fw);` line in Sample 01.
- 2 points to the `WriteFileBuffer.write(str: "First Line");` line in Sample 02.
- 3 points to the `WriteFileBuffer.write(str: "Second Line");` line in Sample 02.
- 4 points to the `WriteFileBuffer.newLine();` line at the end of Sample 02.
- 5 points to the `WriteFileBuffer.close();` line in Sample 03.

LEER

❖ **BufferedReader y FileReader**

Funcionan igual que las anteriores pero en este caso para leer datos de un archivo, es decir la primera crea el flujo principal y la segunda el flujo intermedio.

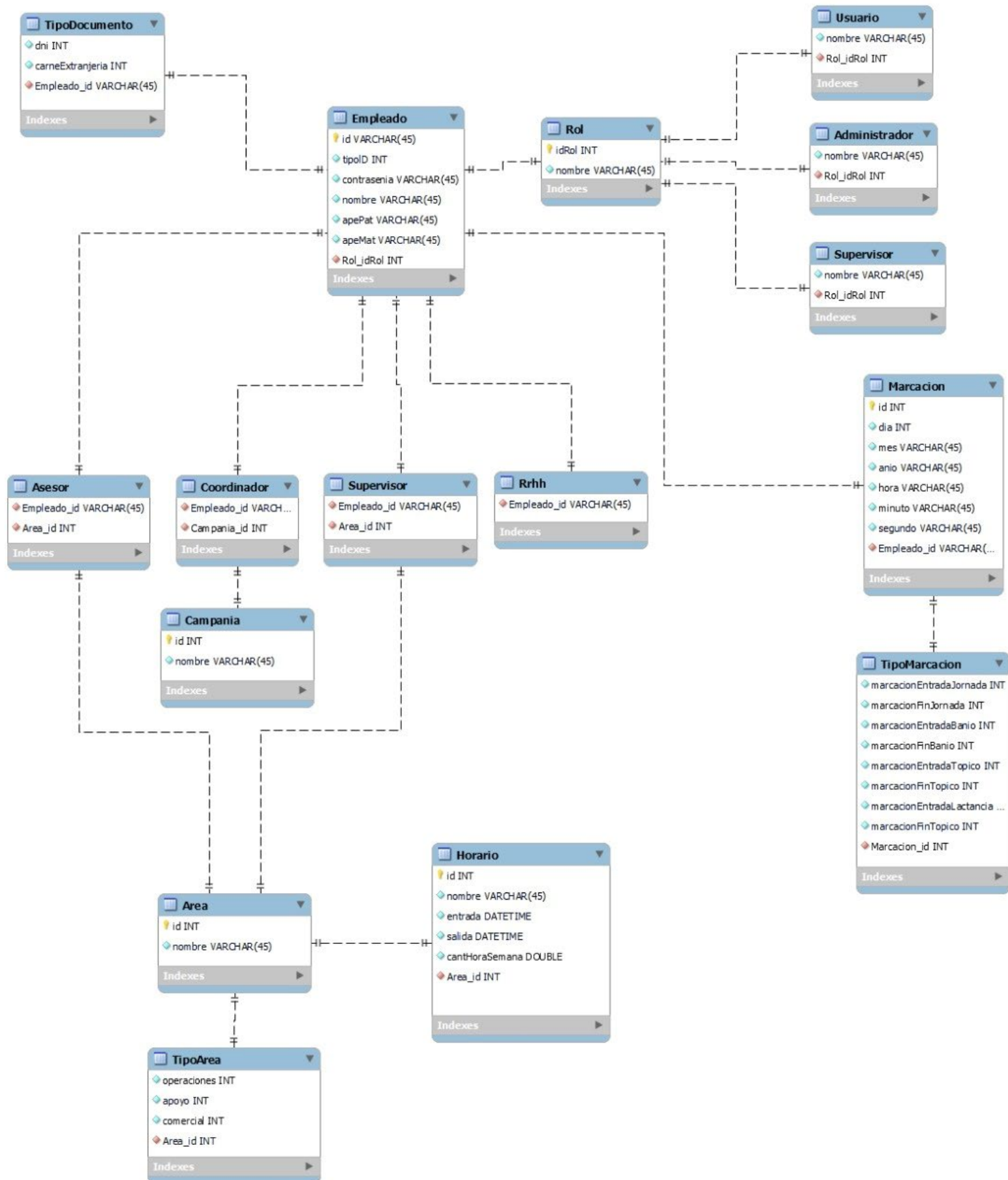
```
1
2 package abriertextnew;
3 import java.io.*;
4 public class Abriertextnew {
5
6     public static void main(String[] args)
7         throws IOException {
8         muestraContenido ("c:/archivos/archivo.txt");
9     }
10
11     public static void muestraContenido(String archivo)
12         throws FileNotFoundException, IOException {
13         String cadena;
14         FileReader f= new FileReader(archivo);
15         BufferedReader b=new BufferedReader(f);
16         while(( cadena=b.readLine())!=null) {
17             System.out.println(cadena);
18         }
19         b.close();
20     }
21 }
```

❖ **Scanner**

Otra opción es utilizar el `Scanner` clase para leer cada línea del archivo con el `nextLine()` método. El siguiente es un ejemplo simple que demuestra el uso de este método. Tenga en cuenta que el `hasNextLine()` El método devuelve verdadero solo si al escáner le quedan más líneas y el `nextLine()` El método avanza el escáner a la

siguiente línea y devuelve la línea actual.

```
1  import java.io.File;
2  import java.io.IOException;
3  import java.util.Scanner;
4
5  public class Main {
6      public static void main(String[] args) {
7          String fileName = "/home/data/file.txt";
8
9          try {
10             Scanner sc = new Scanner(new File(fileName));
11             while (sc.hasNextLine()) {
12                 String line = sc.nextLine();
13                 System.out.println(line);
14             }
15         } catch (IOException e) {
16             e.printStackTrace();
17         }
18     }
19 }
```



DICCIONARIO DEL DIAGRAMA DE BASE DE DATOS

TABLA: EMPLEADO

Esta tabla muestra la información detallada de los EMPLEADOS registrados en la base de datos

Columnas:

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	X	ID	VARCHAR(45)	Cadena identificadora. Se ha convenido que este sea el DNI (Documento nacional de Identidad) o CE (Carne de extranjería). Debe de ser único para cada instancia
2		TIPOID	INT	Tipo de documento. Por el momento puede ser de tipo DNI o CARNEEXTRANJERIA
3		CONTRASENIA	VARCHAR(45)	Cadena que representa la contraseña del usuario.
4		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del usuario.
5		APEPAT	VARCHAR(45)	Cadena que representa el Apellido paterno del usuario.
6		APEMAT	VARCHAR(45)	Cadena que representa el Apellido materno del usuario.
7		ROL	VARCHAR(45)	Rol que tiene un usuario para ejercer funciones en el sistema

TABLA: TIPODOCUMENTO

Esta tabla muestra la información detallada de los **TIPODOCUMENTO** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		DNI	INT	Entero que representa el DNI del usuario.
2		CARNEEXTRANJERIA	INT	Entero que representa el CARNEEXTRANJERIA del usuario.

TABLA: ASESOR

Esta tabla muestra la información detallada de los **ASESORES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Empleado_id	VARCHAR(45)	Cadena identificadora en este caso DNI del empleado
2		Area_id	INT	Numero identificador en este caso del area

TABLA: COORDINADOR

Esta tabla muestra la información detallada de los **COORDINADORES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Empleado_id	VARCHAR(45)	Cadena identificadora en este caso DNI del empleado
2		Campania_id	INT	Numero identificador en este caso de la campania

TABLA: SUPERVISOR

Esta tabla muestra la información detallada de los **SUPERVISORES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Empleado_id	VARCHAR (45)	Cadena identificadora en este caso DNI del empleado
2		Area_id	INT	Numero identificador en este caso del area

TABLA: RRHH

Esta tabla muestra la información detallada de los **RRHHS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Empleado_id	VARCHAR (45)	Cadena identificadora en este caso DNI del empleado

TABLA: CAMPANIA

Esta tabla muestra la información detallada de las **CAMPANIAS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	x	ID	INT	Identificador de la campaña.
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre de la campaña.
3		COORDINADOR	VARCHAR(45)	Gestor de la campaña tipo Coordinador. Solo debe de existir uno por instancia de campaña.
4		AREAS	VARCHAR(45)	ArrayList de áreas que tiene una campaña.

TABLA: AREA

Esta tabla muestra la información detallada de las **AREAS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	X	ID	INT	Numero identificador para el área
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del área.
3		TIPO	INT	Tipo de área. Por el momento puede ser de tipo OPERACIONES, APOYO y COMERCIAL.
4		SUPERVISOR	VARCHAR(45)	Gestor de área de tipo Supervisor. Solo puede y debe de tener uno por área.
5		ASESORES	VARCHAR(45)	ArrayList de asesores que tiene una campaña.
6		HORARIO	VARCHAR(45)	Rango de horas laborables de un área de tipo Horario. Solo puede tener un horario por área.

TABLA: TIPOAREA

Esta tabla muestra la información detallada de las **TIPOAREAS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		OPERACIONES	INT	Numero identificador para OPERACIONES
2		APOYO	INT	Numero identificador para APOYO
3		COMERCIAL	INT	Numero identificador para COMERCIAL

TABLA: HORARIO

Esta tabla muestra la información detallada de las **HORARIOS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	x	ID	INT	Identificador del horario.
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del horario.
3		ENTRADA	DATETIME	Fecha y hora de entrada del horario.
4		SALIDA	DATETIME	Fecha y hora de salida del horario.

TABLA: ROL

Esta tabla muestra la información detallada de las **ROLES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	X	ID	INT	Numero identificador para ROL
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del ROL.

TABLA: USUARIO

Esta tabla muestra la información detallada de las **USUARIOS** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Rol_idRol	INT	Numero identificador para USUARIO
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del USUARIO.

TABLA: ADMINISTRADOR

Esta tabla muestra la información detallada de las **ADMINISTRADORES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Rol_idRol	INT	Numero identificador para ADMINISTRADOR
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del ADMINISTRADOR .

TABLA: SUPERVISOR

Esta tabla muestra la información detallada de las **SUPERVISORES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1		Rol_idRol	INT	Numero identificador para SUPERVISOR
2		NOMBRE	VARCHAR(45)	Cadena que representa el nombre del SUPERVISOR .

TABLA: MARCACION

Esta tabla muestra la información detallada de las **MARCACIONES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	X	ID	INT	Numero Identificador de la marcación.
2		DIA	INT	Tipo de documento. Por el momento puede ser de tipo DNI o CARNEEXTRANJERIA
3		MES	VARCHAR(45)	Cadena que representa el día en el que se hizo la marcación.
4		ANIO	VARCHAR(45)	Cadena que representa el mes en el que se hizo la marcación.
5		HORA	VARCHAR(45)	Cadena que representa la hora en el que se hizo la marcación.
6		MINUTO	VARCHAR(45)	Cadena que representa el minuto en el que se hizo la marcación.
7		SEGUNDO	VARCHAR(45)	Cadena que representa el segundo en el que se hizo la marcación.
8		TIPO	INT	Numero de Tipo de Marcación por el momento puede ser de tipo MARCACIONENTRAJORNADA, MARCACIONFINJORNADA, MARCACIONENTRADABANIO, MARCACIONFINBANIO, MARCACIONENTRADATOPICO, MARCACIONFINTOPICO, MARCACIONENTRADALACTANCIA y MARCACIONFINLACTANCIA

TABLA: TIPOMARCACION

Esta tabla muestra la información detallada de las **TIPOMARCACIONES** registrados en la base de datos.

#	PK	NOMBRE	TIPO DE DATO	DESCRIPCION
1	X	MARCACIONENTRAJORNADA	INT	Numero identificador para MARCACIONENTRAJORNADA
2		MARCACIONFINJORNADA	INT	Numero identificador para MARCACIONFINJORNADA
3		MARCACIONENTRADABANIO	INT	Numero identificador para MARCACIONENTRADABANIO
4		MARCACIONFINBANIO	INT	Numero identificador para MARCACIONFINBANIO
5		MARCACIONENTRADATOPICO	INT	Numero identificador para MARCACIONENTRADATOPICO
6		MARCACIONFINTOPICO	INT	Numero identificador para MARCACIONFINTOPICO
7		MARCACIONENTRADALACTANCIA	INT	Numero identificador para MARCACIONENTRADALACTANCIA
8		MARCACIONFINLACTANCIA	INT	Numero identificador para MARCACIONFINLACTANCIA