

# Fundamentos de Bases de Datos

## Practica 2

### Manipulación de archivos

01 de Marzo del 2019

## 1. Implementación del programa en Java

En esta practica se hace un avance del caso de uso enfocado en la información de las sucursales y empleados, así que se implementa un programa escrito en Java que simulará una base de datos, solicita la información y lo guarda en un archivo CSV.

De acuerdo a los requerimientos solicitados en el caso de uso, para los empleados se almacena su información básica: nombre, apellido paterno, apellido materno, puesto salario y sucursal a la que pertenece, se pueden guardar empleados siempre y cuando estén asociados a una sucursal existente, además al registrar un empleado se le genera un identificador. La información de una sucursal consistirá en el nombre y su identificador.

El programa permite agregar, eliminar y modificar sucursales o empleados, además tenemos la opción de buscar una sucursal mediante el identificador de un empleado.

Para lograr esto se decidió modelar el programa con las siguientes clases:

### 1. Sucursal

La clase Sucursal simula la entidad sucursal, esta clase implementa la interfaz CSV. El constructor correspondiente a esta clase es **Sucursal** que recibe un String que representa el nombre del objeto Sucursal y en dicho constructor inicializa un identificador cuyo valor depende de la variable estática `proxNumSucursal` que se incrementa automáticamente en uno al crear una nueva instancia. En seguida se mencionan los principales métodos de esta clase:

- `toCSV()`. Este método permite regresar una representación de la instancia Sucursal en formato CSV.

### 2. Empleado

Al igual que la clase Sucursal, la clase Empleado modela la entidad empleado, también implementa la interfaz CSV. El constructor para esta clase recibe varios parámetros relacionados con la información básica de un empleado, de igual forma se inicializa un identificador único a partir de la variable estática `proxNumEmpleado` que se va incrementando en uno de forma automática al crear una nueva instancia. Los métodos principales para esta clase son:

- `toCSV()`. Método que regresa una representación en formato CSV de la instancia Empleado.

### 3. Manejador

La clase `Manejador`, se diseñó con el fin tener una clase que se encargue de administrar los datos del empleado y la sucursal por medio de tablas Hash que son una estructura de datos implementada en Java por la clase `HashMap`, por lo tanto se tendrá una tabla hash para empleados y otra para sucursales. Esta clase contiene los métodos principales de nuestro programa, que es el de agregar, eliminar, modificar y buscar por número de empleado.

- `Manejador()`. Corresponde al constructor de la clase que inicializa las tablas hash.
- `agregaSucursal()`. Agrega una instancia de `Sucursal` a la tabla hash sucursales.
- `agregaEmpleado()`. Agrega una instancia de `Empleado` a la tabla hash empleados.
- `eliminaSucursal()`. Elimina una sucursal usando el numero de identificador de la sucursal.
- `eliminaEmpleado()`. Elimina un empleado usando el numero de identificador del empleado.
- `modificaSucursal()`. Este método modifica el nombre de la sucursal.
- `modificaEmpleado()`. En este método se puede modificar alguno de los atributos de la entidad empleado utilizando el número de identificador del empleado.
- `buscaPorEmpleado()`. Este método regresará el número de sucursal utilizando el numero del empleado.

#### 4. Menu

Esta clase es necesaria para la interacción entre el usuario y la aplicación desde la terminal.

- `guardaBD()`. Guarda la nueva información proporcionada en los archivos de la base de datos.
- `leerBD()`. Carga la información de la base de datos en dos tablas hash para después asignárselas a un objeto de la clase `Manejador`.
- `iniciaMenu()`. Este método lee los archivos de la base de datos, después inicia el menú principal y guarda la información.
- `menuPrincipal()`. El método se encarga de darle la bienvenida al usuario pidiéndole información acerca de que entidad quiere consultar (`Sucursal` o `Empleado`).
- `menuAcciones()`. Este método recibe el parámetro que se introdujo en el menú principal y despliega un menú de opciones acerca de las acciones que desea realizar, ya sea el de agregar, eliminar, modificar o buscar.
- `menuElimina()`. Recibe el parámetro proporcionado en el menú principal para poder eliminar la información de una sucursal o de un empleado, para eso es necesario pedirle al usuario el numero de sucursal o el numero de empleado respectivamente.
- `menuAgrega()`. Se le pide al usuario que ingrese la información con un formato específico para poder agregar una nueva instancia de la entidad deseada.
- `menuModifica()`. El usuario debe elegir que atributo desea modificar ingresando el número de identificador de acuerdo a la entidad elegida e ingresar la nueva información.
- `menuBuscar()`. Simplemente se muestra un mensaje que le pide al usuario que ingrese el numero de empleado para después regresar el número de sucursal a la que pertenece.

#### 5. IO

Esta clase se encarga de leer y escribir los archivos u objetos en formato CSV. Por lo tanto existen dos métodos. Esta el método `leer` que lee un archivo CSV y devuelve una tabla hash que contiene la información y también se encuentra el método `escribir` que escribe un archivo CSV con los datos proporcionados y los guarda.

Se menciona 5 diferencias entre almacenar la información mediante un sistema de archivos a almacenarla en una base de datos:

- Mientras mas entidades se tengan mas complicado se vuelve el manejo de las referencias.
- Ineficiencia para realizar consultas.