



Universidad Libre de Colombia

Programación II

Carlos Leonardo Adames Camargo

Diego Alejandro Aguilar Cárdenas

Santiago Andres Baquero Cortes

Miguel Felipe Cano Bogotá

Hugo Néstor Herrera Arias

Brayan Ricardo Moreno Espinosa

Bogotá D.C

Tabla de Contenidos

Tabla de Contenidos	2
Introducción.....	2
Objetivos.	3
Objetivo General.....	3
Objetivos Específicos.....	3
Marco Teórico	3
Explicación del código.....	6
Conclusiones.....	9
Referencias	10

Introducción.

Las bases de datos son el método preferido para el almacenamiento estructurado de datos, desde las grandes aplicaciones multiusuario, hasta los teléfonos móviles y las agendas electrónicas utilizan tecnología de bases de datos para asegurar la integridad de los datos y facilitar la labor tanto de usuarios como de los programadores que las desarrollaron.

Desde la realización del primer modelo de datos, pasando por la administración del sistema gestor, hasta llegar al desarrollo de la aplicación, los conceptos y la tecnología asociados son muchos y muy heterogéneos. Sin embargo, es imprescindible conocer los aspectos clave de cada uno de estos temas para tener éxito en cualquier proyecto que implique trabajar con bases de datos

Las excepciones son situaciones anómalas que pueden ocurrir durante la ejecución de las aplicaciones, como, por ejemplo, acceder a una posición de un vector fuera rango. En algunos casos las excepciones no se podrán gestionar, por ejemplo, los de la propia JVM, y en otros casos sí.

Java proporciona un mecanismo de gestión de excepciones en los casos en los que estemos obligados a gestionarlas. En caso de que no queramos gestionar algún tipo de

excepción estaremos obligados a indicarlo explícitamente. No podremos compilar el código a menos que gestionemos la excepción, o indiquemos explícitamente que no lo queremos hacer.

La definición de excepciones propias se hace a través del mecanismo de extensión de clases, la Herencia.

Objetivos.

Objetivo General.

- Adquirir las habilidades y conocimientos de MySQL necesarios para utilizar el programa en el entorno necesario.

Objetivos Específicos.

- Ser capaz de plantear cualquier tipo de consulta a la base de datos.
- Conocer las posibilidades que Java ofrecen para la conexión y uso de bases de datos en general de MySQL y exportación de archivos tipo .txt
- Ser capaz de realizar el modelo de una base de datos relacional, a partir de la especificación de requerimientos de un proyecto, comprendiendo y aplicando los conceptos y transformaciones implicados.

Marco Teórico

El acceso a bases de datos desde Java se realiza mediante el estándar JDBC (Java data base connectivity), que permite un acceso uniforme a las bases de datos independientemente del SGBD. De esta manera, las aplicaciones escritas en Java no necesitan conocer las especificaciones de un SGBD en particular, basta con comprender el funcionamiento de JDBC. Cada SGBD que quiera utilizarse con JDBC debe contar con un adaptador o controlador.

La característica relacional más importante de SQL es que permite acceder a los datos sin necesidad de especificar cómo se ha de realizar dicho acceso permitiendo así la "navegación automática" por los datos. SQL está diseñado para responder a preguntas del tipo Qué quiero hacer al contrario que los otros lenguajes de programación, como Pascal, Java y C++, que intentan resolver cuestiones del tipo Cómo lo hago?

SQL usa los términos tabla, fila y columna para relación, tupla y atributo, respectivamente.

SQL significa Lenguaje Estructurado de Consultas (Structured Query Language) y se ha establecido Como el lenguaje estándar de bases de datos relacionales, esto significa que su uso esta generalizado a nivel internacional. (Fuentes, 2013)

Java Swing, que permitieron hacer la conexión con la base de datos y una interfaz gráfica que facilitara la selección de la acción que va a ejecutar.

Librería Java SQL: Hace referencia a una paquetería de java que permite y facilita la consulta e interacción con una base de datos; cuenta con sus propias clases y estructura (Belmont, 2011).

Librería Java Swing: Es un paquete de java que hace parte de Java Foundation Classes (JFC), que suministra una serie de herramientas necesarias para la construcción de interfaces graficas de Usuario (Henao, 2013).

De igual manera, se implementó excepciones de java, que permiten la generación de métodos que informan acerca de algún error en lo invocado por el código o situación anormal que se produzca durante la ejecución (Walton, 2020). Por otra parte, para el desarrollo de la base de datos se elaboró en MySQL, un sistema de gestión de bases de datos del tipo relacional, caracterizada por su funcionamiento cliente y servidor, lo cual permite una continua comunicación con el servidor facilitando acciones como consultar, modificar, guardar o generar nuevas tablas (Robledano, 2019).

Base de datos relacional: Es un tipo de base de datos que organiza, almacena y da acceso a datos relacionados entre sí, haciendo uso de tablas para la organización de dichos datos

Libreria File Writer y File Reader:

```
FileWriter fw=new FileWriter("D:\\Prueba.txt");
```

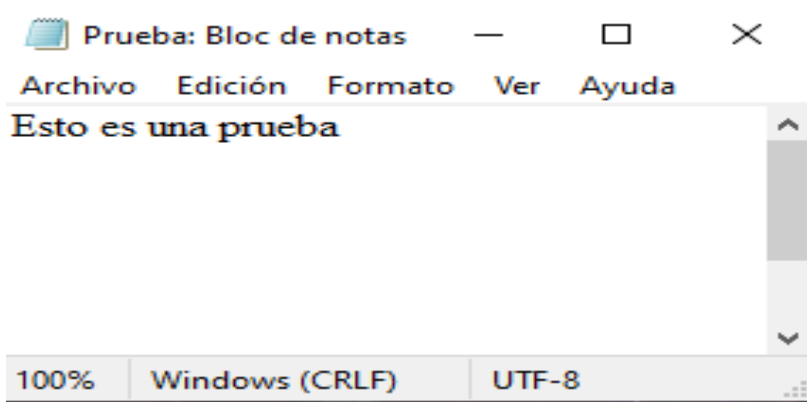
Esta línea permite determinar la ruta en la que se encuentra el archivo que se crea este será modificado de acuerdo con la necesidad del trabajo.

```
fw.write("Esto es una prueba");
```

Esta línea permite modificar el contenido del archivo .txt en este caso el archivo presentara el contenido de ("Esto es una prueba") , esto haciendo referencia al Write que es de escribir.

```
FileReader fr=new FileReader("D:\\fichero1.txt");
```

El file Reader permite la lectura del documento como indica el Reader significa leer.



Conexión

En este se realiza toda la conexión del programa con la base de datos usando de forma externa la siguiente librería.



En este se ingresan los datos de la base de datos para que el programa conozca la ruta correspondiente también contraseña si es necesaria para acceder a ella.

```
static String bd = "registro";
static String login = "root";
static String password = "mortadelal0";
static String url = "jdbc:mysql://localhost:3306/"+bd;
```

Explicación del código

Para el desarrollo de este código se reutilizo el algoritmo del informe realizado anteriormente, en el cual se establecía una conexión con una base de datos desarrollada en MySQL, que permitía el registro, búsqueda y eliminación de cualquier dato alojado en la base de datos, por esta razón solo fue necesario realizar una modificación a dicho código la cual fue:

Se creo un botón en la interfaz de ventana registro para que se ejecutara la función “actionPerformed” de dicho botón llamado Escribir en el fichero y el cual esta parametrizado con la variable “e”.

```
botonEscribir = new JButton("Escribir en el Fichero");  
botonEscribir.addActionListener(new ActionListener() {  
public void actionPerformed(ActionEvent e) {  
  
String codigo = textCod.getText();  
String nombre = textNombre.getText();  
String edad = textEdad.getText();  
String telefono = textTelefono.getText();  
String profesion = textProfesion.getText();  
textArea.setText("Codigo:" + codigo + "Nombre:" + nombre + "Edad:" + edad + "Telefono:" + tel  
efono + "Profesion:" + profesion);  
String pasareltextarea = textArea.getText();  
File fichero = new File("archivo.txt");  
try {  
BufferedWriter fichero2 = new BufferedWriter(new FileWriter(fichero));  
fichero2.append("codigo" + codigo);  
fichero2.newLine();  
fichero2.append("Nombre" + nombre);  
fichero2.newLine();
```

```

fichero2.append("Edad"+edad);

fichero2.newLine();

fichero2.append("Telefono"+telefono);

fichero2.newLine();

fichero2.append("Profesion"+profesion);

fichero2.newLine();

fichero2.append("El text Area"+pasareltextarea);

fichero2.newLine();

fichero2.close();

}catch(IOException ex) {

Logger.getLogger(VentanaRegistro.class.getName()).log(Level.SEVERE, null, ex);

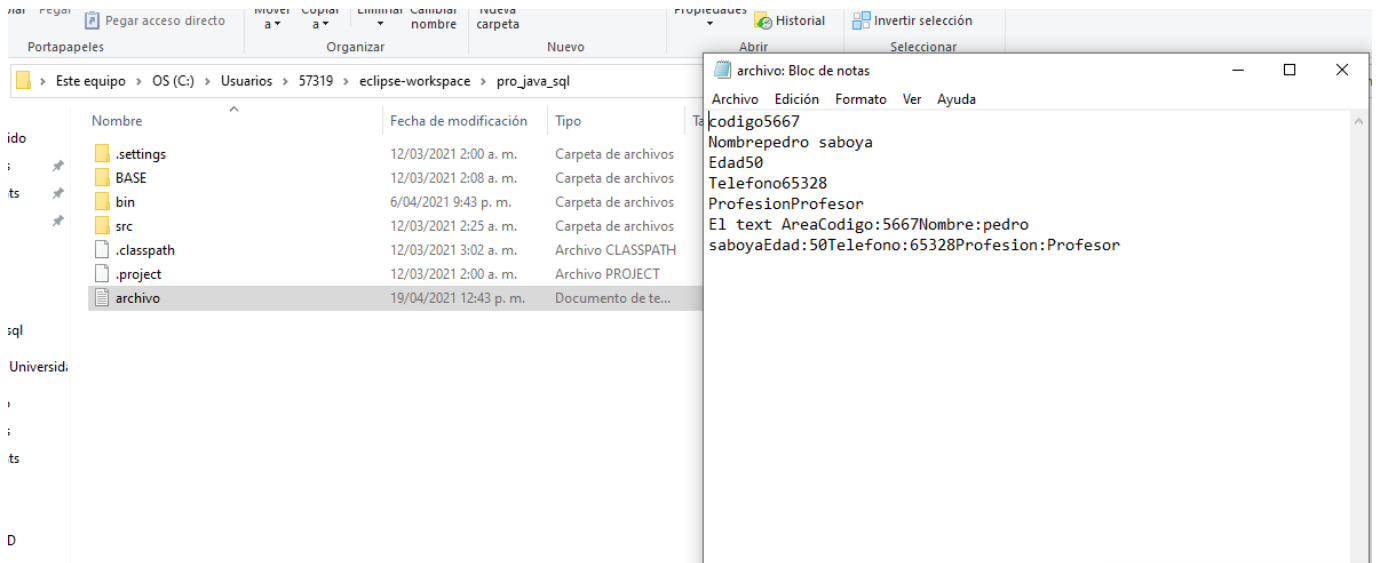
}

}

});

```

Como se observa en el fragmento tomado del código, si inicio con la creación del botón mencionado, luego se realizó la función en la cual se escribe la función que desempeñaría, inicialmente se le ordena que todos lo que se encuentren digitado en los *JTextField* se tomen como una cadena de caracteres, luego con el *textArea.setText* se recopila la información de las cadenas caracteres para colocarlo en un *textArea* que luego funcionara para poder convertir toda la información en una única cadena de caracteres, que facilitara la exportación de los datos al archivo de extensión .txt, luego de ello con la sentencia *File fichero= new File("archivo.txt");* se inicializa el objeto que no permitirá la creación del archivo en el almacenamiento del PC, luego con ayuda de un *try* se le ordeno que inicializara la sentencia *BufferedWriter fichero2=new BufferedWriter(new FileWriter(fichero));* la cual permite un almacenamiento de buffer más amplio evitando que se tengan que hacer un vínculo con la memoria por cada línea de texto, por ultimo con ayuda de la sentencia *fichero2.append* se le ordena que escriba en el archivo de texto la información obtenida del registro y las cadenas de caracteres que anteriormente se elaboraron, luego de ello con las sentencias *fichero2.newLine();* se organizaba la información con saltos de línea que permitieran una mejor visualización en el archivo de texto, por último, se finaliza la ejecución con el comando *fichero2.close();* , con esto el programa generara el archivo en la carpeta ubicada en C:\Users\57319\eclipse-workspace\pro_java_sql, la cual es la carpeta de instalación del ambiente de desarrollo en cual se elaboró el programa.



Por otro lado, se agregó un *JTextArea* en el cual se tuviera una previsualización de los datos que se escribirían en el archivo de texto el cual se activa al dar clic en el botón mostrar información.

```

botonMostrar = new JButton("Mostrar Informaci\u00F3n");

botonMostrar.addActionListener(new ActionListener() {

    public void actionPerformed(ActionEvent e) {

        String codigo = textCod.getText();

        String nombre = textNombre.getText();

        String edad = textEdad.getText();

        String telefono = textTelefono.getText();

        String profesion = textProfesion.getText();

        textArea.setText("Codigo:" + codigo + "\n" + "Nombre:" + nombre + "\n" + "Edad:" + edad + "\n"
            + "Telefono:" + telefono + "\n" + "profesion:" + profesion);

    }
}

```



```
});
```

```
botonMostrar.setBounds(20, 270, 150, 25);
```

```
getContentPane().add(botonMostrar);
```

registro : Proyecto_JAVA_SQL/MVC

REGISTRO DE PERSONAS

Codigo: 5667

Nombre: pedro saboya Edad: 50

Profesion: Profesor telefono: 65328

Registrar Cancelar

Mostrar Información

Escribir en el Fiche...

Codigo:5667
Nombre:pedro saboya
Edad:50
Telefono:65328
profesion:Profesor

Conclusiones.

- Se desarrollo el crecimiento de capacidades en programación gracias a la implementación de la búsqueda en MYSQL y la exportación a un archivo .txt
- Tener más conocimiento en la clase FileWriter y BufferedWriter las cuales se utilizan para la creación del archivo .txt y modificación del contenido del mismo.
- De igual manera el programa mejoró nuestra parte lógica y se pone en relación directa lo que es bases de datos con programación. A su vez manejar sintaxis y aplicarla a casos reales de casos de negocios básicos y/o estándar.
- La importancia de los datos está en su capacidad de asociarse dentro de un contexto para convertirse en información. Por si mismos los datos no tienen capacidad de

comunicar un significado y por tanto no pueden afectar el comportamiento de quien los recibe. Para ser útiles, los datos deben convertirse en información para ofrecer un significado, conocimiento, ideas o conclusiones

Referencias

Fuentes, D. M. (02 de 2013). Notas_del_curso_Bases_de_Datos.pdf. Obtenido de http://www.cua.uam.mx/pdfs/conoce/libroselec/Notas_del_curso_Bases_de_Datos.pdf

Belmont, S. (26 de mayo de 2011). Slideshare. Obtenido de Slideshare:

<https://es.slideshare.net/SonnyBelmont/javasql/>

Henao, C. D. (11 de agosto de 2013). CoDejavu. Obtenido de CoDejavu:

<https://codejavu.blogspot.com/2013/08/que-es-java-swing.html> Oracle. (s.f.). Oracle.

Obtenido de Oracle: <https://www.oracle.com/co/database/what-is-a-relational-database/>

Robledano, Á. (24 de septiembre de 2019). OpenWebinars. Obtenido de OpenWebinars:

<https://openwebinars.net/blog/que-es-mysql/>

Walton, A. (4 de junio de 2020). Java Desde Cero. Obtenido de Java Desde Cero:

<https://javadesdecero.es/intermedio/manejo-de-excepciones/>

Eclipse. (s.f.). Obtenido de Eclipse: <https://www.eclipse.org/windowbuilder/> LenguajeSQL.

(s.f.). Obtenido de LenguajeSQL: <https://www.lenguajesql.com.ar/default.asp?id=1>

Universidad nacional del Sur. (s.f.). Departamento de Ciencias e Ingeniería de la

Computación. Obtenido de Departamento de Ciencias e Ingeniería de la Computación:

<https://cs.uns.edu.ar/~drg/bd/downloads/Clases%20Practicas/Java%20y%20MySQL/Clase-Java-MySQL.pdf>

