

Actividad | #3 | Programa Banco Mexicano Lenguajes de Programación IV

Ingeniería en Desarrollo de Software

TUTOR: Aarón Iván Salazar Macías

ALUMNO: Erick Alfredo Quiroz Figueroa

FECHA: 08/08/2025

Contenido

[**Introducción** 3](#_Toc205567274)

[**Descripción** 4](#_Toc205567275)

[**Justificación** 5](#_Toc205567276)

[**Desarrollo** 6](#_Toc205567277)

[**Interfaz** 6](#_Toc205567278)

[**Codificación** 8](#_Toc205567279)

[**Conclusión** 17](#_Toc205567280)

# **Introducción**

Como sabemos la programación en Java es una de las mas importantes hoy en día ya que gracias a las tecnologías Java fue el primer lenguaje en usar API en sus aplicaciones, y, por consiguiente, el primero en usar al 100% el paradigma orientado a objetos ya que si nos basamos en la historia el primero en usarlo fue C++ con la utilización de clases y objetos y todo lo que conlleva dicho paradigma.

Pero Java utilizo una interfaz única para poder utilizar los componentes de un programa de una manera mas sencilla creando una IDE exclusiva para su programación eficiente.

Para que un programa sea ejecutado correctamente en nuestro equipo es necesario correrlo en la maquina virtual de Java que por sus siglas en ingles podemos encontrarlo como JVM y se puede instalar en un paquete que la misma empresa proporciona para que sea fácil usar el intérprete.

Como todos los lenguajes de programación que existen podemos encontrar un kit con las librerías necesarias en el lenguaje Java para que las funciones y métodos a utilizar en la codificación hagan funcionar el programa sin ningún error en él.

# **Descripción**

Como pudimos observar en la actividad anterior, creamos una aplicación en el lenguaje Java en la cual hicimos la simulación de un cajero automático, el cual hace los siguientes movimientos: “Deposito”, “Retiro”, “Consulta de Saldo” y “Salir”. En cada una de las opciones programamos la opción de que al ingresar una cantidad nos mandara una caja de texto en la que nos dijera o nos preguntara si queríamos realizar otro deposito o retiro según sea el caso, y conforme a lo elegido pasar al siguiente movimiento.

Ahora dejando en la aplicación esos eventos programados, realizaremos su complemento que es realizar la conexión con un gestor de base de datos, en este caso con MySQL para poder guardar las cantidades ingresadas y se reflejen en la aplicación o en este caso en la tabla de la Base de Datos creada.

Al momento de ingresar la cantidad nos tendrá que arrojar el mensaje de que el depósito fue exitoso caso contrario será que hubo algún error, de igual manera sucederá el mismo escenario con la opción de retiro.

# **Justificación**

Como sabemos la complejidad de un sistema determina que tan bien este estructurado y organizado el código que hay detrás de este, ya que esto nos permite observar e interactuar de manera sencilla con el programa.

El motivo de estas actividades es poder conocer un poco mas a fondo como es que se crea la interfaz de un programa y como es que se puede programar dándole un funcionamiento optimo empezando por la simplicidad del código, dándole una estructura y entendimiento lo mas sencillo que se pueda realizar.

Sabemos que a parte de Java existen cientos de lenguajes de alto nivel que nos permiten crear interfaces graficas de manera sencilla y se pueden hacer conexiones entre dichos lenguajes para tener un mayor control sobre el sistema.

Dicho esto, pasaremos a terminar y realizar la actividad.

# **Desarrollo**

## **Interfaz**

Interfaz gráfica principal del programa con las opciones a utilizar.

Interfaz de usuario gráfica, Texto, Aplicación, Chat o mensaje de texto

El contenido generado por IA puede ser incorrecto.

Interfaz de “Depósitos”.

Interfaz de usuario gráfica, Aplicación

El contenido generado por IA puede ser incorrecto.

Interfaz gráfica de “Retiros”.

Interfaz de usuario gráfica, Aplicación

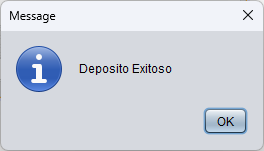
El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaces de mensajes de confirmación de “Deposito” y “Retiro”.

Interfaz de usuario gráfica, Texto, Aplicación

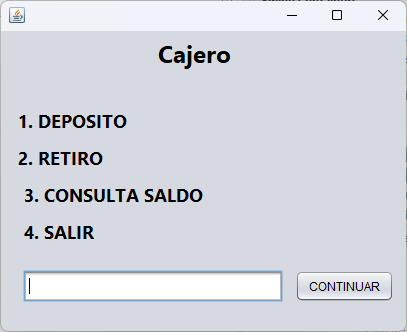
El contenido generado por IA puede ser incorrecto.

Interfaz de usuario gráfica, Texto, Aplicación

El contenido generado por IA puede ser incorrecto.Interfaces de “Deposito” y “Retiro” exitosos. Por alguna extraña razón no me aparecen las interfaces de error.

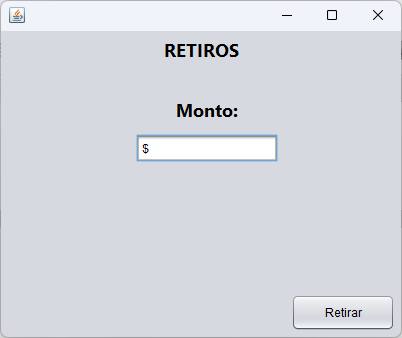
## **Codificación**

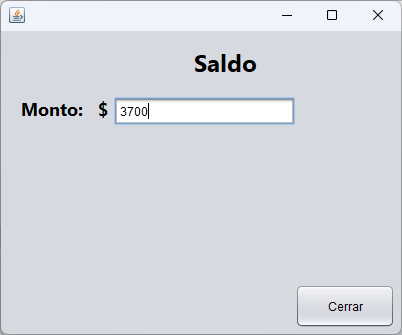
Como punto de partida tenemos la interfaz principal del “Cajero” con las diferentes opciones enumeradas.

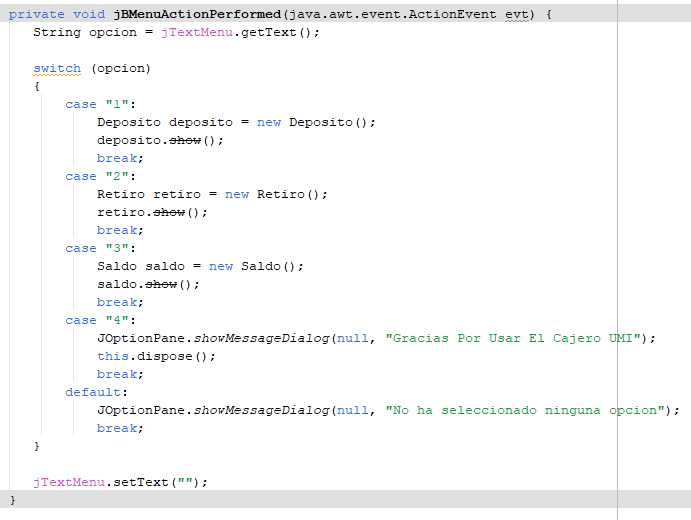


Y de aquí se despliegan las interfaces siguientes.

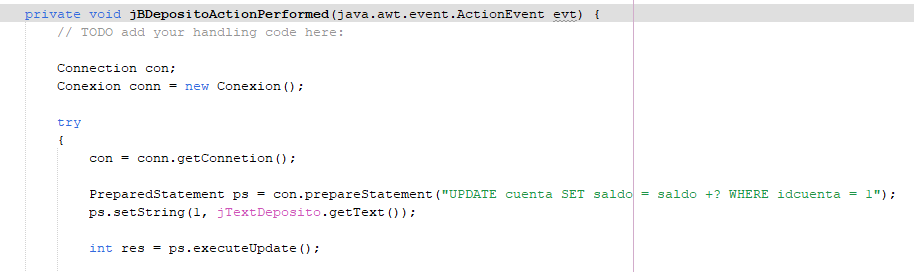






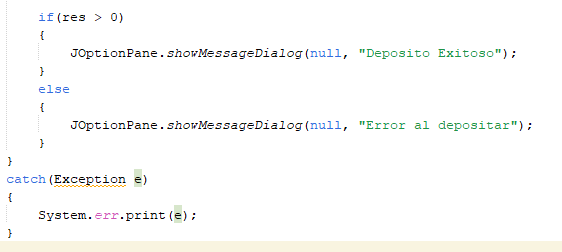
Como es primordial veremos el código de nuestra interfaz principal que nos ayudara a saltar a la siguiente interfaz poniendo el numero de la opción de la siguiente manera:

Para que nuestra interfaz de “Depósitos” funcione tenemos que codificar lo siguiente:



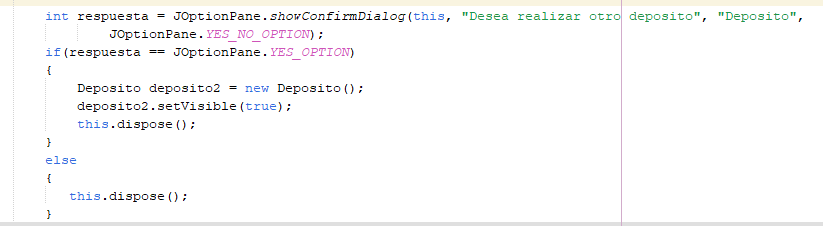
En esta parte del código hacemos la conexión con la Base de Datos usando el comando *UPDATE – SET* el cual nos permite actualizar la información ingresada para reflejarla en la base de datos hecha en MySQL.

A continuación, tenemos el siguiente código:



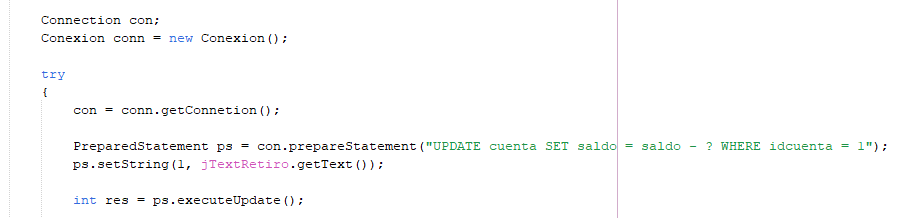
En este fragmento mostramos el uso de nuestra condicional *IF – ELSE* el cual nos ayuda a ver que acciones tomar si se cumple una opcion y que pasa si no se cumple del todo.

Despues, tenemos lo siguiente:



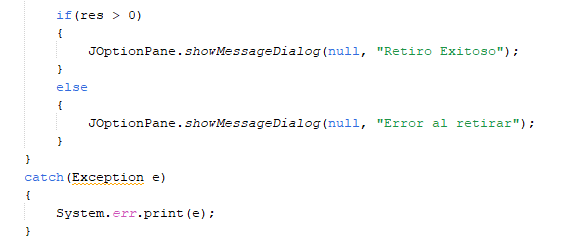
Este fragmento al mismo tiempo que muestra el uso de una condicional nos ayuda a ver que se manda llamar un Panel de opciones de Si – No para ver si se realiza otro deposito o no, esto después de haber realizado el depósito.

Después pasamos a la interfaz de “Retiro” donde los códigos tienen la misma función solo que con variables y funciones diferentes a la parte de la interfaz de “Deposito”.

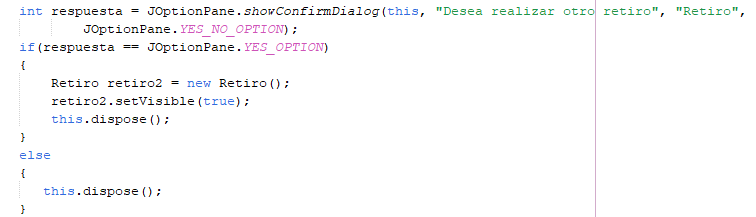


Esta es la parte de la conexión.

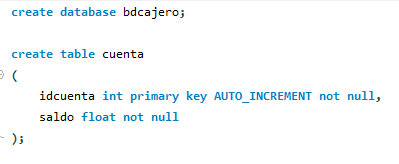
Después tenemos la parte de código que pertenece a la los mensajes de situación del retiro, si este fue exitoso o no.



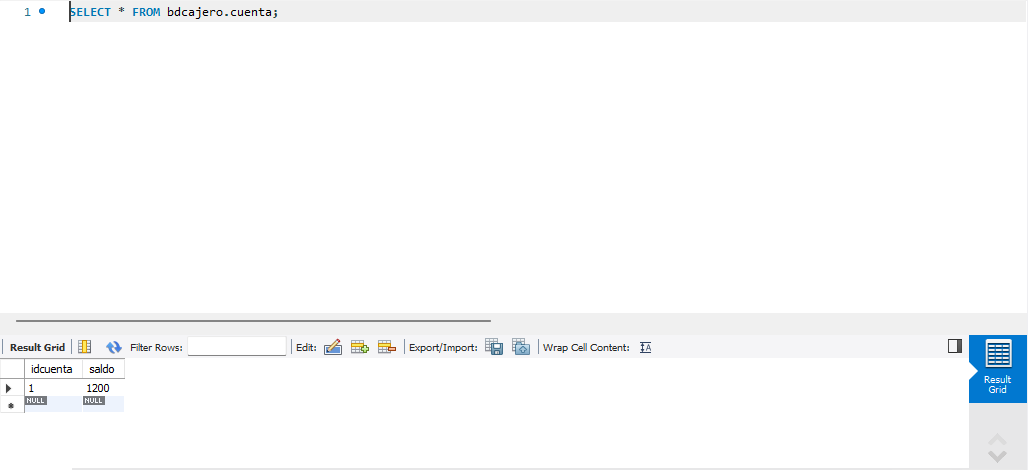
Por último, tememos la parte del código donde se nos pregunta si se realizara otro retiro o no.



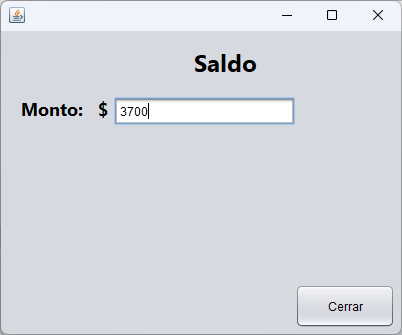
Para la consulta de saldo tenemos que hacer la creación de la BD en MySQL lo cual el código quedaría de la siguiente manera:



Y ya corriendo o haciendo la consulta de nuestra BD quedaría algo así:



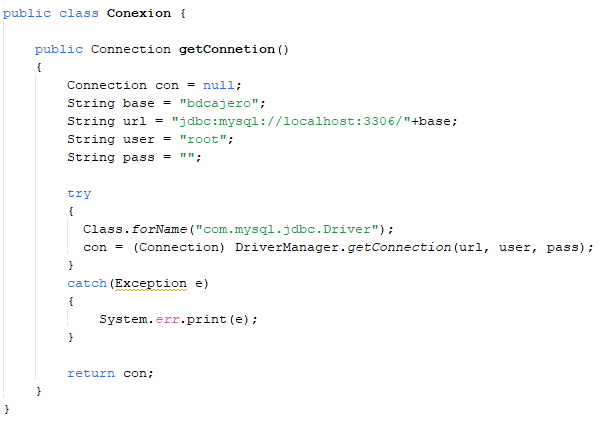
Para que nuestra interfaz de saldo salga con datos ingresados de esta manera:



Es necesario generar y programar el código como se muestra a continuación:



Bien, para que todo esto funcione y la conexión sea exitosa necesitamos crear la conexión de Java con MySQL y para ello creamos una clase llamada “Conexión” que nos ayudara con dicha tarea y se muestra de la siguiente manera:



# **Conclusión**

Para terminar, podemos decir que el saber estructurar y darle su lugar a cada línea de código es importante ya que nos ayuda a identificar de manera sencilla que es lo que hace y como es que este funciona, que parte del código funciona primero y cual le sucede después, para ello, en ocasiones es importante poner comentarios para separar ya que sin estos podemos confundirnos y hasta perdernos en el código.

Saber programar estos sistemas, nos ayudan a despertar nuestra lógica y ver cómo es que podríamos darle solución a los distintos problemas que los sistemas presentan a lo largo del tiempo, siempre y cuando nos haga ahorrar recursos y tiempo para no hacerlo tedioso y costoso.

Analizar y comprender un lenguaje como tal nos ayuda a verificar y conocer como es que podemos dar solución al sistema y así poder corregir de manera rápida la parte del sistema que nos este causando problemas en el momento.