

DOCUMENTACIÓN PROYECTO

UNIDAD 1

La aplicación funciona mediante una base de datos hsqldb. Puedes interactuar con la tabla de Pedidos y de Clientes de la base de datos mediante la aplicación, añadiendo, modificando, buscando y eliminando.

Estas acciones son accesibles a través del menú “Acciones”.

**Los formularios de buscar y añadir están separados del JFrame, en archivos JPanel.*

Secciones (está explicado para la pedidos, pero en clientes es prácticamente lo mismo):

- **Buscar/filtrar:**

Se agrega el JPanel de buscar pedido y se carga la tabla específica para buscar mediante un método

Realiza una consulta a la base de datos, haciendo un SELECT de los campos más importantes del pedido y si rellenas algún componente, como un jTextField, añade al SELECT una condición con AND y cuando pulsas el botón “Buscar” realiza la búsqueda con condiciones.

Columnas para manejar el pedido (Detalles, Eliminar, Modificar)

Añades a la tabla un evento de MouseAdapter y añades el método mouseClicked. Con una consulta SELECT y utilizando los métodos getValueAt(row, column) donde esté la fecha del pedido y el nombre de la compañía cliente, sacamos el código del pedido. Mediante los métodos rowAtPoint(evt.getPoint()) y columnAtPoint(evt.getPoint()) , sacas en que fila y columna se ha hecho click. Creas las condiciones para que cuando hagas click en las columnas de manejos del pedido suceda algo:

-Detalles:

Se crea un objeto del JDialog de detalles, y con el método setVisible(true) lo activas. En este JDialog se realiza un SELECT más extendido, con todos los campos de la tabla pedidos (Exceptuando el código del pedido) y de la tabla DetallesPedido.

-Eliminar:

Aparece un JOptionPane de confirmación, y si se pulsa “sí”, se ejecuta una consulta DELETE para borrar el pedido mediante el código del pedido.

-Modificar:

Se crea un objeto del JDialog de modificar, y con el método setVisible(true) lo activas. Aquí se rellenan automáticamente los componentes del formulario mediante el código del pedido, ejecutando una consulta "SELECT" con un "WHERE codigo = codigoPedido". Cuando le das al botón de modificar, captura todos los datos y si están todos rellenos, ejecuta un "UPDATE SET" de los campos con un PreparedStatement.

● Añadir:

Se agrega el JPanel de añadir pedido y se carga la tabla específica para añadir mediante un método.

Una vez rellenas todos los campos para añadir un pedido, ejecuta una consulta "INSERT INTO" que recoge toda la información de los componentes para crear un pedido y lo ingresa en la tabla pedidos.

JTextField para elegir un cliente:

Mediante el método "setEditable(false)" haces que no puedan escribir en el JTextField. Añades un evento de MouseAdapter y agregas el método MouseClicked. Cuando se pulse el JTextField, se abrirá el JDialog con el formulario y la tabla de clientes. Dentro de este JDialog se añade otro evento, MouseClicked también, para que cuando elijas una fila de la tabla clientes, se seleccione y se agregue al JTextField de cliente.

Problema solucionado:

-Acumulación de eventos en la tabla. Cada vez que se cargaba la tabla de buscar,(por ejemplo, cuando en el formulario de búsqueda le das al botón reiniciar, este borra lo que hay en los componentes y llama al método cargarTablaBuscar para que salga la tabla completa), se añadía un evento. Entonces cuando ibas a darle a detalles, eliminar o modificar, se abría varias veces la ventana emergente o JDialog.

Solución: crear un atributo en el JFrame,
"private MouseListener "mouseClickListener = null"
y usar el método en la tabla "removeMouseListener(mouseClickListener)" para quitar el evento y "addMouseListener(mouseClickListener)" para añadirlo, esto cada vez que se tenga que cargar la tabla. De esta manera se elimina la acumulación de eventos.