

UNIVERSIDAD DE SAN CARLOS DE GUATEMALA
CENTRO UNIVERSITARIO DE OCCIDENTE
DIVISION DE CIENCIA DE LA INGENIERIA
AREA DE SISTEMAS
INGENIERO PEDRO LUIS DOMINGO VÁSQUEZ
MANUAL TECNICO



LUIS ANTONIO MONTERROSO GUZMAN
202031794
QUETZALTENANGO, MARZO 29 DEL 2023

Clase App

- **Static void main(String[] args)**
Acciona el método start y obtiene una conexión con la base de datos.
- **Void start(javafx.stage.Stage primaryStage)**
Inicia la pantalla inicial.

Clase Conector

- **Boolean conectar()**
Establece la conexión con la base de datos en PostgreSQL.

Clase Starter

- **Static void main(String[] args)**
Inicia la app JavaFX.

Interface ConstructorDeObjeto

- **Object construirLista(ResultSet resultado)**
Los métodos deberán ser capaces de construir una lista de objetos a partir de un ResultSet.
- **Object construirObjeto(ResultSet resultado)**
Los métodos deberán ser capaces de construir un objeto a partir de un ResultSet.

Clase ConstructorDeObjetoBodegaDTO

- **javafx.collections.ObservableList<BodegaDTO> construirLista(ResultSet resultado)**
Crea una lista de objetos BodegaDTO a partir de un ResultSet.
- **BodegaDTO construirObjeto(ResultSet resultado)**
Crea un objeto BodegaDTO a partir de un ResultSet.

Clase ConstructorDeObjetoCliente

- **javafx.collections.ObservableList<Cliente> construirLista(ResultSet resultado)**
Crea una lista de objetos Cliente a partir de un ResultSet.
- **Cliente construirObjeto(ResultSet resultado)**
Crea un objeto Cliente a partir de un ResultSet.

Clase ConstructorDeObjetoClienteMasGanancia

- **javafx.collections.ObservableList<ClientesMasGananciaDTO> construirLista(ResultSet resultado)**
Crea una lista de objetos ClientesMasGanancia a partir de un ResultSet.
- **ClientesMasGananciaDTO construirObjeto(ResultSet resultado)**
Crea un objeto ClientesMasGanancia a partir de un ResultSet.

Clase ConstructorDeObjetoCompra

- **javafx.collections.ObservableList<Compra> construirLista(ResultSet resultado)**
Crea una lista de objetos Compra a partir de un ResultSet.
- **Compra construirObjeto(ResultSet resultado)**
Crea un objeto Compra a partir de un ResultSet.

Clase BodegaDTORepository

- **javafx.collections.ObservableList<BodegaDTO> buscarEnBodegaPorCodigoBarras(String codigo)**
Se comunica a la base de datos y obtiene el producto en la bodega segun su codigo de barras.
- **Boolean distribuirBodegaASucursal(BodegaDTO bodega, String sucursal)**
Se comunica a la base de datos, elimina un producto de la bodega y lo agrega a el inventario de una sucursal.
- **javafx.collections.ObservableList<BodegaDTO> mostrarBodega()**
Se comunica a la base de datos y obtiene el inventario de bodega.

Clase ClienteRepository

- **Cliente buscarClientePorNit(long nit)**

Se conecta a la base de datos y busca un cliente por su nit en la tabla Cliente.

- **javafx.collections.ObservableList<Cliente> buscarClientesPorNombre(String nombre)**

Se conecta a la base de datos y busca un cliente por su nit en la tabla Cliente.

- **Boolean editarCliente(Cliente cliente, long nitAnterior)**

Edita un cliente en la base de datos a partir de su antiguo nit.

- **javafx.collections.ObservableList<Cliente> mostrarClientes()**

Busca a todos los clientes registrados en la base de datos a excepción del cliente CF.

Clase CompraRepository

- **Compra buscarCompraPorCodigo(int codigoBarras)**

Busca una compra en la base de datos según un código de barras.

- **Boolean insertarCompraEnBodega(Compra compra)**

En la base de datos, crea una nueva compra, la inserta y luego la asocia a la bodega.

- **Boolean modificarCompra(Compra compra, int codigoAnterior)**

Modifica una tupla de la tabla Compra.

Clase EmpleadoRepository

- **javafx.collections.ObservableList<Empleado> buscarEmpleadoPorNombre(String nombre)**

Se comunica con la base de datos y obtiene los empleados que cumplan con la búsqueda.

- **Boolean insertarEmpleado(Empleado empleado)**

Se comunica con la base de datos e inserta un nuevo empleado a partir del objeto Empleado enviado como parámetro.

- **Empleado traerEmpleadoPorUsuarioYPasswd(Empleado empleado)**

Se comunica con la base de datos y obtiene un Empleado a partir de su usuario y password.

- **javafx.collections.ObservableList<Empleado> traerEmpleados()**

Se comunica con la base de datos y obtiene todos los empleados.

Clase ProductoDTORepository

- **javafx.collections.ObservableList<ProductoDTO> buscarProductoPorCodigoDeBarras(String codigoBarras)**

Se comunica a la base de datos y obtiene el producto segun el codigo de barras.

- **javafx.collections.ObservableList<ProductoDTO> buscarProductoPorCodigoDeBarrasYSucursal(String sucursal, String codigoBarras)**

Se comunica a la base de datos y obtiene el producto segun la sucursal y el codigo de barras.

- **Boolean cambiarProductoDeSucursal(ProductoDTO producto, String sucursal)**

Edita la sucursal en donde un producto esta.

- **javafx.collections.ObservableList<ProductoDTO> mostrarProductosDtoDeSucursales()**

Se comunica a la base de datos y obtiene el inventario de todas las sucursales.

- **javafx.collections.ObservableList<ProductoDTO> mostrarProductosDtoPorSucursal(String sucursal)**

Se comunica a la base de datos y obtiene el inventario de una sucursal.

Clase ProductoRepository

- **javafx.collections.ObservableList<Producto> buscarProducto(String nombre)**

Se comunica con la base de datos para buscar un producto por su nombre.

- **Boolean insertarProducto(Producto producto)**

Se comunica a la base de datos e inserta un producto nuevo a la base de datos.

- **javafx.collections.ObservableList<Producto> mostrarProductos()**

Se comunica con la base de datos para mostrar todos los productos del catálogo.

Clase VentaRepository

- **Boolean registrarVenta(Cliente cliente, Venta, javafx.collections.ObservableList<ProductoDTO> carrito)**

Registra una venta en la base de datos.

- **Venta traerUltimaCompraDeUnCliente(Cliente cliente)**

Trae la última compra de un cliente a partir de su nit.