PRIMERA PREGUNTA:

Crear la base de datos PagaYa con las siguientes tablas:

* Usuarios: id (pk), nombre, dirección, email, fecha\_registro, estado (activo/inactivo).
* Tarjetas: id (pk), numero\_tarjeta, fecha\_expiracion, cvv, tipo\_tarjeta (Visa, Mastercard)
* Productos: id (pk), codigo\_producto, nombre, categoría(celular/pc/televisor), porcentaje\_impuesto, precio
* Pagos. id(pk), código\_pago fecha, estado (exitoso/fallido), monto, producto\_id, tarjeta\_id, usuario\_id.
* Comprobantes\_pago\_xml: id (pk), detalle\_xml.
* Comprobantes\_pago\_json: id (pk), detalle\_json

Crear las siguientes funciones almacenadas con return query:

* Función almacenada: Obtener pagos de un usuario -> parámetros: usuario\_id, fecha. Retornar la tabla (código\_pago, nombre\_producto, monto, estado).
* Función almacenada: Obtener las tarjetas del usuario que han pagado un monto mayor a $1.000 -> parámetros usuario\_id. Retornar la tabla (nombre\_usuario, email, numero\_tarjeta, cvv, tipo\_tarjeta)

SEGUNDA PREGUNTA:

Con cursores realizar las siguientes funciones almacendas:

* Función almacenada usando cursores: obtener tarjetas con detalle de usuario: parámetro usuario\_id. Retornar los siguientes datos en un varchar numero\_tarjeta, fecha\_expiracion, nombre, email.
* Función almacenada usando cursores obtener pagos menores a $1.000 dada una fecha: parámetro fecha. Retornar los siguientes datos en un varchar: monto, estado, nombre\_producto, porcentaje\_impuesto, usuario\_direccion, email.

Crear los siguientes procedimientos almacenados para el campo xml y json de la tabla comprobantes de pago

* Procedimiento guardar\_xml: insertar el xml: <pago><codigoPago></codigoPago><nombreUsuario></nombreUsuario><numeroTarjeta></numeroTarjeta><nombreProducto></nombreProducto><montoPago></montoPago></pago>
* Procedimiento guardar\_json: insertar el json: {emailUsuario: “”, numeroTarjeta: “”, tipoTarjeta: “”, codigoProducto: “”, codigoPago: “”, montoPago: “”}

TERCERA PREGUNTA:

* Crear un before insert trigger llamado validaciones\_producto donde se debe validar que el precio debe ser mayor a 0 y menor a $20.000, el porcentaje debe ser mayor a 1% y menor o igual a 20%
* Crear un after insert trigger llamado trigger\_xml que permita almacenar el xml y el json posterior al insertar un registro en la tabla pagos

CUARTA PREGUNTA.

Crear secuencias para los siguientes campos:

* El código de producto debe iniciar desde el 5 e incrementar de 5 en 5.
* El código único de pagos debe iniciar en 1 e incrementar de 100 en 100.

Crear las siguientes funciones almacenadas para el campo xml de la tabla comprobantes de pago:

* Obtener info xml: Retornar la información del nombreUsuario, nombreProducto y montoPago del campo xml.
* Obtener info json: Retornar la información del emailUsuario, codigoProducto, montoPago.

QUINTA PREGUNTA.

* Realizar el llamado de los funciones y procedimientos almacenados de base de datos desde el lenguaje de programación java de la pregunta 1 y pregunta 2.