



**Tecnológico
de Monterrey**

PUEBLA

Bases de datos avanzadas

Dr. Daniel Pérez Rojas

ICA 1.1

3 Scenarios where you can implement:

Stored procedures

Triggers

Functions

Transactions

Scheduled events

Celeste Paiz Aguila A01327069

22 de febrero de 2018

Escenario 1: Sistema de préstamos de libros de una biblioteca

Las **tablas** necesarias para éste escenario son las siguientes:

Bibliotecario: id_bibliotecario, nombre, telefono, direccion, horario_entrada, horario_salida.

Cliente: id_cliente, nombre, telefono, direccion, correo.

Libro: id_libro, nombre, autor, editorial, edicion, costo_multa

Inventario: id_inventario, id_libro, cantidad, descripcion.

Prestamos: id_prestamo, id_cliente, id_libro, fecha_prestamo, id_bibliotecario, fecha_devolucion.

Devoluciones: id_devolucion, id_prestamo, fecha_prestamo, fecha_devolucion, id_multa.

Multas, id_multa, id_prestamo, id_libro, fecha_prestamo, fecha_devolucion, fecha_multa, costo_multa, multa_total.

Métodos de pago: id_metodo, nombre, descripcion.

Dentro de las **relaciones** de este escenario se encuentran las siguientes, al momento de crear un nuevo préstamo se debe tener información del cliente (id_cliente), libro(id_libro), el bibliotecario responsable del préstamo(id_bibliotecario); para las devoluciones, es necesario tener información del préstamo (id_prestamo), cliente(id_cliente), el

* **Stored Procedure**

Para consultar la disponibilidad de un libro dentro del inventario.

* **Trigger**

Sería necesario para registrar una multa en caso de que el cliente no devuelva el libro antes de la fecha límite indicada en el préstamo.

* **Function**

Función que reciba el costo_multa que tiene asociado un libro y los días para calcular la multa_total de la tabla multas.

Función que reciba el nombre de un cliente y devuelva la lista de libros que adquirió de la librería.

* **Transaction**

Para asegurar que los préstamos de las ventas y las multas hayan sido procesadas correctamente.

* **Scheduled event.**

Crear un evento para calcular cuantos libros fueron prestados por hora o por día.

Escenario 2: Cinépolis

En este escenario las tablas necesarias serían:

Usuario: id_usuario, nombre.

Cliente: id_cliente, nombre_cliente, correo, celular, password, id_tarjetaClubCinepolis.

Cajero: id_cajero, nombre_cajero, turno, celular, direccion.

Venta: id_venta, nombre_cliente, nombre_usuario, nombre_cajero, id_ticket, monto_total, id_cupon, id_promocion, monto_final.

Funcion: id_funcion, nombre_funcion, clasificacion, fecha_inicio, fecha_fin, hora_inicio, hora_fin.

Sucursal: id_sucursal, nombre_sucursal, direccion, telefono, correo.

TipoPago: id_tpago, nombre_tpago,

Cupon: id_cupon, codigo, cantidad_descuento, porcentaje_descuento, fecha_inicio_valida, fecha_vencimiento.

Promociones: id_promocion, cantidad_descuento, porcentaje_descuento, fecha_inicio, fecha_vencimiento.

TarjetaClubCinepolis: id_tarjetaClub, id_tipo_tarjeta, id_cliente.

Tipo_tarjeta: id_tipo_tarjeta, nombre, porcentaje_descuento, descripcion.

ticket: id_ticket, nombre_cliente, id_venta, nombre_cajero, nombre_sucursal, hora_inicio, hora_fin, fecha, nombre_tpago.

factura: id_factura, nombre_cliente_factura, id_ticket, fecha_factura.

Dentro de las relaciones se encuentran las siguientes: una venta necesita los datos del cliente(nombre_cliente) cuando ya está registrado y en caso de que no, se pondría el nombre del usuario(nombre_usuario), nombre del cajero(nombre_cajero), detalle del ticket (id_ticket) y monto total, también es importante considerar si se presentó cupón de descuento, pero éste es un campo opcional, como las promociones (id_promocion), que se reflejarían en el monto_final de la venta.

* Stored Procedure

Un proceso almacenado para mostrar la cartelera por día, que muestre las funciones por sucursal, clasificación y horarios disponibles.

* Trigger

Un trigger que valide y envíe notificaciones cuando ya no haya lugares disponibles para una función determinada.

* Function

Una función que reciba el id de la función y devuelva en cuales sucursales está disponibles.

* Transaction

Necesaria para asegurar que las ventas hayan sido actualizadas de manera correcta.

* Scheduled event.

Crear un evento para calcular cuantos tickets por película se vendieron al día para analizar cuál fue la película más taquillera.

Escenario 3: Aerolínea

cliente: id_cliente, nombre_cliente, num_pasaporte, visa, correo, password, celular, estado, direccion, codigo_postal.

Vuelo: id_vuelo, fecha_partida, fecha_regreso, destino, aeropuerto_origen, aeropuerto_destino,

Escala: id_escala, nombre_escala, lugar, duracion.

TipoDePago: id_tipoPago, nombre_tpago.

CuponDescuento: id_cupon, cantidad_descuento, fecha_inicio, fecha_fin.

Promocion: id_promocion, porcentaje_promocion, fecha_inicio, fecha_fin.

ticket: id_ticket, id_vuelo, id_cliente, nombre_cliente, id_cupon, id_promocion, id_tdvuelo, costo_total, costo_final, fecha_ticket.

factura: id_factura, id_ticket, nombre_cliente_factura, fecha_factura, datos

tipoDeVuelo: id_tdvuelo, nombre_tdvuelo. (nacional o internacional).

Dentro de las relaciones importantes se encuentra la tabla ticket, que debe contener la siguiente información, del vuelo (id_vuelo), del cliente(id_cliente y nombre_cliente), en caso de tener promoción(id_promocion) o descuento(id_descuento), el costo_total y el costo_aplicando descuentos o promociones.

*** Stored Procedure**

Un proceso almacenado para consultar los horarios y disponibilidad de vuelos dado un lugar de origen y otro de destino.

*** Trigger**

Un trigger que alerte con correo a los usuarios cuando hay descuentos en vuelos.

*** Function**

Una función que reciba el número de vuelo y devuelva cuantos lugares disponibles quedan para ese vuelo en específico.

*** Transaction**

Para asegurar que las ventas de boletos en línea fueron hechos de manera exitosa y la actualización de los vuelos disponibles es correcta.

*** Scheduled event.**

Un evento que muestre el vuelo más comprado por año, de ésta manera podemos saber cual fue el destino más popular.