

## **Consultas Select utilizadas**

### **SELECT \* FROM Historial\_Puntos;**

Obtiene todos los registros del historial de puntos acumulados y canjeados por los usuarios, permitiendo un seguimiento completo de los movimientos de puntos en el sistema de fidelización.

### **SELECT \* FROM Puntos;**

Recupera el estado actual de puntos acumulados por cada usuario, usado para mostrar su saldo disponible y gestionar recompensas.

### **SELECT \* FROM Detalles\_Facturas;**

Extrae todos los detalles asociados a las facturas, incluyendo precios unitarios, tasas y descuentos, para poder mostrar y procesar el desglose de cada factura.

### **SELECT \* FROM Pagos;**

Obtiene todos los registros de pagos realizados, incluyendo método, fecha y estado, para la gestión y verificación de transacciones.

### **SELECT \* FROM Facturas;**

Recupera todas las facturas generadas en el sistema, mostrando fecha e importe total, fundamental para el control financiero y facturación.

### **SELECT \* FROM Detalles\_Pedidos;**

Extrae los detalles específicos de cada pedido, como productos y cantidades, para poder mostrar y gestionar los pedidos realizados.

### **SELECT \* FROM Pedidos;**

Obtiene todos los pedidos registrados, relacionando factura, usuario y restaurante, usado para la gestión de pedidos y seguimiento.

### **SELECT \* FROM Resenas;**

Recupera todas las reseñas que los usuarios han dejado sobre restaurantes, incluyendo valoraciones y fechas, para mostrar opiniones y calificaciones.

### **SELECT \* FROM Empleados;**

Extrae información de todos los empleados, incluyendo datos personales, salario y fecha de contratación, para la administración del personal.

### **SELECT \* FROM Proveedores\_Ingredientes;**

Obtiene la relación entre proveedores y los ingredientes que suministran, junto con precios y tiempos de entrega, para la gestión de inventario y compras.

### **SELECT \* FROM Productos\_Ingredientes;**

Obtiene la relación entre proveedores y los ingredientes que suministran, junto con precios y tiempos de entrega, para la gestión de inventario y compras.

### **SELECT \* FROM Productos;**

Obtiene todos los productos disponibles, con sus categorías, nombres, precios y descripciones, para ser mostrados en la aplicación.

### **SELECT \* FROM Ofertas;**

Extrae todas las ofertas vigentes o pasadas, mostrando nombre, precio, descripción y fecha de expiración, para promociones en la tienda.

### **SELECT \* FROM Usuarios;**

Recupera la información completa de los usuarios registrados, incluyendo datos personales y credenciales, esencial para autenticación y gestión de perfiles.

### **SELECT \* FROM Ingredientes;**

Obtiene todos los ingredientes disponibles en el sistema, con detalles sobre stock, unidad de medida y estado, para gestión de inventario.

### **SELECT \* FROM Restaurante;**

Extrae la información de los restaurantes registrados, con sus datos de contacto y aforo, para mostrar en la aplicación y gestionar ubicaciones.

### **SELECT \* FROM Proveedores;**

Recupera datos de los proveedores, incluyendo empresa, contacto y email, para la gestión de relaciones comerciales.

### **SELECT \* FROM Almacen;**

Obtiene todos los almacenes, con nombre, capacidad y ubicación, para controlar la logística y almacenamiento de productos e ingredientes. Aunque solamente haya uno.

### **SELECT \* FROM Categorías;**

Extrae todas las categorías de productos, usadas para organizar y filtrar el catálogo en la aplicación.

### **SELECT \* FROM Alertas\_Stock;**

Recupera las alertas generadas cuando el stock de ingredientes está bajo, permitiendo acciones preventivas en la gestión de inventarios.

## Plan de contingencia

**Drop schema** if exists esquema\_reto → Elimina el esquema con ese nombre si existe, si no, se ejecuta la línea siguiente.

**Create schema** esquema\_reto → Crea un esquema con ese nombre, por el cual no estará llamado otro esquema gracias a la consulta anterior.

**Use esquema\_reto** → Comunica al gestor de la base de datos que las consultas posteriores las aplique al esquema especificado.

**Drop table** if exists en el siguiente orden de las tablas → Elimina las tablas en el orden en el que se encuentren con esos nombres si existen, en caso contrario, se ejecuta la siguiente consulta.

Historial\_Puntos,  
Puntos,  
Detalles\_Facturas,  
Pagos,  
Facturas,  
Detalles\_Pedidos,  
Pedidos,  
Resenas,  
Empleados,  
Proveedores\_Ingredientes,  
Productos\_Ingredientes,  
Productos,  
Ofertas,  
Usuarios,  
Ingredientes,  
Restaurante,  
Proveedores,  
Almacen,  
Categorias,  
Alertas\_Stock;

## Tablas:

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Categorías	ID_Categoria	INT	PK, Auto_increment	Identificador único de la categoría
	Nombre	Varchar (100)	NOT NULL	Nombre de la Categoría

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Ofertas	ID_Oferta	INT	PK, Auto_increment	Identificador de la oferta
	Nombre	Varchar (100)	NOT NULL	Nombre de la oferta
	Precio	Decimal (10,2)	NOT NULL	Precio de la oferta
	Descripción	TEXT	NULL permitido	Descripción de oferta
	columnaImagen	Varchar (255)	NULL permitido	Imagen de la oferta
	fechaExpiracion	DATE	NOT NULL	fecha de caducidad de la oferta

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Productos	ID_Producto	INT	PK, Auto_increment	Identificador del producto
	Nombre	Varchar (100)	NOT NULL	Nombre del producto
	Precio	Decimal (10,2)	NOT NULL	Precio del producto
	Descripción	TEXT	NULL permitido	Descripción del producto
	ImagenProducto	Varchar (250)	NOT NULL	Imagen del producto
	ID_Oferta	INT	FK → Ofertas	Oferta asociada
	ID_Categoria	INT	FK → Categorías	Categoría del producto

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Ingredientes	ID_Ingrediente	INT	PK, Auto_increment	Identificador del ingrediente
	Nombre	Varchar (100)	NOT NULL	Nombre del ingrediente
	UnidadMedida	Varchar(50)	NULL permitido	Unidad de Medida
	StockDisponible	Decimal (10,2)	NOT NULL	Cantidad disponible en stock
	TipoAlmacenamiento	Varchar(50)	NULL permitido	Condiciones almacenamiento
	EstaActivo	Boolean	NULL permitido	Si el ingrediente está disponible

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Productos_Ingr edientes	ID_ProductoIngr ediente	INT	PK, Auto_increment	Identificador del vínculo producto-ingredi ente
	ID_Ingrediente	INT	FK → Ingredientes	Ingrediente relacionado
	ID_Producto	INT	FK → Productos	Producto relacionado
	Cantidad	Decimal (10,2)	NOT NULL	Cantidad de ingrediente usado

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Proveedores	ID_Proveedor	INT	PK, Auto_increment	Identificador del proveedor
	NombreEmpres a	Varchar (100)	NOT NULL	Nombre de la empresa proveedora
	Telefono	Varchar (100)	NOT NULL	Telefono del proveedor
	Email	Varchar (100)	NULL permitido	Email del proveedor

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Almacen	ID_Almacen	INT	PK, Auto_increment	Identificador del almacén
	Nombre	Varchar (100)	NOT NULL	Nombre del almacén
	Capacidad	INT	NOT NULL	Capacidad total del almacén
	Ubicación	Varchar (100)	NULL permitido	Dirección o Ubicación



Tabla	Campo	Tipo de dato	Restricciones	Descripción
Proveedores_Ingredientes	ID_ProveedorIngrediente	INT	PK, Auto_increment	ID de relación proveedor-ingrediente
	ID_Proveedor	INT	FK → Proveedores	Proveedor del ingrediente
	ID_Ingrediente	DECIMAL(10,2)	FK → Ingredientes	Ingrediente suministrado
	precioUnitario	Varchar (100)	NOT NULL	Precio por unidad
	tiempoEntregaDias	INT	NULL permitido	Tiempo estimado de entrega

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Restaurante	ID_Restaurante	INT	PK, Auto_increment	Identificador del restaurante
	Nombre	Varchar (100)	NOT NULL	Nombre restaurante
	Direccion	Varchar (100)	NULL permitido	Dirección
	Telefono	Varchar (100)	NULL permitido	Teléfono
	Email	Varchar (100)	NULL permitido	Correo Electrónico
	Aforo	INT	NULL permitido	Capacidad máxima del restaurante
	imagenRestaurante	Varchar (250)	NULL permitido	Imagen del restaurante

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Empleados	ID_Empleado	INT	PK, Auto_increment	Identificador del empleado
	ID_Restaurante	INT	FK → Restaurante	Restaurante asignado
	Nombre	Varchar (100)	NOT NULL	Nombre empleados
	Apellidos	Varchar (100)	NOT NULL	Apellidos empleados
	DNI	Varchar (20)	NOT NULL	Documento de Identidad
	Telefono	Varchar (20)	NULL permitido	Teléfono
	Sueldo	DECIMAL(10,2)	NOT NULL	Salario
	FechaContratacion	DATE	NOT NULL	Fecha de contratación

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Usuarios	ID_Usuarios	INT	PK, Auto_increment	Identificador del usuario
	Nombre	Varchar (100)	NULL permitido	Nombre usuarios
	DNI	Varchar (20)	NULL permitido	Documento de Identidad
	Telefono	Varchar (20)	NULL permitido	Teléfono
	Email	Varchar (100)	NOT NULL UNIQUE	Correo Electrónico único
	Contrasena	Varchar (100)	NOT NULL	Contraseña
	DNI	Varchar (20)	NULL permitido	Documento de identidad
	Direccion	Varchar (200)	NULL permitido	Dirección

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Reseñas	ID_Reseñas	INT	PK, Auto_increment	ID de reseña
	ID_Usuario	INT	FK → Usuarios	Usuario que opina
	ID_Restaurante	INT	FK → Restaurante	Restaurante reseñado
	Valoracion	INT	NULL permitido	Puntuación
	Fecha	DATE	NULL permitido	Fecha de la reseña

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Pedidos	ID_Pedido	INT	PK, Auto_increment	ID del pedido
	ID_Factura	INT	FK → Facturas	Factura asociada
	ID_Restaurante	INT	FK → Restaurante	Restaurante donde se realiza
	ID_Usuario	INT	FK → usuarios	Usuario que realiza el pedido
	Numero	INT	NOT NULL	Número o folio del pedido

Tabla	Campo	Tipo de dato	Restricciones	Descripción
<b>Detalles_Pedidos</b>	ID_DetallePedido	INT	PK, Auto_increment	ID de detalle del pedido
	ID_Pedido	INT	FK → Pedidos	Pedido al que pertenece
	ID_Producto	INT	FK → Productos	Producto solicitado
	Cantidad	INT	NOT NULL	Cantidad solicitada
	Observaciones	TEXT	NULL permitido	Comentarios o notas

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Facturas	ID_Factura	INT	PK, Auto_increment	ID de la factura
	FechaFactura	DATE	NOT NULL	Fecha de emisión
	ImporteTotal	DECIMAL(10,2)	NOT NULL	Total de la factura

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Detalles_Facturas	ID_DetalleFactura	INT	PK, Auto_increment	ID de detalle de la factura
	ID_Factura	INT	FK → Facturas	Factura correspondiente
	ID_DetallePedidos	INT	FK → Detalles_Pedidos	Línea de pedido asociada
	PrecioUnitario	DECIMAL(10,2)	NOT NULL	Precio por unidad
	TasaLocal	DECIMAL(10,2)	NULL permitido	Tasa local
	Descuento	DECIMAL(10,2)	NULL permitido	Descuento aplicado

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Puntos	ID_puntos	INT	PK, Auto_increment	ID de puntos de usuario
	ID_Usuario	INT	FK → Usuarios	Usuario
	PuntosActuales	INT	NOT NULL	Puntos actuales acumulados

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Historial_Puntos	ID_HistorialPuntos	INT	PK, Auto_increment	ID el movimiento de puntos
	ID_Factura	INT	FK → Facturas	Factura relacionada
	Fecha	DATE	NOT NULL	Fecha del movimiento
	Puntos	INT	NOT NULL	Cantidad de puntos ganados/canjeados
	TipoMovimiento	ENUM('GANADO', 'CANJEADO')	NOT NULL	Tipo de movimiento
	Descripcion	TEXT	NULL permitido	Descripción de movimiento

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Pagos	ID_Pago	INT	PK, Auto_increment	ID del pago
	ID_Factura	INT	FK → Facturas	Factura pagada
	metodoPago	Varchar(50)	NOT NULL	Método de pago (efectivo, tarjeta...)
	fechaPago	DATE	NOT NULL	Fecha de pago
	estadoPago	Varchar(20)	NOT NULL	Estado del pago (pendiente, completado...)

Tabla	Campo	Tipo de dato	Restricciones	Descripción
Alertas_Stock	ID_Alerta	INT	PK, Auto_increment	ID del pago
	ID_Ingrediente	INT	FK → Facturas NOT NULL	ID del ingrediente
	fechaAlerta	DateTime	NOT NULL	fecha en la que se envía la alerta
	stockDisponible	decimal	NULL PERMITIDO	Cantidad de stock disponible
	mensaje	text	NULL PERMITIDO	Mensaje indicando el bajo stock de dicho producto

## Procedimiento necesario:

**DELIMITER \$\$** → Cambia el delimitador de SQL por defecto que es ; a \$\$.

**Drop Procedure** if exists RevisarStockBajo → Elimina un procedimiento con ese nombre en caso de que exista.

**Create Procedure** RevisarStockBajo() → Crea el procedimiento RevisarStockBajo.

**Begin y End\$\$** → Todo lo que ocurre dentro se ejecuta cuando se llama al procedimiento RevisarStockBajo().

```

INSERT INTO Alertas_Stock (ID_Ingrediente, StockDisponible, Mensaje)
SELECT
    ID_Ingrediente,
    StockDisponible,
    CONCAT('Stock bajo: ', StockDisponible, ' unidades disponibles para
    ingrediente ID ', Nombre, '')
FROM Ingredientes
WHERE StockDisponible < 10;

```

→ Consulta que comunica al procedimiento lo que debe hacer. En este caso se insertan los valores de ID\_Ingrediente, StockDisponible, Mensaje en la tabla de Alertas\_Stock cuando un ingrediente de la tabla Ingredientes tiene un StockDisponible menor a 10.

**Delimiter ;** → Final del procedimiento.

**CALL RevisarStockBajo();** → Llama a ese procedimiento para luego poder revisar en la tabla Alertas\_Stock que ingredientes hay que reponer.