

# Base de datos :::::

## Taller 4

**Docentes: Eric ross // Camilo veliz**

**Ayudante talleres: Manuel jerez**



### Objetivo

Normalizar, crear índices útiles, triggers y transacciones, y resolver consultas avanzadas.

### Dominio del problema

Un día cualquiera en la UCN, nuestra amiga **Emile**, una estudiante ingeniosa y emprendedora, conversaba con su compañero **Pauliño** sobre sus planes de lanzar una tienda online. Motivada por él, comenzó **TiendaTech** con gran entusiasmo...

Al principio, Emi lo apuntó todo en una **única hoja de cálculo**:

- Pedidos
- Clientes
- Productos
- Proveedores

Pronto Pauliño se dio cuenta de que aquel sistema manual ya no daba abasto:

- **Duplicación** de datos y riesgo de inconsistencia.
- **Errores al actualizar** dirección y email de clientes.
- **Falta de control de stock** al insertar o cancelar pedidos.
- **Reportes lentos** que tardan horas en procesar miles de registros.

Sabiendo que Emi necesitaba ayuda, Pauliño contactó a su amigo **Luis José**, experto en bases de datos. Él recomendó involucrar a los ¡¡grandiosos estudiantes de ICCI e ITI!! para diseñar juntos un modelo relacional sólido y automatizar todos esos procesos.

A continuación se muestra tabla de "ventasraw"

```
CREATE TABLE VentasRaw (  
  pedido_id          INT,  
  cliente_nombre     VARCHAR(100),  
  cliente_email      VARCHAR(100),  
  cliente_direccion  TEXT,  
  producto_id        INT,  
  producto_nombre     VARCHAR(100),  
  categoria_nombre   VARCHAR(50),  
  proveedor_nombre   VARCHAR(100),  
  proveedor_contacto VARCHAR(100),  
  fecha_pedido       TIMESTAMP,  
  ... )
```

```
cantidad          INT,  
precio_unitario    NUMERIC(10,2)  
);
```

Querys:

- Escribe las sentencias CREATE TABLE que representen tu modelo relacional en **3FN (Normalizado)**, incluyendo claves primarias, foráneas y tipos de datos. (*normalizar la tabla de arriba en pocas palabras*)
- Lista los **5 clientes** que más han gastado en total, mostrando cliente\_id, nombre, email y total\_gastado. Utiliza una función de ventana para ordenar y filtrar.
- La plataforma maneja categorías en múltiples niveles jerárquicos. Crea una consulta que, para cada categoría (hasta 3 niveles), muestre la ruta completa de la jerarquía y el número total de productos asignados a esa categoría. Incluye categorías sin productos.
- Muestra los **3 productos más vendidos** en términos de cantidad durante el último año, indicando producto\_id, nombre y total\_vendido. La consulta debe aprovechar un **índice parcial** que hayas definido previamente.
- Obtén los clientes cuyo email termina en @gmail.com **y** cuyo gasto total supera los 1.000 unidades monetarias. Debe emplearse al menos una subconsulta en la cláusula WHERE.
- Diseña y crea un **índice compuesto y parcial** que acelere la agregación de ventas del último año por producto (por ejemplo, sobre columnas producto\_id y cantidad), de tal forma que la consulta del ejercicio 4 se ejecute eficientemente.
- Implementa un trigger AFTER INSERT y AFTER UPDATE sobre la tabla de items de pedido que actualice el stock de cada producto. Si al actualizar el stock el valor resultante quedara negativo, tu trigger debe revertir (o evitar) la operación y registrar el incidente en una tabla auxiliar de errores.
- **Escribe un bloque transaccional** (BEGIN; ... COMMIT;) que:
  1. Inserte un nuevo pedido.
  2. Inserte varios items de pedido.
  3. Cree un SAVEPOINT antes de cada inserción de item y, si alguno falla, haga ROLLBACK TO SAVEPOINT correspondiente sin abortar toda la transacción.

## pistas

### 3FN:

- Cada tabla refleja una sola entidad.
- No hay dependencias transitivas: los atributos no clave dependen sólo de la clave primaria de su propia tabla.
- Se eliminan redundancias y anomalías de inserción, actualización y eliminación.

## Datos

El llenado de datos, lo harían a su modo, deben de proporcionar un ejemplo de entrada y salida de la query, algo simple con tablas en un Word.

## Consideraciones

- Para la entrega 1 y 2 se deberá de adjuntar en un formato PDF con el formato :  
[Nombre.ApellidoIntegrante1\_Nombre.ApellidoIntegrante2.pdf]
- El taller debe de ser realizado en **pareja** (2 personas)
- Entrega: 04/07/2025
- Se pueden agregar mas atributos si lo encuentra necesario.

- La entrega solo la hace 1 miembro.
- La retroalimentación se vera reflejada en la entrega de su nota vía campus virtual.
- Deben de trabajar para diagramar los modelos en una plataforma de diseño UML.
- Las consultas serán respondidas por correo ( [manuel.jerez@alumnos.ucn.cl](mailto:manuel.jerez@alumnos.ucn.cl) ) (los fin de semana si me hablan puedo no contestar).