

## Entrega 1: Grupo 33

### ESQUEMA

**Tiendas:** tid INT, tnombre VARCHAR(100), tdireccion VARCHAR(100), tcomuna VARCHAR(100), (jefe)nid INT

**Comunas\_despachos:** tid INT, dcomuna VARCHAR(100)

**Tienda\_vende:** tid INT, pid INT.

**Productos:** pid INT, pnombre VARCHAR(100), precio INT, pdescripcion VARCHAR(100), ptipo VARCHAR(100),

**Productos\_Comestibles:** pid INT, fechaexp DATE, categoria VARCHAR(100)

**Comestibles\_Conegelados:** pid INT, peso INT

**Comestibles\_Frescos:** pid INT, duracion INT

**Comestibles\_Conserva:** pid INT, método VARCHAR(100)

**Productos\_No\_Comestibles:** pid INT, ancho INT, largo INT, alto INT, peso FLOAT

**Personal:** nid INT, nnombre VARCHAR(100), rut INT, edad INT, sexo VARCHAR(100), tid INT.

**Usuarios:** uid INT, unombre VARCHAR(100), rut INT, edad INT

**Direcciones\_usuario:** uid INT, udireccion VARCHAR(100), comuna VARCHAR(100),

**Compras:** cid INT, uid\_INT, tid\_INT, udireccion VARCHAR(100)

**Productos\_Compra:** cid INT, pid INT, cantidad INT

### Argumentación Esquema

- Realizamos Comunas\_despachos para poder relacionar las tiendas con las comunas a las que despachan. Así todo queda bien dividido y ordenado. Esto se da para abordar el “cada tienda realiza el despacho de sus productos a varias comunas de Chile”.
- La tabla Tienda\_vende relaciona los productos que tiene cada Tienda con los pid que tiene disponibles para su venta, los cuales serían los id de la lista de Productos.
- Dividimos Productos en Productos\_Comestibles y Productos\_No\_Comestibles, y luego Productos\_Comestibles en Congelados, Frescos y Conserva, en tablas distintas, de esta forma existe una correcta separación por categoría de los productos y es fácil hacer una consulta de forma ordenada.

- Personal y Usuarios son personas con sus atributos. Cada tienda tiene un atributo para definir quién es el jefe (que debe coincidir con el id de algún Personal)
- Relacionamos las direcciones con comunas con columnas extra en la respectiva entidad. Así podemos conseguir Tienda.tdireccion y Tienda.tcomuna igualmente que al momento de conseguir las direcciones de los usuarios. Este supuesto se da luego de consultar en la plataforma Discord el día 14-04-2021.
- Dado que un usuario puede tener varias direcciones, hicimos una tabla que relacione direcciones con id de los usuarios.
- Hicimos una tabla Compras, para así registrar las compras que realizan los usuarios, en que tienda y asignar un id a la compra para luego buscar la lista de los productos comprados en la tabla Productos\_Compra. Además, cada tupla registra la dirección a la que el usuario quiere el despacho, dado que puede tener muchas direcciones. Todo con el fin de mantener el orden y no perder información.

### **Supuestos:**

- Los precios son int al igual que el peso, alto, largo y ancho, es un dato que se podría cambiar a float si así se requiere en la próxima entrega.
- Cada usuario asociado a una o más direcciones: Se asume que un usuario puede poseer distintas direcciones a la cual la tienda puede despachar los productos comprados.
- En la tabla de Tiendas se va a considerar el jefe de tienda para tener un mayor orden de los datos.
- De forma de no tener *nulls* en las tablas hemos decidido dividir la información de los productos comestibles y no comestibles.
- Vamos a omitir el dígito verificador del rut, por simplicidad y porque no aporta información.
- El jefe de Tiendas tiene que ser consecuente con el id del Personal asociado, de modo que un trabajador no sea jefe de una tienda en la que no trabaja.

### **CONSULTAS**

**i) Muestre nombre de todas las tiendas, junto con los nombres de las comunas a cuales realizan despachos.**

#### **Algebra:**

$\pi$  Tiendas.tnombre, Comunas\_despachos.dcomuna (Tiendas  $\bowtie$  Tiendas.tid = Comunas\_despachos.tid Comunas\_despachos)

#### **SQL:**

**SELECT** Tiendas.tnombre, Comunas\_despachos.dcomuna

**FROM** Tiendas, Comunas\_despachos  
**WHERE** Tiendas.tid = Comunas\_despachos.tid

ii) Muestre a todos los jefes de tiendas ubicadas en la comuna de 'San Joaquín'.

**Algebra Relacional:**

```
π Personal.nnombre, Personal.rut, Personal.edad, Personal.sexo, Personal.tid (σ  
Tiendas.tcomuna = 'San Joaquín' (Tiendas ⋈ Tiendas.tid = Personal.tid ∧ Tiendas.tjefe =  
Personal.nid Personal))
```

**SQL:**

```
SELECT Personal.nnombre, Personal.rut, Personal.edad, Personal.sexo, Personal.tid  
FROM Tiendas, Personal  
WHERE Tiendas.tid = Personal.tid AND Personal.nid = Tiendas.nid AND Tiendas.tcomuna =  
'San Joaquín'
```

iii) Muestra todas las tiendas que venden al menos un producto no comestible

**Algebra Relacional:**

```
π Tiendas.tid, Tiendas.tnombre, Tiendas.tdireccion, Tiendas.tcomuna, Tiendas.nid σ  
Tiendas.tid = Tienda_Vende.tid and Tienda_Vende.pid = Productos.pid and Productos.ptipo =  
'No Comestible' ( ( Tiendas ⋈ Productos ) ⋈ Tienda_Vende )
```

**SQL:**

```
SELECT DISTINCT Tiendas.tid, Tiendas.tnombre, Tiendas.tdireccion, Tiendas.tcomuna,  
Tiendas.nid  
FROM Tiendas, Productos, Tienda_Vende  
WHERE Tiendas.tid = Tienda_Vende.tid AND Tienda_Vende.pid = Productos.pid AND  
Productos.ptipo = 'No Comestible'
```

iv) Muestre a todos los usuarios que compraron el producto con la descripción 'pantalla de luz'.

**SQL:**

```
SELECT Usuarios.uid, Usuarios.unombre, Usuarios.rut, Usuarios.edad  
FROM Usuarios, Productos, Compras, Productos_Compra  
WHERE ( Usuarios.uid = Compras.uid AND Compras.cid = Productos_Compra.cid AND  
Productos.pid = Productos_Compra.pid AND Productos.pdescripcion = 'Pantalla de Luz')
```

v) Muestre la edad promedio de los trabajadores de tiendas en la comuna 'Coquimbo'.

**SQL:**

```
SELECT AVG(Personal.edad) AS Prom_edad  
FROM Tiendas, Personal  
WHERE Tiendas.tid = Personal.tid AND Tiendas.tcomuna = 'Coquimbo'
```

vi) Muestre la tienda que ha registrado la mayor cantidad de productos comestibles.

**SQL:**

```
SELECT FOO.tid, FOO.tnombre, FOO.tdireccion, FOO.tcomuna, FOO.nid FROM (SELECT  
Tiendas.tid, Tiendas.tnombre, Tiendas.tdireccion, Tiendas.tcomuna, Tiendas.nid, agg.cuenta  
FROM Tiendas, (  
    SELECT Tienda_Vende.tid as tid, COUNT(Productos.ptipo) as cuenta  
    FROM Tienda_Vende, Productos  
    WHERE Tienda_Vende.pid = Productos.pid AND Productos.ptipo = 'Comestible'  
    GROUP BY tid) as agg  
WHERE Tiendas.tid = agg.tid  
ORDER BY agg.cuenta DESC) as FOO  
FETCH FIRST 1 ROW ONLY
```

## DATA:

**Agregamos todos los datos utilizados en las tablas por cualquier cosa.  
(Aunque ya están incluidas en el servidor)**

group: Proyecto

```
Tiendas = {
tid:number, tnombre:string, tdireccion:string, tcomuna:string, nid:number
1, 'Easy', 'Fray Camilo Henriquez 69', 'Puente Alto', 3
2, 'Jumbo', 'Fernando Hurtado 15', 'San Joaquín', 2
3, 'Sodimac', 'Los pozos 19', 'San Joaquín', 1
4, 'Lider', 'Irrarazabal 23', 'Las Condes', 6
5, 'Casa Ideas', 'Coquim 92', 'Coquimbo', 4
6, 'Donde el Papa', 'Carmen 77', 'Coquimbo', 5
}
Comunas_despachos = {
tid:number, dcomuna:string
1, 'Puente Alto'
2, 'Las Condes'
3, 'Huechuraba'
4, 'Providencia'
1, 'Las Condes'
3, 'Ñuñoa'
5, 'Huechuraba'
5, 'Santiago Centro'
6, 'Lo barnechea'
}
```

```
Personal = {
nid:number, nnombre:string, rut:number, edad:number, sexo:string, tid:number
1, 'Claudio Maniega', 202109187, 21, 'Masculino', 3
2, 'Carolina Astorquiza', 202109183, 89, 'Femenino', 2
3, 'Dominic Toretto', 202109189, 43, 'Masculino', 1
4, 'Mac Miller', 202109181, 10, 'Masculino', 3
5, 'Javier de la Maza', 730274839, 99, 'Masculino', 5
6, 'Jaime del Río', 293658372, 9, 'Masculino', 6
}
```

```
Tienda_Vende = {
tid:number, pid:number
1, 1
1, 3
1, 4
1, 5
}
```

```
2, 2
2, 3
3, 1
3, 6
4, 4
2, 1
2, 7
5, 7
6, 1
}
```

```
Productos = {
pid:number, pnombre:string, precio:number, pdescripcion:string, ptipo:string
1, 'Jamón', 2990, 'Ahumado', 'Comestible'
2, 'Queso', 1990, 'Leche de Vaca', 'Comestible'
3, 'Lapiz', 990, 'Azul tinta', 'No Comestible'
4, 'Computador', 199990, 'Negro', 'No Comestible'
5, 'Telefono', 49990, 'Pantalla de Luz', 'No Comestible'
6, 'Reloj', 20000, 'Pantalla de Luz', 'No Comestible'
7, 'Pizza', 34000, 'de queso', 'Comestible'
}
```

```
Compras = {
cid:number, uid:number, tid:number, udireccion:string
1, 3, 2, 'Los pozos 898'
2, 4, 1, 'Vespucio 22'
3, 1, 1, 'Colon 7200'
4, 2, 3, 'Irarrazaval 69'
}
```

```
Productos_Compra = {
cid:number, pid:number, cantidad:number
1, 2, 1
2, 5, 2
3, 1, 3
4, 6, 1
}
```

```
Usuarios = {
uid:number, unombre:string, rut:number, edad:number
1, 'Luis Jara', 12933468, 999
2, 'Negro Piñera', 69034851, 66
3, 'Luli Love', 9234572, 44
4, 'Karol Dance', 20436856, 21
}
```