



**UNIVERSIDAD**

Nicolás Felipe Bernal Gallo

Juan Daniel Bogotá Fuetes

Modelos y bases de datos

MBDA LAB

Laboratorio #2 Diseño conceptual general SQL-  
DQL Básico

21/02/2025

PROFESOR: María Irma Díaz Roza

## MODELOS Y BASES DE DATOS

### Diseño Conceptual Extendido. SQL - DQL

2025-1

#### Laboratorio 2/6

#### OBJETIVOS

Evaluar el logro de las competencias adquiridas para:

- ✓ Identificar los grandes conceptos presentes en un modelo conceptual
- ✓ Diseñar e implementar considerando ciclos de desarrollo
- ✓ Extender un modelo conceptual considerando la información de tablas. En reversa.
- ✓ Proponer un modelo de casos de uso de funciones, dado un diagrama de conceptos.
- ✓ Proponer un modelo de casos de uso de consultas operativas.
- ✓ Escribir consultas de más de una tabla y usando valor desconocido en SQL.

#### ENTREGA

- Incluyan **lab02.doc** y **adventure.asta** en un archivo **.zip**. El nombre de este archivo debe ser la concatenación en orden alfabético de los primeros apellidos de cada uno de los miembros del equipo.
- Deben publicar el avance al final de la sesión y la versión definitiva en la fecha indicada. Publiquen en Moodle en los espacios correspondientes.

El modelo de datos que vamos a trabajar es **Adventure Works** una de las evaluaciones propuestas en el tutorial **SQLZoo.net** en **MySQL**

**No olviden estructurar las sentencias para que sean claras.**

## PARTE UNO. Refactorización

[En [lab02.doc](#) y [adventure.asta](#)]

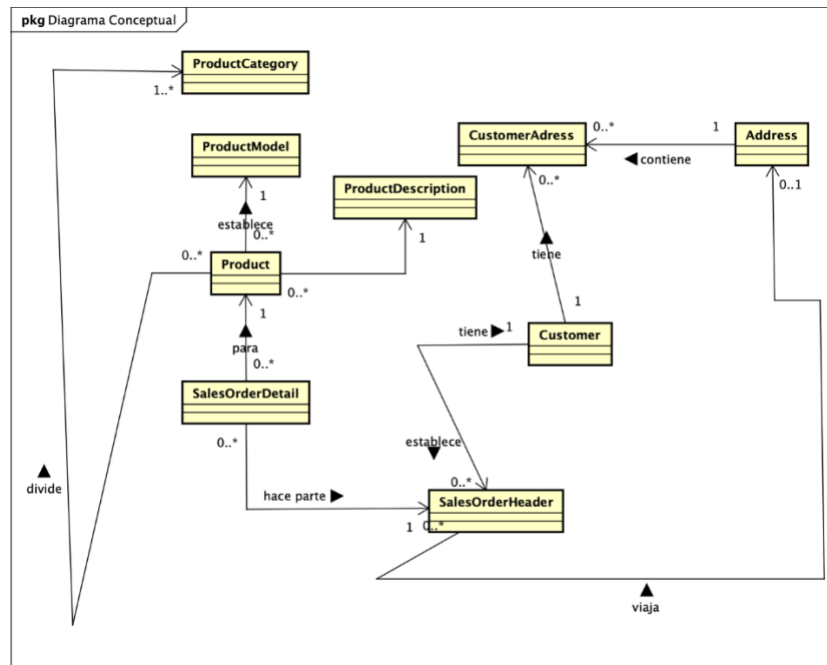
(ContenidoBaseDatos → Modelo lógico → Modelo conceptual)

En este punto vamos a perfeccionar el trabajo realizado en el laboratorio anterior. No olviden considerar los comentarios de la entrega.

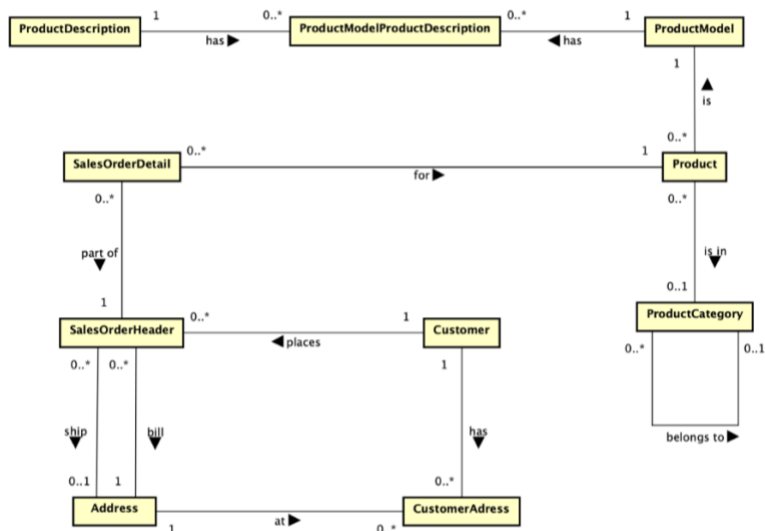
### A. Modelo conceptual

- 1) Revisen su modelo conceptual y perfecciónenlo. ¿Cuáles fueron los cambios realizados?

**Antes:**



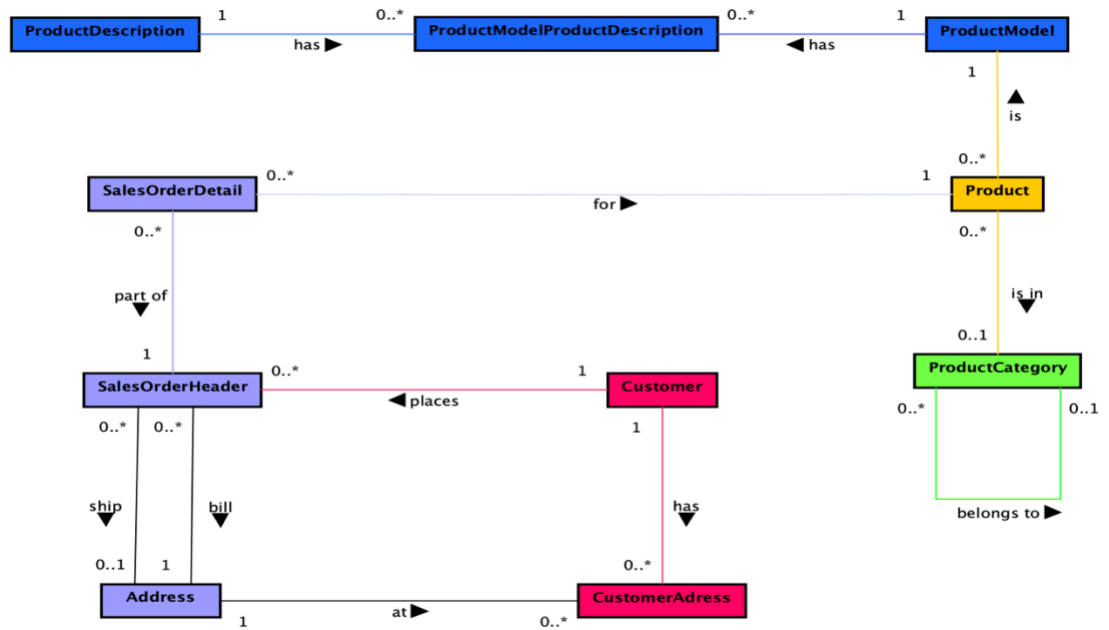
**Después:**



Los cambios realizados son:

Las flechas de las relaciones, ocultamos el compartimiento de los atributos y de las operaciones, corregimos la multiplicidad y también se corrigió las relaciones.

- 2) Señalen los grandes conceptos (GC) con colores diferentes (GC: Conceptos + Relaciones)

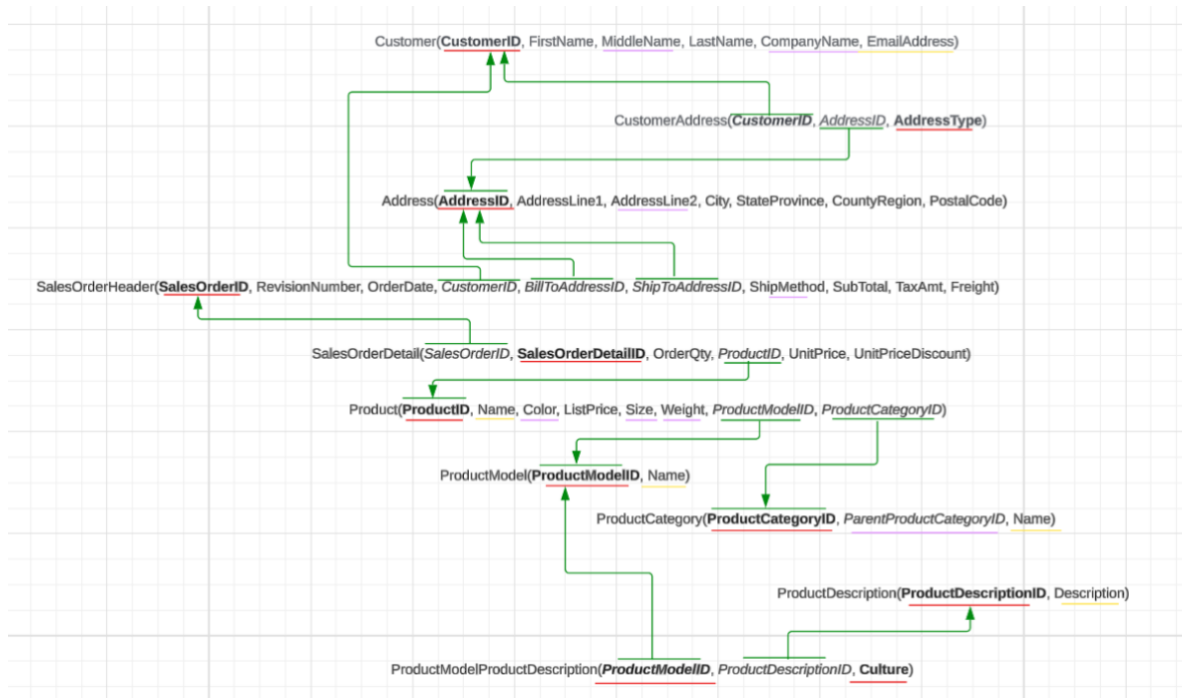
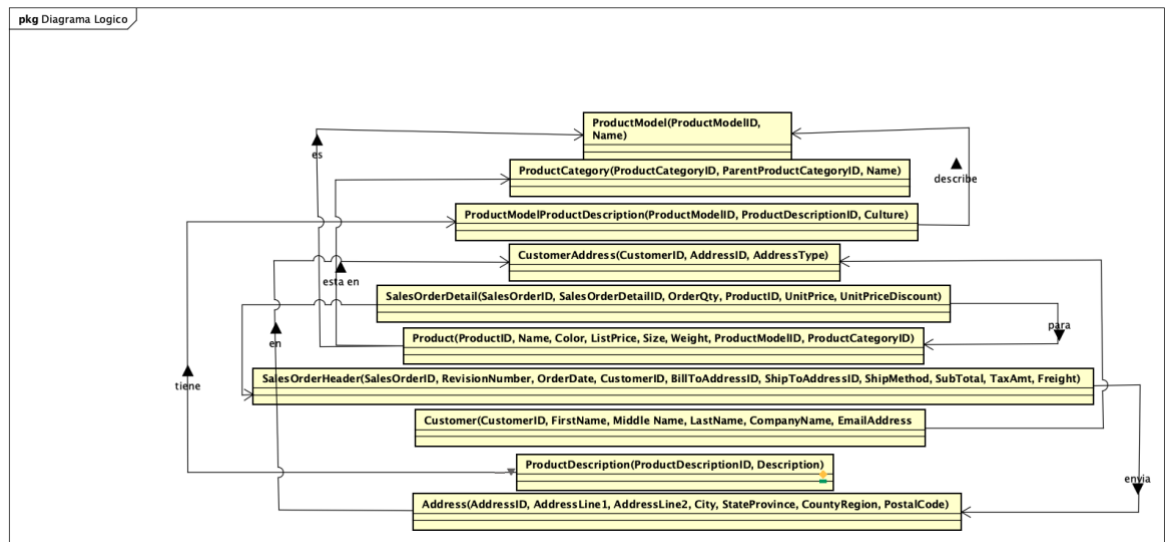


## B. Modelo lógico

### 1) Revisen su modelo lógico y perfecciónenlo. ¿Cuáles fueron los cambios realizados?

(Consulten la definición de claves y nulidades de cada tabla con el comando correspondiente al motor<sup>1</sup>)

Decidan cuáles atributos podrían quedar como desconocidos (que pueden ser nulos). Justifiquen su selección.)

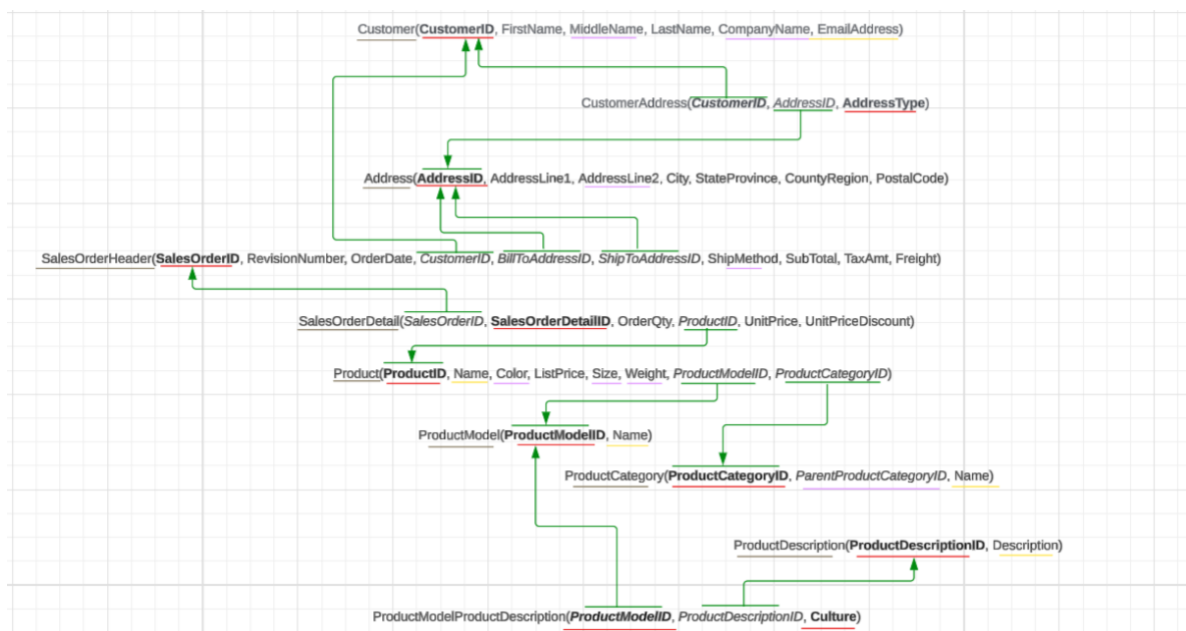


= UK  
 = PK  
 = FK

■ = NULL

- MiddleName: No todos tienen segundo nombre.
- CompanyName: Clientes individuales no tienen empresa.
- AddressLine2: No se necesita segunda línea en todas las direcciones.
- ShipMethod: Si la compra es digital, no hay envío.
- Color: No es importante el color ya que no todos los productos tienen color definido.
- Size: No todos los productos tienen tamaño relevante.
- Weight: No todos los productos requieren un peso registrado.
- ParentProductCategoryID: Algunas categorías no tienen subcategoría.

## 2) Señalen los grandes conceptos con colores diferentes (CRUD: Tablas)



Create: Insertar en customer y product

Read: Consultar en SalesOrderDetail, ProductModel,

ProductDescription y ProductCategory

Update: Actualizar Address y SalesOrderDetail.

Delete: Eliminar registros ProductCategory

## PARTE DOS. División por ciclos

[En [adventure.asta](#)]

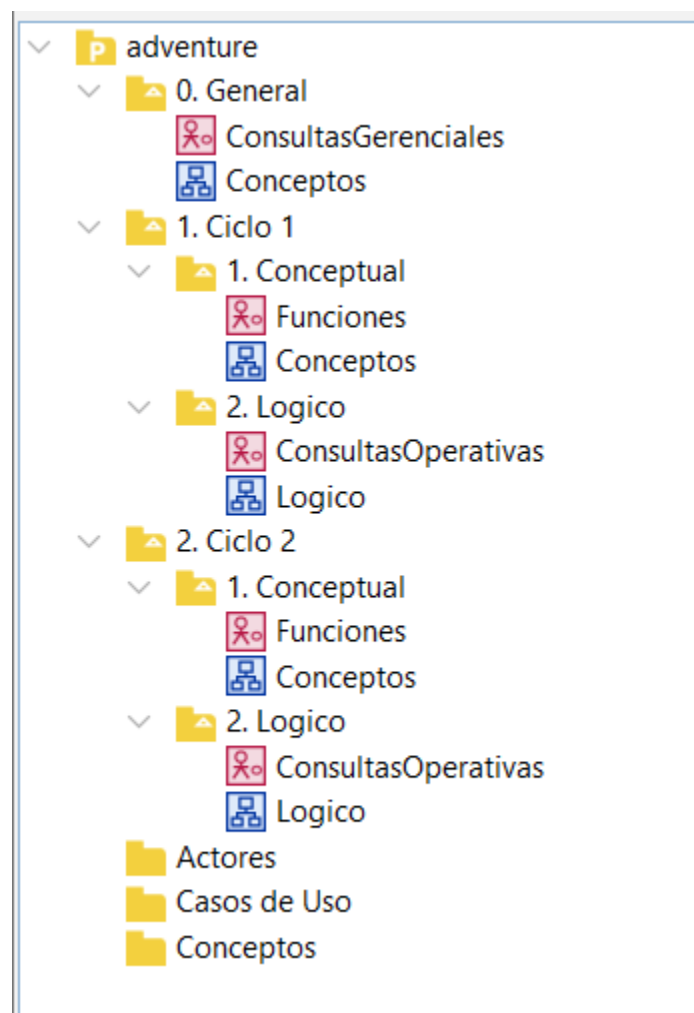
### A. Definición de ciclos

Para continuar, vamos a dividir el trabajo en ciclos de desarrollo e implementar un ciclo:

**Ciclo 1:** Información del área de mercadeo (productos, categorías y modelos)

**Ciclo 2:** Información del área de ventas (clientes, ventas)

- 1) Organicen el contenido en las carpetas de diseño considerando las especificaciones de entrega.<sup>2</sup>
- 2) Preparen las carpetas correspondientes al ciclo dos de desarrollo.<sup>3</sup>



## **PARTE TRES. Ciclo uno.**

[En [adventure.asta lab02.doc](#)]

### **A. Modelo conceptual. Conceptos. (¿qué conoce?)**

- 1) Realicen el diagrama de conceptos<sup>4</sup> extendido.

Únicamente extiendan los conceptos del ciclo.

(Consulten la especificación de los tipos de las columnas con el comando correspondiente al motor<sup>5</sup>)

No olviden indicar para cada uno de los atributos tipo y modificador,

cardinalidad y requisito de unicidad, cuando sea necesario.

Usen Integer, Real, Boolean, String y DateTime como tipos del modelo conceptual. Recuerden los atributos a los que decidieron permitir que fueran desconocidos.

El modelo extendido se encuentra en el archivo adventure.astah siguiendo la ruta:

adventure/ 1.Ciclo 1 / 1. Conceptual/ Funciones

### **B. Modelo conceptual. Funciones. (¿qué hace?)**

- 1) Realicen el diagrama de funciones.<sup>6</sup>

¿Cuáles casos de uso son necesarios para almacenar la información del ciclo?

Usen el estándar Mantener <Objeto>, Mantener <Rol> y Registrar <Evento>

Usen para los casos de uso los colores de los CRUD.



### C. Modelo conceptual. Consultas Operativas. (¿qué ofrece?)

1) Implemente las siguientes nuevas consultas-<sup>7</sup>

- Los productos que más clientes han comprado

[categoria, modelo, producto, número de veces]

(De mayor a menor. Incluir las que se han vendido más veces que el promedio)

WITH Ventas AS (

```
SELECT pc.Name AS Categoria, pm.Name AS Modelo, p.Name AS Producto,
COUNT(DISTINCT soh.CustomerID) AS NumeroDeVeces
FROM SalesOrderDetail sod
JOIN SalesOrderHeader soh ON sod.SalesOrderID = soh.SalesOrderID
JOIN Product p ON sod.ProductID = p.ProductID
LEFT JOIN ProductModel pm ON p.ProductModelID = pm.ProductModelID
LEFT JOIN ProductCategory pc ON p.ProductCategoryID =
pc.ProductCategoryID
GROUP BY pc.Name, pm.Name, p.Name
```

)

```
SELECT Categoria, Modelo, Producto, NumeroDeVeces
FROM Ventas
WHERE NumeroDeVeces > (SELECT AVG(NumeroDeVeces) FROM Ventas)
ORDER BY NumeroDeVeces DESC;
```

Result:

Categoria	Modelo	Producto	NumeroDeVeces
Jerseys	Long-Sleeve Logo Jersey	Long-Sleeve Logo Jersey, L	9
Vests	Classic Vest	Classic Vest, S	9
Jerseys	Short-Sleeve Classic Jersey	Short-Sleeve Classic Jersey, XL	8
Caps	Cycling Cap	AWC Logo Cap	8
Jerseys	Short-Sleeve Classic Jersey	Short-Sleeve Classic Jersey, L	7
Brakes	Front Brakes	Front Brakes	7
Bike Racks	Hitch Rack - 4-Bike	Hitch Rack - 4-Bike	7

- Duós de productos que se venden juntos.  
[Los dos nombres y las veces que se venden juntos]  
(Ordenamos de mayor a menor. Los primeros cinco dúos.)

```

WITH Pares AS (
    SELECT
        p1.Name AS Producto1, p2.Name AS Producto2, COUNT(*) AS
        VecesJuntos
    FROM SalesOrderDetail sod1
    JOIN SalesOrderDetail sod2
        ON sod1.SalesOrderID = sod2.SalesOrderID
        AND sod1.ProductID < sod2.ProductID
    JOIN Product p1 ON sod1.ProductID = p1.ProductID
    JOIN Product p2 ON sod2.ProductID = p2.ProductID
    GROUP BY p1.Name, p2.Name
)

SELECT Producto1, Producto2, VecesJuntos
FROM Pares
ORDER BY VecesJuntos DESC
LIMIT 5;

```

## Result:

Producto1	Producto2	VecesJuntos
Classic Vest, S	Short-Sleeve Classic Jersey, XL	8
Long-Sleeve Logo Jersey, L	Short-Sleeve Classic Jersey, L	7
Hitch Rack - 4-Bike	Short-Sleeve Classic Jersey, XL	7
AWC Logo Cap	Short-Sleeve Classic Jersey, L	7
Classic Vest, S	Hitch Rack - 4-Bike	7

- **Propongan una nueva consulta para esta área**

2) De las consultas **easy o medium** propuestas en SQLZOO, seleccionen y generalicen la que consideren más relevante para este ciclo de desarrollo. Implémentenla, si no lo han hecho. <sup>8</sup>

- Clientes con mayor variedad de productos comprados:

```
SELECT c.CustomerID, c.FirstName + ' ' + c.LastName AS Cliente,  
  
COUNT(DISTINCT sod.ProductID) AS ProductosDistintos  
  
FROM SalesOrderHeader soh  
  
JOIN SalesOrderDetail sod ON soh.SalesOrderID = sod.SalesOrderID  
  
JOIN Customer c ON soh.CustomerID = c.CustomerID  
  
GROUP BY c.CustomerID, c.FirstName, c.LastName  
  
ORDER BY ProductosDistintos DESC;
```

## Result:

CustomerID	Cliente	ProductosDist..
61	0	50
502	0	46
142	0	46
448	0	43
24	0	43
582	0	43
187	0	31

3) De las consultas anteriores, diseñen las dos que consideren son las mejores consultas. Preséntenas en el diagrama de casos de uso<sup>9</sup> .

```
SELECT soh.SalesOrderID, c.CompanyName, soh.SubTotal, SUM(p.Weight *  
sod.OrderQty) AS TotalWeight
```

```
FROM SalesOrderHeader soh
```

```
JOIN Customer c ON soh.CustomerID = c.CustomerID
```

```
JOIN SalesOrderDetail sod ON soh.SalesOrderID = sod.SalesOrderID
```

```
JOIN Product p ON sod.ProductID = p.ProductID
```

```
GROUP BY soh.SalesOrderID, c.CompanyName, soh.SubTotal
```

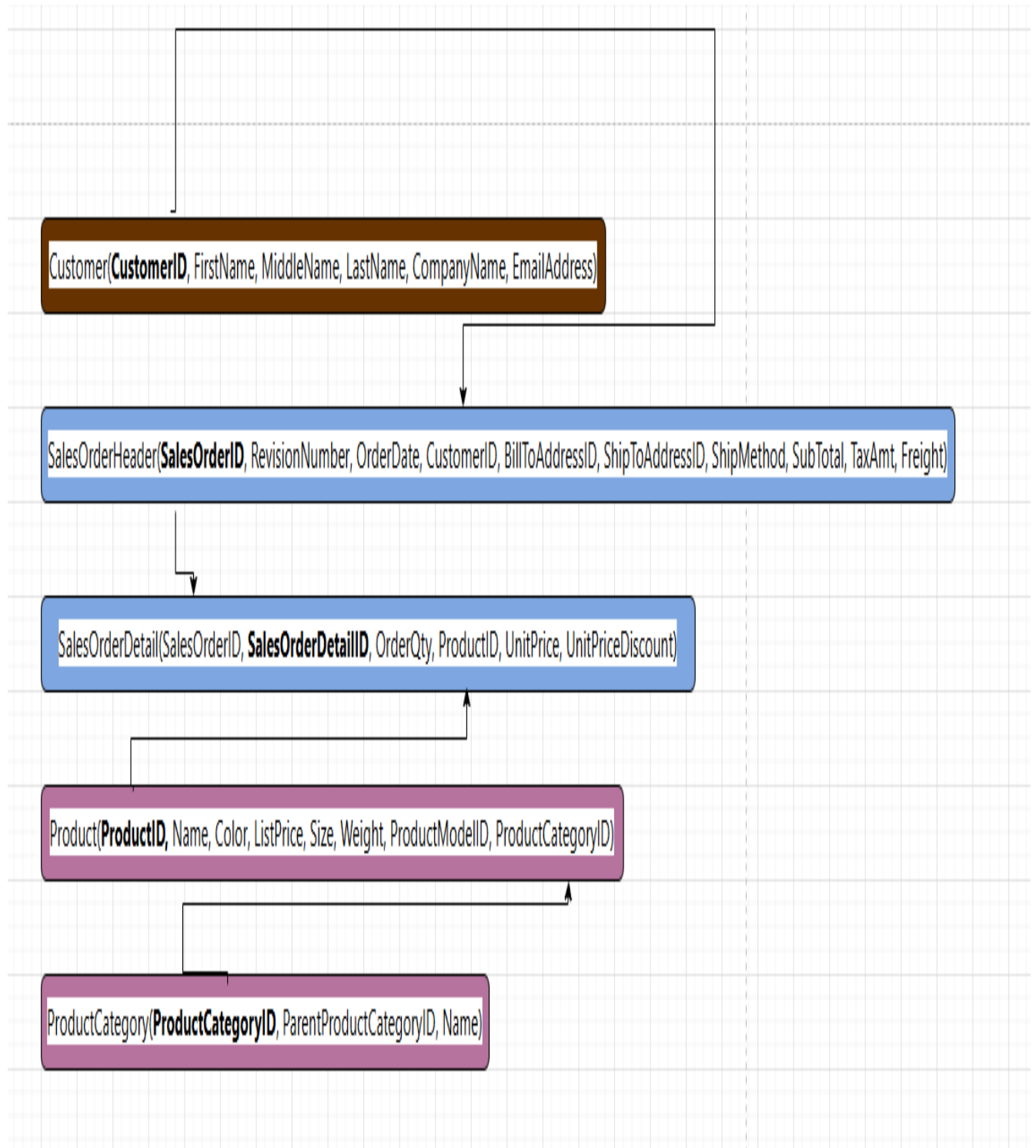
```
ORDER BY soh.SubTotal DESC;
```

## Result:

SalesOrderID	CompanyName	SubTotal	TotalWeight
71784	Action Bicycle Specialists	108561.83	1133911.56
71936	Metropolitan Bicycle Supply	98278.69	679588.02
71938	Bulk Discount Store	88812.86	34813.05
71783	Eastside Department Store	83858.43	565638.72
71797	Riding Cycles	78029.69	504095.33
71902	Many Bikes Store	74058.81	744328.60
71898	Instruments and Parts Company	63980.99	731576.77

#### D. Modelo lógico. (¿cómo se almacena?)

1) Editen el modelo lógico general para que en este sólo queden las tablas necesarias para el ciclo: las propias y las de referencia.



#### **PARTE CUATRO. Extendiendo ciclo uno.**

- Implementen las consultas **Hard questions** en SQL

[Escriban las consultas en los lenguajes pedidos y prueben la consulta SQL en sqlzoo. Si no lograron escribir alguna, Indiquen el punto de problema]

- De las consultas **hard** seleccione la que considere más importante para adicionarla al ciclo uno. Diseñenla (no olvide generalizarlas). Preséntenla en un diagrama de casos de uso<sup>10</sup>.

4 En astah cree un diagrama de clases (adventure/1. Ciclo 1/ 1. Conceptual / Conceptos) Configure la vista del modelo para que únicamente sean visibles la zona de atributos sin visibilidad, pero con tipos.

5 En moodle hay una referencia a los comandos

6 En (adventure/1. Ciclo 1/1. Conceptual/ Funciones). No incluya historias de uso.

7 Incluya la propuesta de la nueva consulta, las sentencias SQL y las respuestas en el documento del laboratorio.

8 Incluya la sentencia SQL y la respuesta de la consulta en el documento del laboratorio.

9 En astah cree un diagrama de casos de uso (adventure/1. Ciclo

1/ConsultasOperativas). Incluya la historia y el detalle (columnas, filtro, orden).

10 En astah cree un diagrama de casos de uso (adventure/1. Ciclo uno/ConsultasOperativas). Incluya la historia y el detalle.

#### **PUNTO 11:**

SELECT

```
    C.CompanyName,
    MAX(CASE WHEN CA.AddressType = 'Main Office' then A.AddressLine1
ELSE " END) as MainOfficeAddress,
    MAX(CASE WHEN CA.AddressType = 'Shipping' then A.AddressLine1
ELSE " END) AS ShippingAddress
FROM CustomerAddress AS CA
JOIN Address AS A
  ON CA.AddressID = A.AddressID
JOIN Customer AS C
  ON CA.CustomerID = C.CustomerID
WHERE A.City ='Dallas'
GROUP BY C.CompanyName
```

# 11.

**For every customer with a 'Main Office' in Dallas show AddressLine1 of the 'Main Office' and AddressLine1 of the 'Shipping' address - if there is no shipping address leave it blank. Use one row per customer.**

```
JOIN Customer AS C  
  ON CA.CustomerID = C.CustomerID  
WHERE A.City = 'Dallas'  
GROUP BY C.CompanyName
```

Submit SQL

restore default

## Result:

CompanyName	MainOfficeAddress	ShippingAddress
Elite Bikes	Po Box 8259024	9178 Jumping St.
Rental Bikes	99828 Routh Street, Suite 825	
Third Bike Store	2500 North Stemmons Freeway	
Town Industries	P.O. Box 6256916	
Unsurpassed Bikes	Po Box 8035996	

## PUNTO 12:

```
SELECT
    SOH.SalesOrderID,
    SOH.SubTotal AS 'Method A',
    SUM(SOD.OrderQty * SOD.UnitPrice) AS 'Method B',
    SUM(SOD.OrderQty * P.ListPrice) AS 'Method C'
FROM SalesOrderHeader AS SOH
JOIN SalesOrderDetail AS SOD
    ON SOH.SalesOrderID = SOD.SalesOrderID
JOIN Product AS P
    ON SOD.ProductID = P.ProductID
GROUP BY SOH.SalesOrderID, SOH.SubTotal
```



# 12.

For each order show the SalesOrderID and SubTotal calculated three ways:

A) From the SalesOrderHeader

B) Sum of OrderQty\*UnitPrice

C) Sum of OrderQty\*ListPrice

```
FROM SalesOrderHeader AS SOH
JOIN SalesOrderDetail AS SOD
  ON SOH.SalesOrderID = SOD.SalesOrderID
JOIN Product AS P
  ON SOD.ProductID = P.ProductID
GROUP BY SOH.SalesOrderID, SOH.SubTotal
```

Submit SQL

restore default

## Result:

SalesOrderID	Method A	Method B	Method C
71774	880.35	713.80	1189.66
71776	78.81	63.90	106.50
71780	38418.69	30600.81	56651.56
71782	39785.33	33319.68	55533.31
71783	83858.43	68141.99	121625.43
71784	108561.83	90341.14	151932.58
71796	57634.63	47848.02	79746.71
71797	78029.69	65218.20	108986.40

### PUNTO 13:

```
SELECT
    SOD.ProductID,
    P.Name,
    P.ListPrice,
    SUM(SOD.OrderQty) AS 'Total Sold',
    SUM(SOD.OrderQty * P.ListPrice) AS 'Order Value'
FROM SalesOrderDetail AS SOD
JOIN Product AS P
    ON SOD.ProductID = P.ProductID
GROUP BY SOD.ProductID, P.Name, P.listPrice
ORDER BY SUM(SOD.OrderQty * P.ListPrice) DESC
LIMIT 1
```

# 13.

**Show the best selling item by value.**

```
JOIN Product AS P
    ON SOD.ProductID = P.ProductID
GROUP BY SOD.ProductID, P.Name, P.listPrice
ORDER BY SUM(SOD.OrderQty * P.ListPrice) DESC
LIMIT 1
```

**Submit SQL**

restore default

**Result:**

ProductID	Name	ListPrice	Total Sold	Order Value
976	Road-350-W Yellow, 48	1700.99	37	62936.63

#### **PUNTO 14:**

```
SELECT
    `Range`,
    COUNT(SalesOrderID) AS 'Num Orders',
    SUM(TotalValue) AS 'Total Value'
FROM (
    SELECT
        SOH.SalesOrderID,
        CASE
            WHEN SUM(SOH.SubTotal + SOH.TaxAmt) BETWEEN 0 AND 99 THEN
'0-99'
            WHEN SUM(SOH.SubTotal + SOH.TaxAmt) BETWEEN 100 AND 999
THEN '100-999'
            WHEN SUM(SOH.SubTotal + SOH.TaxAmt) BETWEEN 1000 AND 9999
THEN '1000-9999'
            ELSE '10000+'
        END AS `Range`,
        SUM(SOH.SubTotal + SOH.TaxAmt) AS TotalValue
    FROM SalesOrderHeader AS SOH
    JOIN SalesOrderDetail AS SOD
        ON SOH.SalesOrderID = SOD.SalesOrderID
    GROUP BY SOH.SalesOrderID
) AS DerivedTable
GROUP BY `Range`;
```

# 14.

Show how many orders are in the following ranges (in \$):

RANGE	Num Orders	Total Value
0- 99		
100- 999		
1000-9999		
10000-		

```
SELECT  
  `Range`,  
  COUNT(SalesOrderID) AS 'Num Orders',  
  SUM(TotalValue) AS 'Total Value'  
FROM (  
  SELECT
```

Submit SQL

restore default

## Result:

Range	Num Orders	Total Value
0-99	2	129.28
100-999	2	1144.62
1000-9999	6	16361.29
10000+	21	31092861.07

### **PUNTO 15:**

```
WITH t1 AS (  
    SELECT city, SUM(unitprice * orderqty) total_sales  
    FROM SalesOrderDetail sod JOIN SalesOrderHeader soh ON sod.salesorderid =  
soh.salesorderid  
    JOIN Address a ON soh.shiptoaddressid = a.addressid  
    GROUP BY city  
    ORDER BY total_sales DESC  
    LIMIT 3  
)  
SELECT city, pc.name, SUM(unitprice * orderqty) total_sales  
FROM SalesOrderDetail sod JOIN SalesOrderHeader soh ON sod.salesorderid =  
soh.salesorderid  
JOIN Address a ON soh.shiptoaddressid = a.addressid  
JOIN Product p ON sod.productid = p.productid  
JOIN ProductCategory pc ON p.productcategoryid = pc.productcategoryid  
WHERE city IN (SELECT city FROM t1)  
GROUP BY city, pc.name  
ORDER BY city, total_sales DESC;
```

# 15.

**Identify the three most important cities. Show the break down of top level product category against city.**

```
WITH t1 AS (  
  SELECT city, SUM(unitprice * orderqty) total_sales  
  FROM SalesOrderDetail sod JOIN SalesOrderHeader soh ON  
sod.salesorderid = soh.salesorderid  
  JOIN Address a ON soh.shiptoaddressid = a.addressid  
  GROUP BY city
```

Submit SQL

restore default

## RETROSPECTIVA

¿Cuál fue el tiempo total invertido en el laboratorio por cada uno de ustedes?  
(Horas/Hombre)

Juan Daniel Bogotá Fuentes 20 horas, más o menos 2 horas y media por día  
(viernes, lunes, martes, miércoles, jueves, viernes).

Nicolas Felipe Bernal Gallo 20 horas (viernes[3 horas], lunes[2 horas], martes[3 horas], miércoles[1 hora], jueves[6 horas], viernes[6 horas])

¿Cuál es el estado actual del laboratorio? ¿Por qué?

El estado es incompleto, falta profundizar mayormente en algunas preguntas.

¿Cuál consideran fue el mayor logro? ¿Por qué?

El mayor logro fue aprender a usar astah y entender cada punto del lab.

¿Cuál consideran que fue el mayor problema técnico? ¿Qué hicieron para resolverlo?

Aprender a usar astah sin ningún tipo de aprendizaje previo, ver videos tutoriales en YouTube.

¿Qué hicieron bien como equipo? ¿Qué se comprometen a hacer para mejorar los resultados?

Trabajar y repartirnos bien las cargas de cada punto.

¿Qué referencias usaron? ¿Cuál fue la más útil? Incluyan citas con estándares adecuados.

Astah. (2024, 30 octubre). Class diagrams - Astah. <https://astah.net/support/astah-pro/user-guide/class-diagrams/>

Yuliza Angela Huanca M. (2020, 7 diciembre). Video tutorial Herramienta de modelado

ASTAH UML [Video]. YouTube. <https://www.youtube.com/watch?v=-jZ8JkC-Pwc>

Siriwardhana, S. (2023, 26 junio). Tutorial de diagramas de casos de uso ( Guía con ejemplos ).