

Entregable 3 – Validaciones

Validaciones para la tabla de hechos

- Validación de cantidad de registros

```
SELECT COUNT(*) AS [Cant registros DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas;
```

```
SELECT COUNT(*) AS [Cant registros DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS;
```

Results		Messages	
		Cant registros DB_COMERCIAL	
1		11992	
		Cant registros DW_COMERCIAL	
1		11992	

Conclusión: Los valores coinciden, validación correcta.

- Validación de Montos Totales

```
SELECT SUM(Monto_Vendido) AS [Monto total ventas DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas;
```

```
SELECT SUM(Monto_Vendido) AS [Monto total ventas DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS;
```

Results		Messages	
		Monto total ventas DB_COMERCIAL	
1		5122990000.00	
		Monto total ventas DW_COMERCIAL	
1		5122990000.00	

Conclusión: Los valores coinciden, validación correcta.

- Validación de suma de cantidades, aperturadas por distintas dimensiones
 - Suma de cantidades vendidas por producto

```
SELECT P.Desc_Producto , SUM(V.Cantidad_Vendida) AS [Sum Cant vendida DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas AS V
      JOIN DB_COMERCIAL.dbo.Producto AS P
            ON P.COD_Producto = V.COD_Producto
GROUP BY P.Desc_Producto
ORDER BY 2 DESC;
```

```
SELECT P.DESCR_PRODUCTO, SUM(V.CANTIDAD_VENDIDA) AS [Sum Cant vendida DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS AS V
      JOIN DW_COMERCIAL.dbo.DIM_PRODUCTO AS P
            ON P.PRODUCTO_KEY = V.PRODUCTO_KEY
GROUP BY P.DESCR_PRODUCTO
ORDER BY 2 DESC;
```

	Desc_Producto	Sum Cant vendida DB_COMERCIAL		DESC_PRODUCTO	Sum Cant vendida DW_COMERCIAL
1	Papel	44559.00	1	Papel	44559.00
2	Accesorios	33280.00	2	Accesorios	33280.00
3	Teléfonos	33191.00	3	Teléfonos	33191.00
4	Muebles	30208.00	4	Muebles	30208.00
5	Impresoras	27984.00	5	Impresoras	27984.00
6	Lápices	24501.00	6	Lápices	24501.00
7	Organizadores	18679.00	7	Organizadores	18679.00
8	Cafetera	15240.00	8	Cafetera	15240.00
9	Sillas	12757.00	9	Sillas	12757.00
10	Mesas	12629.00	10	Mesas	12629.00
11	PC	11833.00	11	PC	11833.00
12	Etiquetas	10304.00	12	Etiquetas	10304.00
13	Sobres	9222.00	13	Sobres	9222.00
14	Elásticos	6785.00	14	Elásticos	6785.00
15	Archivo	6769.00	15	Archivo	6769.00
16	Tijeras	5331.00	16	Tijeras	5331.00
17	Copiadoras	2936.00	17	Copiadoras	2936.00

Conclusión: Los valores coinciden, validación correcta.

b) Suma de montos vendidos por categoría

```

SELECT C.Desc_Categoria, SUM(V.Monto_Vendido) AS [Sum Monto vendido DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas AS V
      JOIN DB_COMERCIAL.dbo.Categoria AS C
      ON C.COD_Categoria = V.COD_CATEGORIA
GROUP BY C.Desc_Categoria
ORDER BY 2 DESC;

SELECT C.DESCR_CATEGORIA, SUM(V.MONTO_VENDIDO) AS [Sum Monto vendido DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS AS V
      JOIN DW_COMERCIAL.dbo.DIM_CATEGORIA AS C
      ON C.CATEGORIA_KEY = V.CATEGORIA_KEY
GROUP BY C.DESCR_CATEGORIA
ORDER BY 2 DESC;

```

	Desc_Categoria	Sum Monto vendido DB_COMERCIAL		DESC_CATEGORIA	Sum Monto vendido DW_COMERCIAL
1	Oficina	2183840000.00	1	Oficina	2183840000.00
2	Tecnologia	1610600000.00	2	Tecnologia	1610600000.00
3	Mobiliario	1328550000.00	3	Mobiliario	1328550000.00

Conclusión: Los valores coinciden, validación correcta.

c) Top 3 vendedores por total vendido (\$)

```
SELECT TOP (3)
  DV.COD_Vendedor, DV.Desc_Vendedor,
  SUM(FV.Monto_Vendido) AS [Total vendido DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas AS FV
      JOIN DB_COMERCIAL.dbo.Vendedor AS DV
            ON DV.COD_Vendedor =
FV.COD_VENDEDOR
GROUP BY DV.COD_Vendedor, DV.Desc_Vendedor
ORDER BY 3 DESC;
```

```
SELECT TOP (3)
  DV.COD_VENDEDOR, DV.NOMBRE, DV.APELLIDO,
  SUM(FV.MONTO_VENDIDO) AS [Total vendido DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS AS FV
      JOIN DW_COMERCIAL.dbo.DIM_VENDEDOR AS DV
            ON DV.VENDEDOR_KEY =
FV.VENDEDOR_KEY
GROUP BY DV.COD_VENDEDOR, DV.NOMBRE, DV.APELLIDO
ORDER BY 3 DESC;
```

	COD_Vendedor	Desc_Vendedor	Total vendido DB_COMERCIAL
1	V00038	MARIA LLANOS CANO	106320000.00
2	V00006	PILAR SERRANO	101470000.00
3	V00014	ISABEL SANCHEZ	98350000.00

	COD_VENDEDOR	NOMBRE	APELLIDO	Total vendido DW_COMERCIAL
1	V00015	EMILIO	VALERO	75610000.00
2	V00019	PAULA	VALERO	67500000.00
3	V00046	CONCEPCION	TORRES	63030000.00

Conclusión: Los valores coinciden, validación correcta.

d) Top 3 clientes por total comprado (\$)

```
SELECT TOP (3)
  C.COD_Cliente, C.Desc_Cliente,
  SUM(V.Monto_Vendido) AS [Total comprado DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas AS V
      JOIN DB_COMERCIAL.dbo.Cliente AS C
            ON C.COD_Cliente = V.COD_CLIENTE
GROUP BY C.COD_Cliente, C.Desc_Cliente
ORDER BY 3 DESC;
```

```
SELECT TOP (3)
  C.COD_CLIENTE, C.NOMBRE, C.APELLIDO,
  SUM(V.MONTO_VENDIDO) AS [Total comprado DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS AS V
      JOIN DW_COMERCIAL.dbo.DIM_CLIENTE AS C
            ON C.CLIENTE_KEY = V.CLIENTE_KEY
GROUP BY C.COD_CLIENTE, C.NOMBRE, C.APELLIDO
ORDER BY 4 DESC;
```

	COD_Cliente	Desc_Cliente	Total comprado DB_COMERCIAL
1	CLI00786	Emily Phan	38200000.00
2	CLI00044	Darren Budd	24740000.00
3	CLI00624	Grant Carroll	23520000.00

	COD_CLIENTE	NOMBRE	APELLIDO	Total comprado DW_COMERCIAL
1	CLI00786	Emily	Phan	38200000.00
2	CLI00044	Darren	Budd	24740000.00
3	CLI00624	Grant	Carroll	23520000.00

Conclusión: Los valores coinciden, validación correcta.

e) Top 3 productos más vendidos (\$)

```

SELECT TOP (3)
    P.COD_Producto, P.Desc_Producto,
    SUM(V.Monto_Vendido) AS [Ingresos totales DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Ventas AS V
    JOIN DB_COMERCIAL.dbo.Producto AS P
        ON P.COD_Producto = V.COD_PRODUCTO
GROUP BY P.COD_Producto, P.Desc_Producto
ORDER BY 3 DESC;

SELECT TOP (3)
    P.COD_PRODUCTO, P.DESC_PRODUCTO,
    SUM(V.MONTO_VENDIDO) AS [Ingresos totales DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.FACT_VENTAS AS V
    JOIN DW_COMERCIAL.dbo.DIM_PRODUCTO AS P
        ON P.PRODUCTO_KEY = V.PRODUCTO_KEY
GROUP BY P.COD_PRODUCTO, P.DESC_PRODUCTO
ORDER BY 3 DESC;

```

	COD_Producto	Desc_Producto	Ingresos totales DB_COMERCIAL
1	P00015	Teléfonos	597820000.00
2	P00007	Papel	503230000.00
3	P00017	Accesorios	471120000.00

	COD_PRODUCTO	DESC_PRODUCTO	Ingresos totales DW_COMERCIAL
1	P00015	Teléfonos	597820000.00
2	P00007	Papel	503230000.00
3	P00017	Accesorios	471120000.00

Conclusión: Los valores coinciden, validación correcta.

Validaciones para las tablas de dimensiones

- Cantidad de Clientes total (#)

```

SELECT COUNT(*) AS [Cant de clientes DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Cliente;

```

```

SELECT COUNT(*) AS [Cant de clientes DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.DIM_CLIENTE;

```

Results		Messages
		Cant de clientes DB_COMERCIAL
1	795	
		Cant de clientes DW_COMERCIAL
1	797	

Conclusión: La cantidad de clientes es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

- Cantidad de Vendedores total (#)

```
SELECT COUNT(*) AS [Cant de vendedores DB_COMERCIAL]  
FROM DB_COMERCIAL.dbo.Vendedor;
```

```
SELECT COUNT(*) AS [Cant de vendedores DW_COMERCIAL]  
FROM DW_COMERCIAL.dbo.DIM_VENDEDOR;
```

Results	
Messages	
Cant de vendedores DB_COMERCIAL	
1	65
Cant de vendedores DW_COMERCIAL	
1	67

Conclusión: La cantidad de vendedores es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

- Cantidad de Paises total (#)

```
SELECT COUNT(*) AS [Cant de paises DB_COMERCIAL]  
FROM DB_COMERCIAL.dbo.Pais;
```

```
SELECT COUNT(*) AS [Cant de paises DW_COMERCIAL]  
FROM DW_COMERCIAL.dbo.DIM_PAIS;
```

Results	
Messages	
Cant de paises DB_COMERCIAL	
1	13
Cant de paises DW_COMERCIAL	
1	15

Conclusión: La cantidad de paises es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

- Cantidad de Productos Total (#)

```
SELECT COUNT(*) AS [Cant de productos DB_COMERCIAL]  
FROM DB_COMERCIAL.dbo.Producto;
```

```
SELECT COUNT(*) AS [Cant de productos DW_COMERCIAL]  
FROM DW_COMERCIAL.dbo.DIM_PRODUCTO;
```

Results	
Messages	
Cant de productos DB_COMERCIAL	
1	17
Cant de productos DW_COMERCIAL	
1	19

Conclusión: La cantidad de productos es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

- Cantidad de Categorías Total (#)

```
SELECT COUNT(*) AS [Cant de categorías DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Categoria;
```

```
SELECT COUNT(*) AS [Cant de categorías DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.DIM_CATEGORIA;
```

Results		Messages	
Cant de categorias DB_COMERCIAL			
1	3		
Cant de categorias DW_COMERCIAL			
1	5		

Conclusión: La cantidad de categorías es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

- Cantidad total de Sucursales Total (#)

```
SELECT COUNT(*) AS [Cant de sucursales DB_COMERCIAL]
FROM DB_COMERCIAL.dbo.Sucursal;
```

```
SELECT COUNT(*) AS [Cant de sucursales DW_COMERCIAL]
FROM DW_COMERCIAL.dbo.DIM_SUCURSAL;
```

Results		Messages	
	Cant de sucursales DB_COMERCIAL		
1	65		
	Cant de sucursales DW_COMERCIAL		
1	67		

Conclusión: La cantidad de sucursales es la esperada. Difiere en 2 unidades únicamente por los valores de los inserts iniciales (-1 para valor 'Sin Especificar' y 0 para valor 'No Aplica'). Validación correcta.

Visualización de los datos:

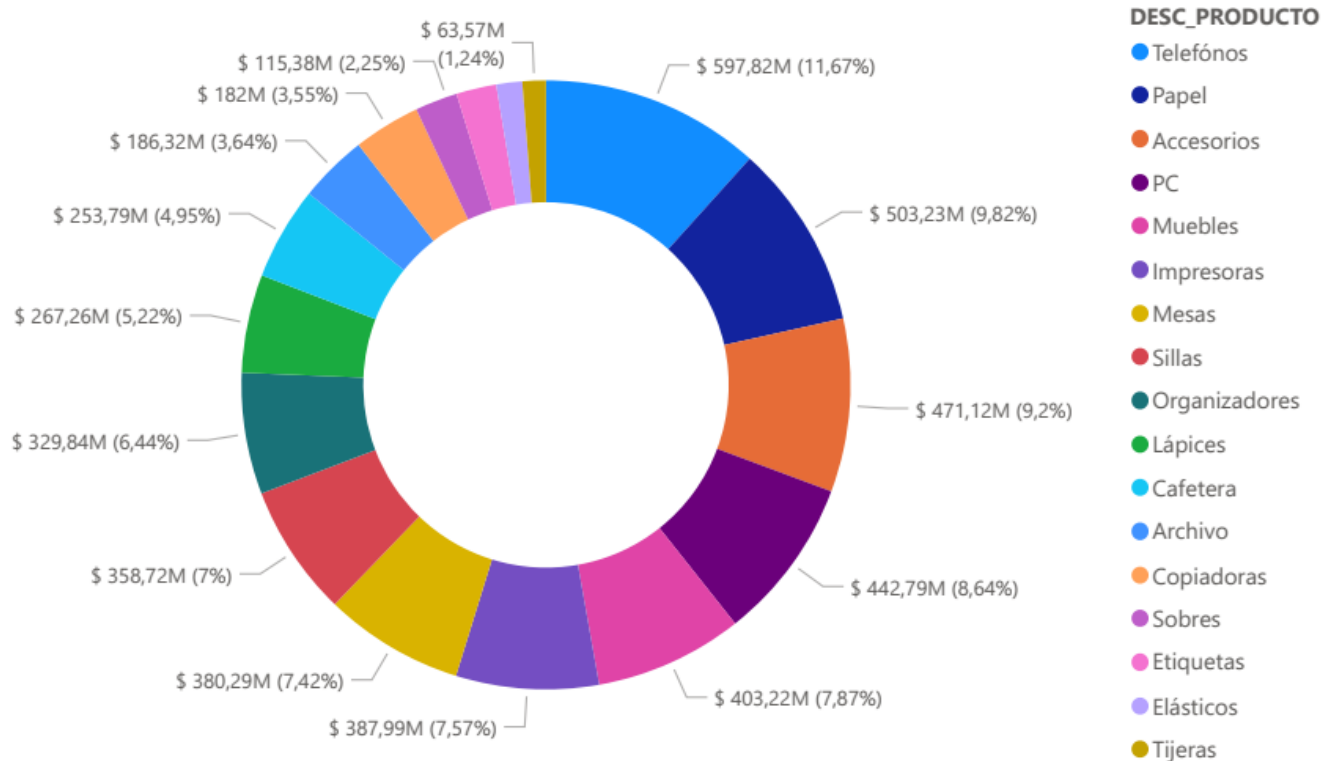
- Porcentaje de Total vendido que representa cada producto

```
DECLARE @Suma_total AS DECIMAL(18,2);
SELECT
@Suma_total = SUM(MONTO_VENDIDO) FROM FACT_VENTAS;

WITH Aux1 AS
(
    SELECT DESC_PRODUCTO, SUM(MONTO_VENDIDO) AS Suma_prod
    FROM FACT_VENTAS AS FV
    JOIN DIM_PRODUCTO AS DP
    ON DP.PRODUCTO_KEY = FV.PRODUCTO_KEY
    GROUP BY DP.DESC_PRODUCTO
)
SELECT Aux1.DESC_PRODUCTO,
(100.0 * Aux1.Suma_prod/@Suma_total) AS [Porcentaje de ventas total]
FROM Aux1
ORDER BY [Porcentaje de ventas total] DESC;
```

	DESC_PRODUCTO	Porcentaje de ventas total
1	Teléfonos	11.669357
2	Papel	9.822974
3	Accesorios	9.196192
4	PC	8.643194
5	Muebles	7.870794
6	Impresoras	7.573506
7	Mesas	7.423204
8	Sillas	7.002160
9	Organizadores	6.438427
10	Lápices	5.216875
11	Cafetera	4.953942
12	Archivo	3.636938
13	Copiadoras	3.552612
14	Sobres	2.252200
15	Etiquetas	2.129615
16	Elásticos	1.377125
17	Tijeras	1.240876

Porcentaje de total vendido que representa cada producto



- Porcentaje de total vendido que representa cada categoría

```

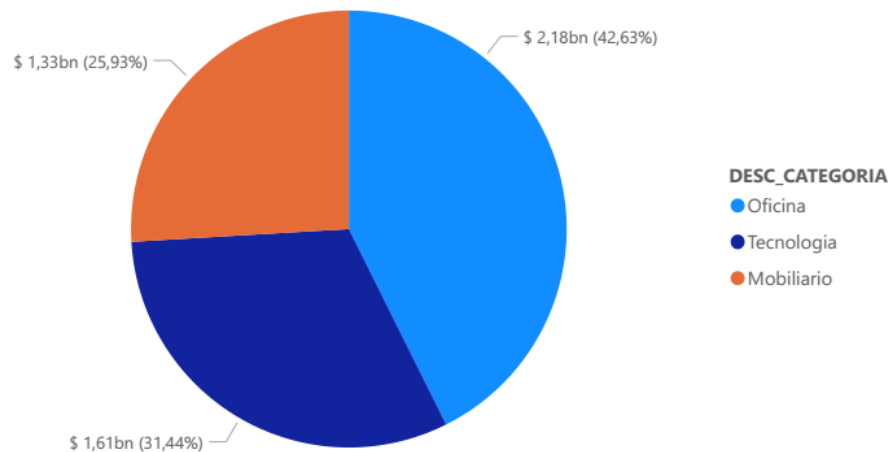
DECLARE @Suma_total AS DECIMAL(18,2);
SELECT
@Suma_total = SUM(MONTO_VENDIDO) FROM FACT_VENTAS;

WITH Aux2 AS
(
    SELECT DC.DISC_CATEGORIA, SUM(MONTO_VENDIDO) AS Suma_cat
    FROM FACT_VENTAS AS FV
    JOIN DIM_CATEGORIA AS DC
    ON DC.CATEGORIA_KEY = FV.CATEGORIA_KEY
    GROUP BY DC.DISC_CATEGORIA
)
SELECT DISC_CATEGORIA,
(100.0 * Suma_cat/@Suma_total) AS [Porcentaje de ventas total]
FROM Aux2
ORDER BY [Porcentaje de ventas total] DESC;

```

	DESC_CATEGORIA	Porcentaje de ventas total
1	Oficina	42.628230
2	Tecnologia	31.438671
3	Mobiliario	25.933097

Porcentaje de total vendido que representa cada categoria

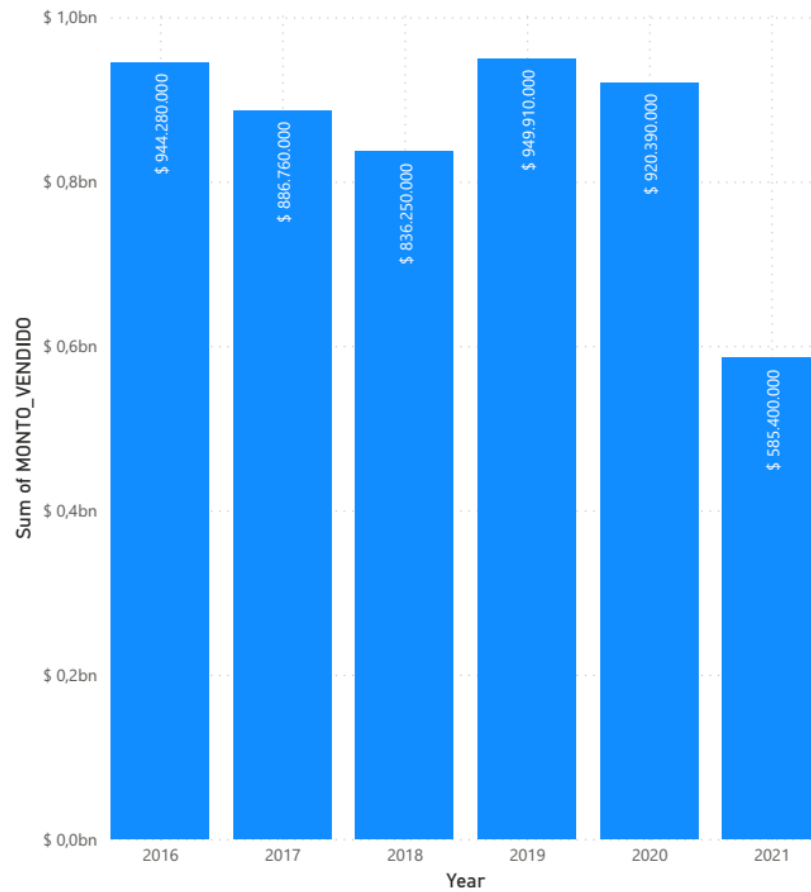


- Suma de monto vendido por año

```
SELECT DATEPART(YEAR, TIEMPO_KEY) AS ANIO,
       SUM(MONTO_VENDIDO) AS [Monto total vendido]
FROM FACT_VENTAS
GROUP BY DATEPART(YEAR, TIEMPO_KEY)
ORDER BY ANIO ASC;
```

	ANIO	Monto total vendido
1	2016	944280000.00
2	2017	886760000.00
3	2018	836250000.00
4	2019	949910000.00
5	2020	920390000.00
6	2021	585400000.00

Suma de monto vendido por año



- Suma de monto vendido por año y cuatrimestre

```
SELECT DATEPART(YEAR, TIEMPO_KEY) AS ANIO,
       DATEPART(QUARTER, TIEMPO_KEY) AS CUATRIMESTRE,
       SUM(MONTO_VENDIDO) AS [Monto total vendido]
FROM FACT_VENTAS
GROUP BY DATEPART(YEAR, TIEMPO_KEY), DATEPART(QUARTER, TIEMPO_KEY)
ORDER BY ANIO ASC, CUATRIMESTRE ASC;
```

	ANIO	CUATRIMESTRE	Monto total vendido
1	2016	1	261040000.00
2	2016	2	216290000.00
3	2016	3	242760000.00
4	2016	4	224190000.00
5	2017	1	206800000.00
6	2017	2	217500000.00
7	2017	3	221970000.00
8	2017	4	240490000.00
9	2018	1	196310000.00
10	2018	2	209270000.00
11	2018	3	200140000.00
12	2018	4	230530000.00

	ANIO	CUATRIMESTRE	Monto total vendido
13	2019	1	227200000.00
14	2019	2	230200000.00
15	2019	3	227390000.00
16	2019	4	265120000.00
17	2020	1	236850000.00
18	2020	2	222070000.00
19	2020	3	223030000.00
20	2020	4	238440000.00
21	2021	1	196310000.00
22	2021	2	188950000.00
23	2021	3	200140000.00

