

EVIDENCIA DE APRENDIZAJE 1.

MODELO ESTRELLA DE UN DATA MART

CARLOS ALBERTO VELEZ MUÑOZ GRUPO B010076

GERMAN DARIO ARBELAEZ RIOS GRUPO B010093

JUAN CARLOS PRESIGA MONTOYA GRUPO B010094

BASES DE DATOS II

VICTOR HUGO MERCADO



MEDELLÍN

MAYO DE 2024

Instrucciones:

Sigue los pasos a continuación.

1. Análisis de la base de datos Jardinería, consúltala aquí.
 - Revisa la estructura de la base de datos Jardinería para identificar las tablas relevantes y sus relaciones.
 - Identifica los campos necesarios para construir el modelo estrella, incluyendo la tabla de hechos y las dimensiones pertinentes.
2. Diseño del modelo estrella:
 - Diseña la estructura tabla de hechos que represente las ventas o transacciones de la empresa.
 - Identifica y diseña las dimensiones relevantes que se relacionarán con la tabla de hechos.
 - Diseña las relaciones entre la tabla de hechos y las dimensiones, asegurando la integridad referencial.
3. Documentación y presentación

INTRODUCCIÓN

Un Data Mart se establece como un sistema de almacenamiento de datos que está conformado por información específica de una unidad de una empresa. Se establece como una parte pequeña y específica de los datos más grandes.

Con este trabajo se pretende establecer un Data Mart a partir de una estructura de Base de Datos con datos, se realizará un análisis de la estructura para identificar la tabla de hechos y las tablas de dimensiones.

Continuando con el proceso, se va a crear una base de datos staging, basados en el Data Mart creado anteriormente, se poseionan unas tablas en la nueva base de datos, estas están determinadas por la procedencia de lo anterior, luego les introducimos la información pertinente, la extraemos de la base de datos jardinería principal, para con ellas poder tener nuestro propios datos en staging y así darle continuidad al trabajo.

OBJETIVOS

Construir un modelo estrella para un data mart que permita analizar y responder a tres categorías específicas: identificar el producto más vendido, la categoría con más productos y el año con más ventas, utilizando la base de datos Jardinería como fuente de datos.

Luego del modelo estrella, se construirá un base de datos staging, con tablas e información pertinente a nuestra necesidad de obtener información precisa para su respectivo uso en consultas posteriores.

PLANTEAMIENTO DEL PROBLEMA

Revisar la estructura de la Base de Datos con las tablas y sus relaciones para crear un Data Mart diseñando una estructura en Estrella, identificando la tabla de hechos y las tablas de dimensiones.

También construir una base de datos staging, creación de tablas y llenado de las mismas.

ANÁLISIS DEL PROBLEMA

Las características principales de esta gestión, es comprender el problema, identificar el contenido o la pregunta que necesitas resolver. Por ejemplo, puede ser analizar las ventas por región y producto.

Identifica las dimensiones relevantes para el análisis. En el ejemplo de ventas, las dimensiones podrían ser Producto Clientes, pagos, categorías y detalle de pedido.

Definir la Métrica: Decide la métrica principal que deseas analizar. En nuestro caso, podría ser el Total de Ventas en un año, los productos más vendidos, los mejores clientes entre otros cuestionamientos que nos podríamos hacer.

Combinar la tabla de hechos con las tablas de dimensiones y calcular la métrica deseada.

Itera y refina si es necesario, ajusta la consulta y repite el proceso de análisis para explorar diferentes aspectos del problema o mejorar la precisión de los resultados.

Este enfoque te permite analizar problemas de manera eficiente utilizando la estructura flexible y optimizada del modelo de estrella.

DESCRIPCIÓN DEL MODELO ESTRELLA

El Modelo Estrella es una técnica de modelo de datos que se utiliza para diseñar y optimizar almacenes de datos. El nombre se debe a la forma que tiene el esquema, que consta de una tabla principal la cual es llamada tabla de hechos y otras tablas periféricas llamadas tablas de dimensiones. Las tablas se relacionan entre sí a través de llaves primarias y llaves foráneas.

DISEÑO

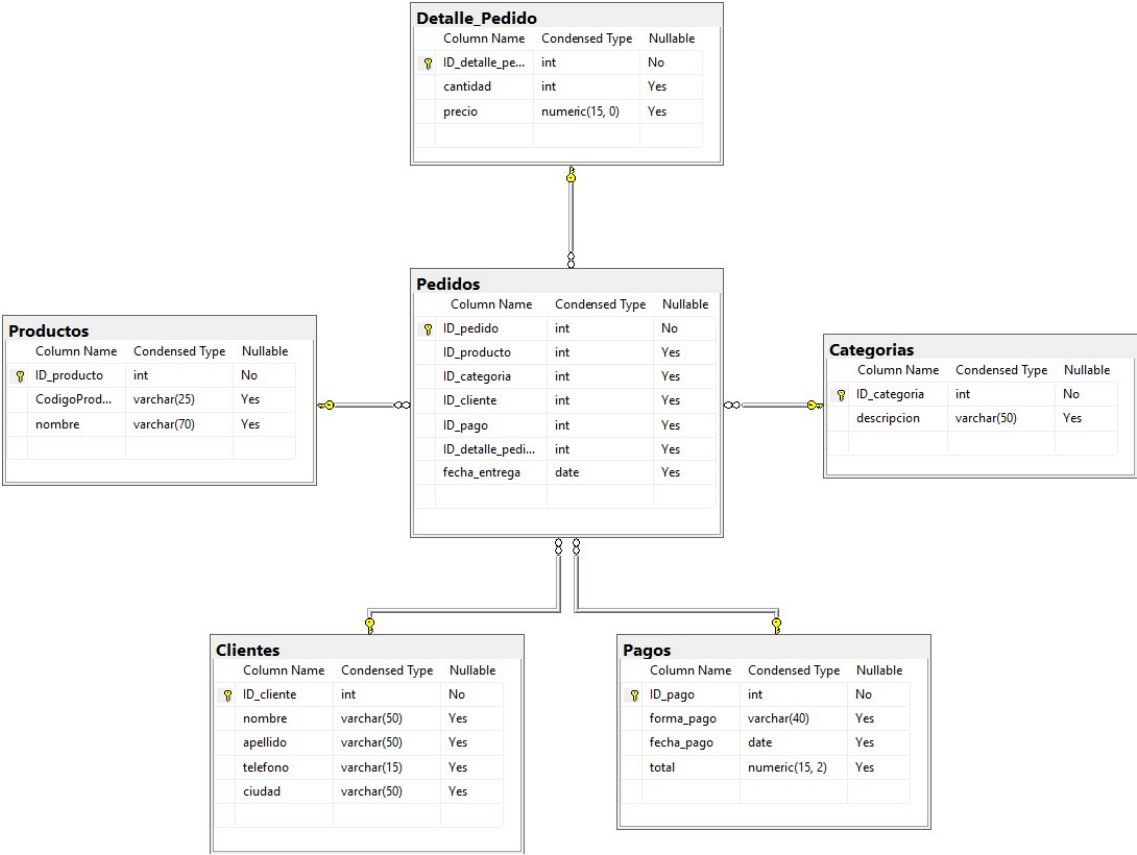


TABLA DE HECHOS

Tabla de Pedidos

La tabla de Pedidos fue seleccionada como la tabla de hechos, está compuesta por hechos relacionados con las operaciones de la empresa, adicional representas las transacciones que conforman el corazón del esquema de estrella

TABLAS DE DIMENSIONES

Tabla de Productos

Tabla de Clientes

Tabla de Pagos

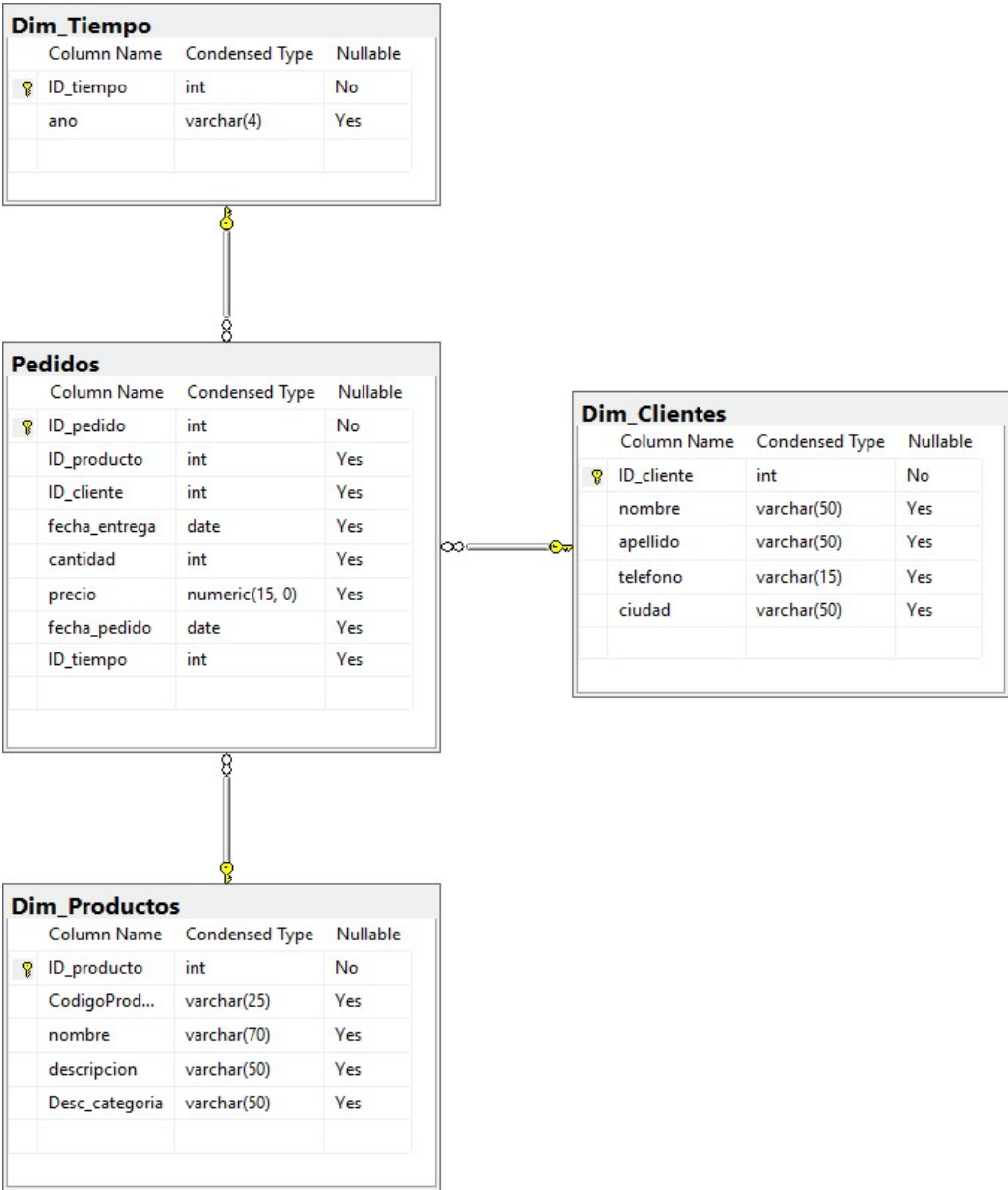
Tabla de Categorías

Tabla Detalle Pedido

Están conformadas por una variedad de información, la cual es útil al momento de realizar análisis y estudios para crear o mejorar una estrategia. Cada tabla de dimensiones se conecta a través de sus claves primarias hacia la tabla de hechos con claves foráneas

Evidencia de aprendizaje 2. Creación de una base de datos de Staging

Correcciones a la entrega 1.



Se realiza corrección al gráfico de la primera entrega, se elimina la tabla **Detalle_pedido** y se unifica con la tabla de hechos de **Pedidos**, también se elimina la tabla **Categoría**, la cual se unifica con la **dim_productos**, igualmente se elimina la tabla **Pagos**, adicional se crea la tabla **dim_tiempo**, esta tabla se crea a partir de la fecha de entrega de la tabla **Pedidos** extrayendo el año de ésta.

Descripción del análisis realizado a los datos Jardinería y cómo estos se trasladaron a la base de datos Staging.

Se comienza con la creación de las tablas en la base de datos staging de acuerdo con lo establecido en el modelo estrella, cada una se organiza teniendo en cuenta su clave primaria y sus respectivos ítems, para la información que solicitaremos a la base de datos jardinería que es la matriz de la cual nos alimentaremos, definidos la información a solicitar, se comienza con las consulta o script que necesitaremos,

la primera creada es la tabla de dimensiones (productos).

```
CREATE TABLE dim_productos (  
  ID_producto int identity(1,1),  
  CodigoProducto VARCHAR(25) NOT NULL,  
  nombre VARCHAR(70) NOT NULL,  
  descripcion text NOT NULL,  
  Desc_categoria VARCHAR(50) DEFAULT NULL,  
  PRIMARY KEY (ID_producto)
```

En esta tabla se asocia campos de las tablas productos y categoria_producto

continuamos con clientes.

```
CREATE TABLE dim_clientes (  
  ID_cliente int identity(1,1),  
  nombre_cliente varchar(50) NOT NULL,  
  direccion varchar(50) NOT NULL,  
  telefono varchar(15) NOT NULL,  
  ciudad varchar(50) NOT NULL,  
  region varchar(50),  
  pais varchar(50) NOT NULL,  
  PRIMARY KEY (ID_cliente)
```

Encontramos las características del cliente, con sus distintas informaciones para la plena identificación,

La siguiente tabla es tiempo.

```
CREATE TABLE dim_tiempo (  
    ID_tiempo int identity(1,1),  
    anio varchar(4) NOT NULL,  
    PRIMARY KEY (ID_tiempo)
```

En esta información encontraremos las estadísticas del tiempo en el cual se hicieron las ventas. La tabla se crea a partir de la fecha_entrega de la tabla pedidos, tomando el campo anio

Por último, se crea la tabla de hechos Pedidos.

```
CREATE TABLE pedidos (  
    ID_pedido int identity(1,1),  
    ID_producto int NOT NULL,  
    ID_cliente int NOT NULL,  
    ID_tiempo int NOT NULL,  
    fecha_entrega date,  
    fecha_pedido date,  
    cantidad int NOT NULL,  
    precio numeric(15,2) NOT NULL  
    PRIMARY KEY (ID_pedido),  
    FOREIGN KEY (ID_producto) REFERENCES dim_productos (ID_producto),  
    FOREIGN KEY (ID_cliente) REFERENCES dim_clientes (ID_cliente),  
    FOREIGN KEY (ID_tiempo) REFERENCES dim_tiempo (ID_tiempo)
```

Identificamos las claves primarias y las foráneas, que se unirán a las tablas de dimensiones,

Con las tablas creadas en Jardinería staging, comenzamos con los scripts para solicitar la información de la base de datos de Jardinería principal y con estos tener las declaraciones pertinentes en las tablas.

Se inicia con el script para trasladar la información hacia la tabla productos

```
--SCRIPT LLENAR DATOS DESDE JARDINERIA A JARDINERIASTAGING
INSERT INTO JardineriaStaging.dbo.dim_productos (CodigoProducto, nombre, descripcion, Desc_categoria)
SELECT p.CodigoProducto, p.nombre, p.descripcion, cp.Desc_Categoria
FROM jardineria.dbo.producto p
INNER JOIN jardineria.dbo.Categoria_producto cp ON cp.Id_Categoria = p.Categoria;
```

Se tiene por referencia para obtener la información de Jardinería principal la tabla producto la cual contiene la información necesaria para este fin.

Nos enfocamos en el segundo script, tabla clientes.

```
-----SCRIPT LLENAR DATOS DE CLIENTES
INSERT INTO JardineriaStaging.dbo.dim_clientes (nombre_cliente, direccion, telefono, ciudad, region, pais)
select nombre_cliente, linea_direccion1, telefono, ciudad, region, pais
from jardineria.dbo.cliente
```

Insertamos los campos, para ser llenados de Jardinería Staging, esta información que requerimos proviene de la tabla clientes de la base de datos jardinería principal.

Se continua con el script tiempo.

```
-----SCRIPT LLENAR DATOS DE TIEMPO
INSERT INTO JardineriaStaging.dbo.dim_tiempo (anio)
select distinct YEAR(fecha_entrega) as anio from jardineria.dbo.pedido where fecha_entrega is NOT NULL
```

El más corto de todos por la información que necesitamos de este, por el momento sólo nuestra necesidad es el año. En el traslado y llenado de la tabla se tiene en cuenta que algunos registros no tienen fecha lo que genera un campo NULL que se debe controlar

Se finaliza con el script de la tabla de hechos pedidos:

```
----SCRIPT LLENAR DATOS DE TABLA HECHOS PEDIDOS
INSERT INTO JardineriaStaging.dbo.pedidos (ID_producto, ID_cliente, ID_tiempo, fecha_entrega, fecha_pedido,
select ID_producto, ID_cliente, id_tiempo, fecha_entrega, fecha_pedido, cantidad, precio_unidad
from jardineria.dbo.pedido p
join jardineria.dbo.detalle_pedido dp
on dp.ID_pedido = p.ID_pedido
join JardineriaStaging.dbo.dim_tiempo t
on YEAR(fecha_entrega) = t.anio
```

Se hace la inserción a Jardinería staging, se selecciona cada uno de sus campos y se obtienen la información de las tablas que contienen los requerimientos necesarios para la sustracción de Jardinería principal.

Para el traslado de la información se tiene en cuenta que los registros que no tienen fecha entregada cuyo campo es NULL no pasan a la tabla de hechos de staging dado que se asocia cada registro según el año al cual pertenece en el campo de fecha entregada.

En conclusión, se crearon las tablas necesarias, se hace una abstracción de la información de la Base de datos Jardinería principal, se lleva a las tablas ya creadas y se inserta según la necesidad que tenemos para el caso de estudio que nos aqueja.

Evidencia de aprendizaje 3. Proceso de transformación de datos y carga en el data mart final

1. Transformación de datos según las necesidades analíticas:
 - a. Aplicar técnicas de transformación de datos, como limpieza, normalización y enriquecimiento, para preparar los datos de acuerdo con las necesidades analíticas específicas.

Para la entrega número 3 se realiza la transformación de los datos a las tablas que son requeridas

TABLA DIM_PRODUCTOS

Para la tabla dim_productos se realiza la transformación y la normalización, a continuación, se adjunta el script utilizado

```
select ID_producto, CodigoProducto, upper(nombre) Nombre_Producto,  
IF(LEN(descripcion) = 0, 'N/A', descripcion) as Descripcion  
, upper(Desc_categoria) Categoria  
  
from dim_productos
```

Se pasan los datos a mayúsculas de los campos nombre_producto y descripcion_categoria, para la descripción de la categoría los registros que no contienen información se le asigna N/A

TABLA DIM_CLIENTES

La siguiente tabla que se trabaja corresponde a dim_clientes, a continuación se adjunta el script

```
select ID_cliente, upper(nombre_cliente) as Nombre_Cliente, upper(direccion) as  
Direccion, telefono as Telefono, upper(ciudad) as Ciudad  
, upper(isnull(region, 'N/A')) as Region, upper(pais) as Pais  
  
from dim_clientes
```

Los campos nombre_cliente, direccion, ciudad, país se pasan a mayúscula, para el campo región se valida que a cada registro NULL se le asigna N/A

Para la tabla dim_tiempo y la tabla de hechos pedidos no se realiza ninguna transformación ya que no es requerida, todos los datos están normalizados

PRUEBAS DE DATOS

- Atendiendo en resolver la pregunta que se tiene desde el comienzo de la actividad “identificar el producto más vendido” podemos concluir que es el producto Cerezo.

Comparación

Consulta Base de Datos Jardinería

```

SQLQuery5.sql - DES...jardineria (sa (94))*  Script Entrega 3.sql...eriaStaging (sa (54))*
--producto mas vendido
select dp.ID_producto, pro.nombre, COUNT(dp.cantidad) as cantidad
from pedido p
inner join detalle_pedido dp on dp.ID_pedido = p.ID_pedido
inner join producto pro on pro.ID_producto = dp.ID_producto
group by dp.ID_producto, pro.nombre
order by 3 desc

```

	ID_producto	nombre	cantidad
1	87	Cerezo	28
2	4	Azadón	8
3	1	Sierra de Poda 400MM	7
4	3	Rastrillo de Jardín	7
5	26	Limonero 30/40	7
6	251	Chamaerops Humilis	7
7	271	Trachycarpus Fortunei	6
8	232	Tuja orientalis \"Aurea nana\"	6
9	181	Acer Pseudoplatanus	6
10	13	Thymus Vulgaris	6
11	2	Pala	6
12	107	Kaki	5
13	117	Melocotonero	5
14	109	Manzano	4
15	25	Peral	4
16	72	Peral Blanq. de Aranjuez	4
17	5	Ajedrea	4
18	15	Expositor Cítricos Mix	4
19	10	Petrosilium Hortense (Peregil)	4

Consulta Base de Datos JardineríaDataMart_Final

SQLQuery5.sql - DES...jardineria (sa (94))* Script Entrega 3.sql...eriaStaging (sa (54))*

```
select pro.ID_producto, pro.nombre, count(cantidad) as cantidad
from pedidos pe
inner join dim_productos pro on pe.ID_producto = pro.ID_producto
group by pro.ID_producto, pro.nombre
order by 3 desc
```

100 %

Results Messages

	ID_producto	nombre	cantidad
1	87	CEREZO	19
2	13	THYMUS VULGARIS	6
3	251	CHAMAEROPS HUMILIS	6
4	271	TRACHYCARPUS FORTUNEI	5
5	232	TUJA ORIENTALIS \"AUREA NANA\"	5
6	2	PALA	5
7	117	MELOCOTONERO	5
8	72	PERAL BLANQ. DE ARANJUEZ	4
9	4	AZADÓN	4
10	26	LIMONERO 30/40	4
11	19	OLEA-OLIVOS	3
12	57	NARANJO CALIBRE 8/10	3
13	5	AJEDREA	3
14	3	RASTRILLO DE JARDÍN	3
15	15	EXPOSITOR CÍTRICOS MIX	3
16	104	HIGUERA	3
17	106	KAKI	3
18	109	MANZANO	3
19	181	ACER PSEUDOPLATANUS	3
20	1	SIERRA DE PODA 400MM	3
21	180	ACER PLATANOIDES	2

CONCLUSION

Se presentan diferencias en las cantidades, pero corresponden a la normalización y validación de los datos, los cuales se encontraban campos NULL en algunas fechas, por lo que se toma como venta no realizada

- Dando respuesta a la segunda pregunta, “categoría con más productos” podemos concluir lo siguiente

Consulta Base de Datos Jardinería

SQLQuery5.sql - DES...jardineria (sa (94))* Script Entrega 3.sql...eriaStaging (sa (54))*

```

---categoria con mas productos
select Categoria, ct.Desc_Categoria, COUNT(codigoproducto) as cantidad
from producto p
inner join Categoria_producto ct on ct.Id_Categoria = p.Categoria
group by Categoria, ct.Desc_Categoria
order by 3 desc

```

100 %

Results Messages

	Categoria	Desc_Categoria	cantidad
1	5	Ornamentales	154
2	4	Frutales	108
3	3	Aromaticas	10
4	2	Herramientas	4

Consulta Base de Datos JardineríaDataMart_Final

SQLQuery5.sql - DES...jardineria (sa (94))* Script Entrega 3.sql...eriaStaging (sa (54))*

```

select Desc_categoria, count(CodigoProducto) as cantidad
from dim_productos
group by Desc_categoria
order by 2 desc

```

100 %

Results Messages

	Desc_categoria	cantidad
1	ORNAMENTALES	154
2	FRUTALES	108
3	AROMATICAS	10
4	HERRAMIENTAS	4

- Por último, resolvemos la pregunta “el año con más ventas” podemos concluir que corresponde al año 2009.

Comparación

Consulta Base de Datos Jardinería

SQLQuery5.sql - DES...jardineria (sa (94))* Script Entrega 3.sql...eriaStaging (sa (54))*

```

-----el año con mas ventas
select YEAR(fecha_pedido) as año, COUNT(cantidad) as cantidad
from pedido p
inner join detalle_pedido dp on dp.ID_pedido = p.ID_pedido
group by YEAR(fecha_pedido)
order by 2 desc

```

100 %

	año	cantidad
1	2009	180
2	2008	100
3	2007	29
4	2006	9

Consulta Base de Datos JardineríaDataMart_Final

SQLQuery5.sql - DES...jardineria (sa (94))* Script Entrega 3.sql...eriaStaging (

```

select anio, count(cantidad) as cantidad
from pedidos pe
inner join dim_tiempo ti on ti.ID_tiempo = pe.ID_tiempo
group by anio
order by 2 desc

```

100 %

	anio	cantidad
1	2009	128
2	2008	58
3	2007	20
4	2006	9

CONCLUSION

Se presentan diferencias en las cantidades, pero corresponden a la normalización y validación de los datos, los cuales se encontraban campos NULL en algunas fechas, por lo que se toma como venta con estado rechazada o pendiente

CONCLUSIONES

Basados en el ejercicio y de acuerdo con el modelo de estrella, la solicitud de información que queremos obtener, se crea una tabla de hechos con los productos, que es la fuente por la cual vamos a encontrar las respuesta al planteamiento solicitado, en esta tabla encontramos las asociaciones a las tablas de dimensión, las cuales nos permiten medir los fundamentos que queremos tener presente, esta son las de detalle de pedido, con esta tenemos básicamente los datos asociados a cada una de las solicitudes del cliente, la cual también está contenida, la tabla cliente, hace parte de este modelo y nos entrega los testimonios de nuestros compradores, tenemos la tabla pagos esta nos acerca a la forma y tiempos de retorno de los pagos, con un amplio aporte de respuesta que nos ayuda con los informes y por último la tabla categoría así con esta podemos dar un amplio contenido el cuál es nuestro rango o nivel del tamaño de la muestra, por todo lo anterior este modelo de estrella nos da amplios y precisos datos valiosos para la organización.

BIBLIOGRAFÍA

Soluciones Tecnológicas Tecon (s.f) Que es el Modelo Estrella?. [https://www.tecon.es/que-es-el-modelo-](https://www.tecon.es/que-es-el-modelo-estrella/#:~:text=El%20Modelo%20Estrella%20es%20una,perif%C3%A9ricas%20llamadas%20tablas%20de%20dimensiones.)

[estrella/#:~:text=El%20Modelo%20Estrella%20es%20una,perif%C3%A9ricas%20llamadas%20tablas%20de%20dimensiones.](https://www.tecon.es/que-es-el-modelo-estrella/#:~:text=El%20Modelo%20Estrella%20es%20una,perif%C3%A9ricas%20llamadas%20tablas%20de%20dimensiones.)

Keepcoding Tech School. Navarro Sandra. 18/04/2024. Modelos estrella y copo de nieve. <https://keepcoding.io/blog/modelos-estrella-y-copo-de-nieve/>