



Práctico 1

Modelado

Inteligencia de negocios

Micaela Olivera

Docentes: Lorena Laborde – Ignacio Chartier

2021

Ejercicio 1

La base de datos de la ADUANA está compuesta por las siguientes tablas principales:

- Transacción (ID_Transaccion, Fecha_Transaccion, RUT_Desp, RUT_Imp, Producto, Proveedor, Pais_Origen_Producto, Precio, Contenedor)
- Despachante (RUT, Nombre, Apellido, Domicilio_Fiscal)
- Importador (RUT, CI, Nombre, Apellido, Domicilio, Sexo, Fecha_Nacimiento)
- Producto (ID_Producto, Nombre_Producto, Descripcion, Pais_Origen, ID_Clase_Producto)
- Clase_Producto (ID_Clase_Producto, Descripcion)
- Contenedor (ID_Contenedor, Desc_Contenedor)
- Traza_Contenedor (ID_Contenedor, COD_Pais, COD_Puerto, Fecha)
- Proveedor (ID_Proveedor, Nombre_Proveedor, País_Proveedor)
- País (COD_Pais, Nombre_Pais)
- Puerto (COD_Puerto, Nombre_Puerto, COD_Pais)

A. Confeccionar un Data Warehouse (FTs y DTs) de la base relacional de arriba para responder las siguientes inquietudes:

1. Productos ingresados, por año, puerto y proveedor.
2. Precios de los productos con origen en China.
3. Países que realizan más importaciones, discriminadas por producto.
4. Despachante que más productos ingresa, discriminado por rubro de producto, origen y proveedor.

A) **Dimensiones:**

- País (DT_PAIS):
 - o SK_PAIS
 - o ID_PAIS
 - o NOMBRE
- Producto (DT_PRODUCTO):
 - o SK_PRODUCTO
 - o ID_PRODUCTO
 - o NOMBRE
 - o DESCRIPCION
 - o ID_PAIS_ORIGEN
 - o NOMBRE_PAIS
 - o ID_CLASE
 - o DESCRIPCION_CLASE
- Despachante (DT_DESPACHANTE):
 - o SK_DESPACHANTE
 - o ID_DESPACHANTE
 - o NOMBRE
 - o APELLIDO
 - o DOMICILIO
- Proveedor (DT_PROVEEDOR):
 - o SK_PROVEEDOR
 - o ID_PROVEEDOR
 - o NOMBRE
 - o ID_PAIS_PROVEEDOR

- NOMBRE_PAIS
- Puerto (DT_PUERTO)
 - SK_PUERTO
 - ID_PUERTO
 - NOMBRE
 - ID_PAIS_PUERTO
 - NOMBRE_PUERTO
- Contenedor (DT_CONTENEDOR)
 - SK_CONTENEDOR
 - ID_CONTENEDOR
 - DESCRIPCION
- Importador (DT_IMPORTADOR)
 - SK_IMPORTADOR
 - ID_IMPORTADOR
 - CI
 - NOMBRE
 - APELLIDO
 - DOMICILIO
 - SEXO
 - FECHA_NAC

Hechos:

- Transacción (FT_TRANSACCION)
 - FECHA_TRANSACCION
 - SK_DESPACHANTE
 - SK_IMPORTADOR
 - SK_PRODUTO
 - SK_PROVEEDOR
 - SK_PAIS_ORIGEN
 - SK_CONTENEDOR
 - PRECIO
 - CANT (=1)
- Traza (FT_TRAZA)
 - SK_CONTENEDOR
 - SK_PAIS
 - SK_PUERTO
 - FECHA_TRAZA

1)

SELECT

YEAR(FECHA_TRANSACCION), PRODUCTO.NOMBRE, PUERTO.NOMBRE,
PROVEEDOR.NOMBRE, COUNT (*)

FROM

FT_TRANSACCION, DT_PUERTO, DT_PROVEEDOR, DT_PRODUCTO

WHERE

TRANSACCION.SK_PRODUCTO = DT_PRODUCTO.SK_PRODUCTO
TRANSACCION.SK_PROVEEDOR = DT_PROVEEDOR.SK_PROVEEDOR
TRANSACCION.SK_PUERTO = DT_PUERTO.SK_PUERTO
AND YEAR(FECHA_TRANSACCION) = YYYY

```
GROUP BY
    YEAR(FECHA_TRANSACCION), PRODUCTO.NOMBRE, PUERTO.NOMBRE,
    PROVEEDOR.NOMBRE
```

2)

```
SELECT
    PRECIO, PRODUCTO.NOMBRE
FROM
    TRANSACCION, DT_PRODUCTO
WHERE
    TRANSACCION.SK_PRODUCTO = DT_PRODUCTO.SK_PRODUCTO
AND
    DT_PRODUCTO.NOMBRE_PAIS = "CHINA"
```

3)

```
SELECT
    PAIS.NOMBRE, PRODUCTO.NOMBRE, IMPORTADOR.NOMBRE, COUNT
    (IMPORTADOR.NOMBRE) AS CANTIDADES
FROM
    TRANSACCION, DT_PAIS, DT_PRODUCTO
WHERE
    TRANSACCION.SK_PAIS_ORIGEN = DT_PAIS.SK_PAIS
    TRANSACCION.SK_PRODUCTO = DT_PRODUCTO.SK_PRODUCTO
    TRANSACCION.SK_IMPORTADOR = DT_IMPORTADOR.SK_IMPORTADOR
ORDER BY
    IMPORTADOR.NOMBRE
GROUP BY
    CANTIDADES
```

4)

```
SELECT
    DESPACHANTE.NOMBRE, COUNT (DESPACHANTE.NOMBRE) AS DESPACHANTE,
    PRODUCTO.NOMBRE, PRODUCTO.DESCRIPCION_CLASE, PRODUCTO.NOMBRE_PAIS,
    PROVEEDOR.NOMBRE
FROM
    TRANSACCION, DT_DESPACHANTE, DT_PRODUCTO, DT_PROVEEDOR
WHERE
    TRANSACCION.SK_DESPACHANTE = DT_DESPACHANTE.SK_DESPACHANTE
    TRANSACCION.SK_PRODUCTO = DT_PRODUCTO.SK_PRODUCTO
    TRANSACCION.SK_PRODUCTO = DT_PROVEEDOR

GROUP BY
    DESPACHANTE
ORDER BY
    PRODUCTO.DESCRIPCION_CLASE, PRODUCTO.NOMBRE_PAIS, PROVEEDOR.NOMBRE
```

Ejercicio 2

Diseñar el DWH para una empresa de muebles “UCU-Amoblamientos”, al por mayor. El DWH tiene que permitir analizar la situación de la empresa, al menos con respecto a los muebles, los clientes y el tiempo.

Además, la empresa tiene que analizar:

- Los muebles con respecto a su tipo (silla, mesa, armario, mueble ...), categoría (cocina, salón, dormitorio, baño, oficina ...) y materiales (madera, mármol ...)
- Los clientes con respecto a su ubicación espacial, considerando, al menos, ciudades, regiones y estados

Los tres socios de la empresa, Daniel, Héctor y Bernardo, están interesados en aprender, al menos, la cantidad, el ingreso y el descuento de sus ventas.

A. Identificar facts, dimensiones y medidas.

Dimensiones:

- DT_MUEBLE:
 - SK_MUEBLE
 - ID_MUEBLE
 - ID_TIPO
 - TIPO_NOMBRE
 - ID_CATEGORIA
 - NOMBRE_CATEGORIA
 - ID_MATERIAL
 - MATERIAL_NOMBRE
- DT_CLIENTE:
 - SK_CLIENTE
 - ID_CLIENTE
 - CLIENTE_DIRECCION
 - ID_CIUDAD
 - CIUDAD_NOMBRE
 - ID_PAIS
 - PAIS_NOMBRE
- DT_TIPO:
 - SK_TIPO
 - ID_TIPO
 - TIPO_NOMBRE
- DT_CATEGORIA:
 - SK_CATEGORIA
 - ID_CATEGORIA
 - CATEGORIA_NOMBRE
- DT_MATERIAL:
 - SK_MATERIAL
 - ID_MATERIAL
 - MATERIAL_NOMBRE
- DT_CIUDAD:
 - SK_CUIDAD
 - ID_CIUDAD

- CIUDAD_NOMBRE
- ID_ESTADO
- ESTADO_NOMBRE
- DT_ESTADO:
 - SK_ESTADO
 - ID_ESTADO
 - ESTADO_NOMBRE
 - ID_PAIS_REGION
 - PAIS_REGION_NOMBRE
- DT_PAIS_REGION:
 - SK_PAIS_REGION
 - ID_PAIS_REGION
 - PAIS_REGION_NOMBRE

Facts:

- FT_VENTA:
 - SK_CLIENTE
 - SK_MUEBLE
 - FECHA
 - CANTIDAD
 - GANANCIA
 - DESCUENTO

Medidas:

- Cantidad de ventas por país.
- Cantidad de ventas por estado.
- Cantidad de ventas por ciudad.
- Cantidad de clientes por ciudad.
- Cantidad de clientes por estado.
- Cantidad de clientes por país.
- Descuentos aplicados al precio de un mueble.
- Cantidad de ingreso (ganancia) por mercadería.

B Justificar la elección de esquema estrella o esquema Snowflake.

Se selecciona un esquema Snowflake dado que, por el posible gran volumen de información, de esta forma se evita la redundancia de datos y con ello se ahorra espacio.

C. Resolver en SQL las siguientes consultas:

1. Cómo encuentro la cantidad, el total de ingresos y el descuento con respecto a cada ciudad, el tipo de muebles y el mes

```

1  ---1. Cómo encuentro la cantidad, el total de ingresos y el descuento con respecto a cada ciudad, el tipo de muebles y el mes
2
3  SELECT
4      dt_cliente.ciudad_nombre, dt_mueble.tipo_nombre, fk_venta.descuento , SUM(ganancia) AS ganancia, fk_venta.id_cliente,
5      fk_venta.fecha
6  FROM
7      fk_venta
8  INNER JOIN dt_cliente
9      ON fk_venta.sk_cliente = dt_cliente.sk_cliente
10
11 INNER JOIN dt_mueble
12     ON fk_venta.sk_mueble = dt_mueble.sk_mueble
13 GROUP BY
14     fk_venta.cantidad, fk_venta.descuento, fk_venta.id_cliente, dt_cliente.ciudad_nombre, dt_mueble.tipo_nombre, fk_venta.fecha
15
16 ORDER BY
17     fk_venta.fecha

```

2. Cómo encuentro la cantidad promedio, el ingreso y el descuento con respecto a cada país, material de los muebles y el año

```

19 ---2. Cómo encuentro la cantidad promedio, el ingreso y el descuento con respecto a cada país, material de los muebles y el año
20
21 SELECT
22     AVG(cantidad) AS promedio, SUM(ganancia) AS ingreso, fk_venta.descuento, dt_cliente.pais_nombre, dt_mueble.tipo_nombre,
23     fk_venta.fecha
24 FROM
25     fk_venta
26 INNER JOIN dt_cliente
27     ON fk_venta.sk_cliente = dt_cliente.sk_cliente
28
29 INNER JOIN dt_mueble
30     ON fk_venta.sk_mueble = dt_mueble.sk_mueble
31 GROUP BY
32     fk_venta.descuento, dt_cliente.pais_nombre, dt_mueble.tipo_nombre, fk_venta.fecha
33
34 ORDER BY
35     fk_venta.fecha
36

```

3. Determinar los 5 muebles más vendidos durante el mes de mayo

```

38 ---3. Determinar los 5 muebles más vendidos durante el mes de mayo
39
40 SELECT
41     dt_mueble.tipo_nombre, MAX(ganancia) AS ganancia, fk_venta.fecha
42 FROM fk_venta
43 INNER JOIN dt_mueble
44     ON fk_venta.sk_mueble = dt_mueble.sk_mueble
45 WHERE
46     fk_venta.fecha = 'mayo'
47 GROUP BY
48     dt_mueble.tipo_nombre, fk_venta.fecha
49 ORDER BY MAX(ganancia) DESC
50 LIMIT 5
51

```

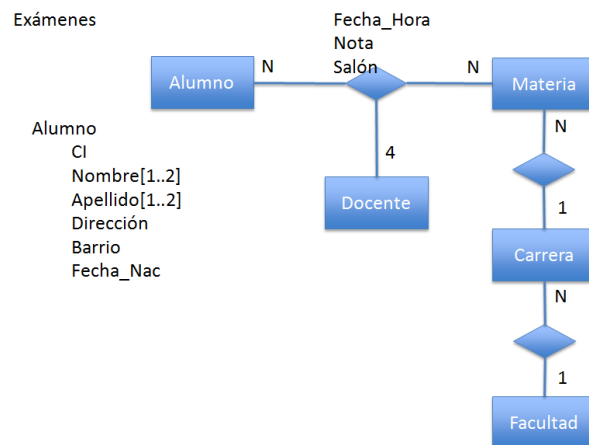
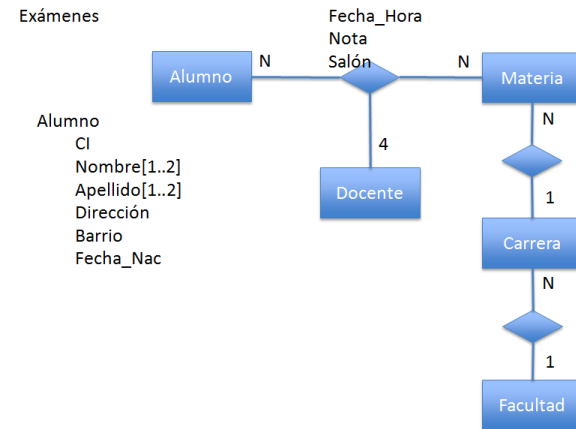
Ejercicio 3

Se plantea la creación de una solución de BI para una Universidad, a partir de la existencia de varios sistemas transaccionales que deberán ser integrados para la creación de este.

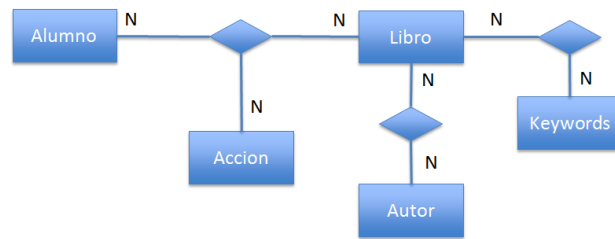
Dicha Universidad cuenta con 3 sistemas principales:

1. Sistema de inscripción de alumnos a exámenes
2. Sistema de inscripción de alumnos a materias
3. Sistema de reserva y préstamo de libros en biblioteca

Los 3 modelos de datos de los sistemas son los que se detallan a continuación en los siguientes MER.



Biblioteca



La universidad desea conocer, por ejemplo:

- A. Cantidad de alumnos inscriptos por materia.
- B. Cantidad de alumnos inscriptos a exámenes.
- C. Porcentaje de alumnos que aprueban, exoneran o pierden un curso.
- D. Libros más requeridos.
- E. Periodo promedio que se presta un libro.

Dimensiones:

- DT_ALUMNO
 - SK_ALUMNO
 - ID_ALUMNO
 - NOMBRE
 - APELLIDO
- DT_LIBRO
 - SK_LIBRO
 - ID_LIBRO
 - NOMBRE_AUTOR
 - NOMBRE_LIBRO
 - ESTADO
- DT_MATERIA
 - SK_MATERIA
 - ID_MATERIA
 - NOMBRE_CARRERA
 - NOMBRE_FACULTAD
 - NOMBRE_DOCENTE

Hechos:

- FT_INSCRIPCION_MATERIA
 - SK_ALUMNO
 - SK_MATERIA
 - NOTA_FINAL
- FT_INSCRIPCION_EXAMEN
 - SK_ALUMNO
 - SK_MATERIA
 - NOTA_FINAL
- FT_BIBLIOTECA_PRESTAMOS
 - SK_LIBRO

- SK_ALUMNO
- TIEMPO_PRESTAMO
- VECES_PRESTADO

A. Cantidad de alumnos inscriptos por materia.

```
SELECT
    COUNT (SK_ALUMNO) AS INSCRIPTOS, NOMBRE_MATERIA
FROM
    FT_INCRIPCION_MATERIA, DT_MATERIA
WHERE
    FT_INCRIPCION_MATERIA.SK_MATERIA = DT_MATERIA.SK_MATERIA
GROUP BY
    NOMBRE_MATERIA
```

B. Cantidad de alumnos inscriptos a exámenes.

```
SELECT
    COUNT (SK_ALUMNO) AS INSCRIPTOS, NOMBRE_MATERIA
FROM
    FT_INCRIPCION_EXAMEN, DT_MATERIA
WHERE
    FT_INCRIPCION_EXAMEN.SK_MATERIA = DT_MATERIA.SK_MATERIA
GROUP BY
    NOMBRE_MATERIA
```

C. Porcentaje de alumnos que aprueban, exoneran o pierden un curso.

```
SELECT
    NOMBRE_MATERIA,
    COUNT(NOTA_FINAL) WHEN NOTA_FINAL = EXONERADO *100 / SELECT (*) FROM
    FT_INCRIPCION_MATERIA AS EXONERADOS,
    COUNT(NOTA_FINAL) WHEN NOTA_FINAL = APROBADOS *100 / SELECT (*) FROM
    FT_INCRIPCION_MATERIA AS APROBADOS,
    COUNT(NOTA_FINAL) WHEN NOTA_FINAL = REPROBADO *100 / SELECT (*) FROM
    FT_INCRIPCION_MATERIA AS REPROBADO,
FROM
    FT_INSCRIPCION_MATERIA, DT_MATERIA
WHERE
    FT_INCTRIPCION_MATERIA.SK_MATERIA = DT_MATERIA.SK_MATERIA
GROUP BY
    NOMBRE_MATERIA
```

D. Libros más requeridos.

```
SELECT
    NOMBRE_LIBRO, COUNT(VECES_PRESTADO) AS PRESTAMOS
FROM
    FT_BIBLIOTECA_PRESTAMOS, DT_LIBRO
WHERE
    FT_BIBLIOTECA_PRESTAMOS.SK_LIBRO = DT_LIBRO.SK_LIBRO
ORDER BY
    PRESTAMOS
```

E. Periodo promedio que se presta un libro.

```
SELECT
    NOMBRE_LIBRO, AVG(TIEMPO_PRESTADO) AS PROMEDIO
FROM
    FT_BIBLIOTECA_PRESTAMOS, DT_LIBRO
WHERE
    FT_BIBLIOTECA_PRESTAMOS.SK_LIBRO = DT_LIBRO.SK_LIBRO
AND
    NOMBRE_LIBRO = DT_LIBRO.NOMBRE_LIBRO
```

Ejercicio 4

UCU-Sport es una empresa cuya actividad se centra en la venta al por mayor. Cuenta con agencias en todo Uruguay. Cada agencia tiene un número de agentes que están asignados un distrito venta. La mayoría de los clientes son cadenas de tiendas con muchas ramas en diferentes provincias y regiones de todo el país.

Una venta se compone de un recibo con una o más filas, cada una representando unidades de mercancías vendidas. La venta de UCU-Sport dependerá en gran medida de la temporada, por lo tanto, el volumen de negocios de la compañía cambia a lo largo del año para diferentes productos.

Cada año se compone principalmente de dos temporadas de venta: (otoño e invierno) y (primavera y verano). Cada estación del año la venta comienza y termina en diferentes meses con respecto a los pedidos y la facturación.

La empresa tiene como objetivo la agregación de información de sus agencias en un DWH. Los usuarios del DWH están interesados en dos eventos principales: el volumen de negocios (es decir, las ventas) y los pedidos.

El DWH debe estar diseñado para responder con prontitud a las consultas realizadas por los directivos de la empresa.

Algunas de las consultas que los gerentes están interesados en poder responder son las siguientes.

a. Volumen de negocios (ventas):

1. Las ventas para cada provincia de forma mensual, trimestral, semestral, y anual.
2. Las ventas para cada región de forma mensual, trimestral, semestral, y anual.
3. Las ventas para cada cliente de forma mensual, trimestral, semestral, y anual.
4. Las ventas para cada grupo de clientes de forma mensual, trimestral, semestral, y anual.
5. Las ventas para cada agente de forma mensual, trimestral, semestral, y anual.
6. Las ventas para cada agencia de forma mensual, trimestral, semestral, y anual.
7. Las ventas para cada categoría de artículo de forma mensual, trimestral, semestral, y anual.
8. Las ventas para cada agencia, para cada mes y para cada categoría.
9. Número de artículos vendidos para cada categoría, período de tiempo (mensual, trimestral, semestral, y anual), cliente y agencia
10. El valor de descuento absoluto (no porcentaje) en función de la categoría del artículo, período de tiempo (mensual, trimestral, semestral, y anual), el cliente y la agencia.

b. Órdenes (Pedidos)

1. Total del precio de los pedidos para cada provincia de forma mensual, trimestral, semestral, y anual.
2. Total del precio de los pedidos para cada categoría de producto de acuerdo con la temporada de venta (otoño e invierno o primavera y verano) para cada provincia.
3. Total del precio de los pedidos mensual para cada categoría de artículo, cliente y agente.
4. Plazo de entrega medio de las diferentes categorías de producto para cada cliente, provincia, agencia y mes.

TAREA-1. Diseñar el DWH para UCU-Sport.

Dimensiones

- DT_AGENCIA
 - SK_AGENCIA
 - ID_AGENCIA
 - ID_AGENTE
 - NOMBRE_AGENTE
 - NOMBRE
 - ID_TIENDA
 - ID_REGION
 - ID_PAIS
- DT_AGENTE
 - SK_AGENTE
 - ID_AGENTE
 - NOMBRE
- DT_CLIENTE
 - SK_CLIENTE
 - ID_CLIENTE
 - NOMBRE
 - ID_GRUPO_CLIENTE
 - NOMBRE_GRUPO
- DT_GRUPO_CLIENTE
 - SK_GRUPO_CLIENTE
 - ID_GRUPO_CLIENTE
 - NOMBRE
- DT_TIENDA
 - SK_TIENDA
 - ID_TIENDA
 - NOMBRE
 - ID_CLIENTE
 - ID_REGION
 - ID_PAIS
 - ID_PROVINCIA
- DT_PROVINCIA
 - SK_PROVINCIA
 - ID_PROVINCIA
 - NOMBRE
- DT_REGION
 - SK_REGION
 - ID_REGION
 - NOMBRE
- DT_PAIS
 - SK_PAIS
 - ID_PAIS
 - NOMBRE
- DT_MERCADERIA
 - SK_MERCADERIA

- ID_MERCADERIA
- DESCRIPCION
- ID_CATEGORIA
- NOMBRE_CATEGORIA
- DT_TEMPORADA
 - SK_TEMPORADA
 - ID_TEMPORADA
 - NOMBRE
- DT_CATEGORIA
 - SK_CATEGORIA
 - ID_CATEGORIA
 - NOMBRE

Hechos

- FT_VENTA
 - SK_AGENCIA
 - SK_CLIENTE
 - SK_TIENDA
 - SK_REGION
 - SK_PAIS
 - SK_MERCADERIA
 - SK_CATEGORIA
 - SK_TEMPORADA
 - PRECIO
 - FECHA
 - DESCUENTO
- FT_PEDIDO
 - SK_REGION
 - SK_PROVINCIA
 - SK_CATEGORIA
 - SK_CLIENTE
 - SK_AGENTE
 - SK_TEMPORADA
 - TIEMPO_ENTREGA
 - PRECIO
 - FECHA_ENTREGA

A. Ventas

1. Las ventas para cada provincia de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    PROVINCIA.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_PROVINCIA
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
  
```

```

        FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
        FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
        FT_VENTA.SK_PROVINCIA = DT_PROVINCIA.SK_PROVINCIA
GROUP BY
    PROVINCIA.NOMBRE
ORDER BY
    FECHA

```

2. Las ventas para cada región de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    REGION.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_REGION
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_REGION = DT_REGION.SK_REGION
GROUP BY
    REGION.NOMBRE
ORDER BY
    FECHA

```

3. Las ventas para cada cliente de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    REGION.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_REGION
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_REGION = DT_REGION.SK_REGION
GROUP BY
    CLIENTE.NOMBRE
ORDER BY
    FECHA

```

4. Las ventas para cada grupo de clientes de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    REGION.NOMBRE, CLIENTE.NOMBRE_GRUPO

```

```

FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_REGION
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_REGION = DT_REGION.SK_REGION
GROUP BY
    CLIENTE.NOMBRE_GRUPO
ORDER BY
    FECHA

```

5. Las ventas para cada agente de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    AGENTE.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_AGENTE
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_AGENTE = DT_AGENTE.SK_AGENTE
GROUP BY
    AGENTE.NOMBRE
ORDER BY
    FECHA

```

6. Las ventas para cada agencia de forma mensual, trimestral, semestral, y anual.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    AGENCIA.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_AGENCIA
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_AGENCIA = DT_AGENCIA.SK_AGENCIA
GROUP BY
    AGENCIA.NOMBRE
ORDER BY
    FECHA

```

7. Las ventas para cada categoría de artículo de forma mensual, trimestral, semestral, y anual.


```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    CATEGORIA.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_AGENCIA
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_CATEGORIA = DT_CATEGORIA.SK_CATEGORIA
GROUP BY
    CATEGORIA.NOMBRE
ORDER BY
    FECHA

```

8. Las ventas para cada agencia, para cada mes y para cada categoría.

```

SELECT
    TIENDA.NOMBRE, CLIENTE.NOMBRE, MERCADERIA.NOMBRE, PRECIO,
    AGENCIA.NOMBRE
FROM
    FT_VENTA, DT_TIENDA, DT_CLIENTE, DT_MERCADERIA, DT_AGENCIA
WHERE
    FT_VENTA.SK_TIENDA = DT_TIENDA.SK_TIENDA
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_MERCADERIA = DT_MERCADERIA.SK_MERCADERIA
    FT_VENTA.SK_AGENCIA = DT_AGENCIA.SK_AGENCIA
GROUP BY
    AGENCIA.NOMBRE
ORDER BY
    FECHA

```

9. Número de artículos vendidos para cada categoría, período de tiempo (mensual, trimestral, semestral, y anual), cliente y agencia

```

SELECT
    CLIENTE.NOMBRE, AGENCIA.NOMBRE COUNT (SK_MERCADERIA) AS CANTIDAD
FROM
    FT_VENTA, DT_CLIENTE, DT_AGENCIA
WHERE
    FT_VENTA.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_VENTA.SK_AGENCIA = DT_AGENCIA.SK_AGENCIA
ORDER BY
    FECHA

```

10. El valor de descuento absoluto (no porcentaje) en función de la categoría del artículo, período de tiempo (mensual, trimestral, semestral, y anual), el cliente y la agencia.

```

SELECT
    DESCUENTO
FROM
    FT_VENTA
WHERE
    FT_VENTA.SK_CATEGORIA = XXX
AND
    FT_VENTA.SK_CLIENTE = XXX
AND
    FT_VENTA.SK_AGENCIA = XXX
AND
    FECHA = XXX

```

B. Ordenes

1. Total del precio de los pedidos para cada provincia de forma mensual, trimestral, semestral, y anual.

```

SELECT
    SUM (PRECIO), PROVINCIA.NOMBRE
FROM
    FT_PEDIDO, DT_PROVINCIA
WHERE
    FT_PEDIDO.SK_PROVINCIA = DT_PROVINCIA.SK_PROVINCIA
AND
    FECHA_ENTREGA = XXX

```

2. Total del precio de los pedidos para cada categoría de producto de acuerdo con la temporada de venta (otoño e invierno o primavera y verano) para cada provincia.

```

SELECT
    SUM(PRECIO), CATEORIA.NOMBRE, TEMPORADA.NOMBRE, PROVICIA.NOMBRE
FROM
    FT_PEDIDO, DT_CATEGORIA, DT_PROVINCIA, DT_TEMPORADA
WHERE
    FT_PEDIDO.SK_CATEGORIA = DT_CATEGORIA.SK_CATEGORIA
    FT_PEDIDO.SK_ PROVINCIA = DT_ PROVINCIA.SK_ PROVINCIA
    FT_PEDIDO.SK_ TEMPORADA = DT_CATEGORIA.SK_ TEMPORADA
ORDER BY
    TEMPORADA.NOMBRE

```

3. Total del precio de los pedidos mensual para cada categoría de artículo, cliente y agente.

```

SELECT
    SUM(PRECIO), CATEGORIA.NOMBRE, CLIENTE.NOMBRE, AGENTE.NOMBRE
FROM
    FT_PEDIDO, DT_ CATEGORIA, DT_ CLIENTE, DT_ AGENTE
WHERE

```

```
FT_PEDIDO.SK_CATEGORIA = DT_CATEGORIA.SK_CATEGORIA
FT_PEDIDO.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
FT_PEDIDO.SK_AGENTE = DT_AGENTE.SK_AGENTE
ORDER BY
    CATEGORIA.NOMBRE
```

4. Plazo de entrega medio de las diferentes categorías de producto para cada cliente, provincia, agencia y mes.

```
SELECT
    AVG(TIEMPO_ENTREGA) AS PROMEDIO, CATEGORIA.NOMBRE, PROVINCIA.NOMBRE,
    CLIENTE.NOMBRE, AGENCIA.NOMBRE
FROM
    FT_PEDIDO, DT_CATEGORIA, DT_PROVINCIA, DT_CLIENTE, DT_AGENCIA
WHERE
    FT_PEDIDO.SK_CATEGORIA = DT_CATEGORIA.SK_CATEGORIA
    FT_PEDIDO.SK_CLIENTE = DT_CLIENTE.SK_CLIENTE
    FT_PEDIDO.SK_PROVINCIA = DT_PROVINCIA.SK_PROVINCIA
    FT_PEDIDO.SK_AGENCIA = DT_AGENCIA.SK_AGENCIA

ORDER BY
    CATEGORIA.NOMBRE
```

TAREA-2. COMPLEMENTARIO –

Teniendo en cuenta las características y las cardinalidades del DWH diseñado, decidir que vistas materializadas (tablas agregadas), particiones e índices pueden ser útiles para optimizar el rendimiento de las siguientes consultas.

- Q1. Las ventas trimestrales, semestrales y anuales para cada provincia.
 - Q2. Las ventas trimestrales, semestrales y anuales para cada región.
 - Q3. Las ventas trimestrales, semestrales y anuales para cada categoría de producto y provincia.
 - Q4. Precio del pedido total para cada categoría de producto de acuerdo con la temporada de venta para cada provincia.
 - Q5. Tiempo de entrega de las diferentes categorías de artículos para cada región y mes.
- Para evaluar el beneficio de las vistas materializadas, se conocen las siguientes estadísticas.

- Clientes

- o Clientes ≈ 1.000
- o Provincias ≈ 100
- o Regiones ≈ 20
- o Grupos de clientes ≈ 4

- Agentes

- o Agentes ≈ 100
- o Agencias ≈ 30

- Artículos

- o Categorías de artículos ≈ 10

El peor de los casos tiene las siguientes cardinalidades para los hechos.

- Ventas

- o 1.000 clientes
- o 12 meses
- o 10 categorías de producto
- o 100 agentes
- o Total $\approx 12 \times 10^6$ (12 millones)

- Los pedidos

- o 1.000 clientes
- o 360 días
- o 10 categorías de producto
- o 100 agentes
- o Total $\approx 360 \times 10^6$ (360 millones)

Ejercicio 5

Se quiere armar un Data Warehouse de un campeonato de Fútbol para poder llevar las estadísticas históricas de los partidos, jugadores, etc.

Asimismo, la Federación de Fútbol tiene un sistema en el que va cargando las incidencias de los partidos. Las tablas que usa el sistema son las siguientes:

- Jugador (CI_Jugador, Nombre_Jugador, Apellido_Jugador, Fecha_Nacimiento, COD_Posicion)
- Técnico (CI_Técnico, Nombre_Técnico, Apellido_Técnico, Fecha_Nacimiento)
- Arbitro (CI_Árbitro, Nombre_Árbitro, Apellido_Árbitro, Fecha_Nacimiento, Nacionalidad)
- Posicion_Jugador (COD_Posicion, Descripcion_Posicion)
- Equipo (COD_Equipo, Nombre, Sede, Fecha_Fundacion, COD_Estadio, Color_Camiseta_Titular, Color_Camiseta_Alternativa)
- Estadio (COD_Estadio, Nombre)
- Jugador_Equipo (CI_Jugador, COD_Equipo, Fecha_Desde, Fecha_Hasta)
- Técnico_Equipo (CI_Técnico, COD_Equipo, Fecha_Desde, Fecha_Hasta)
- Partido (ID_Partido, COD_Local, COD_Visitante, COD_Estadio, Camiseta_Local, Camiseta_Visitante, Goles_Local, Goles_Visitante)
- Arbitro_Partido (CI_Arbitro, ID_Partido)
- Alineaciones (ID_Partido, COD_Equipo, CI_Jugador, Minuto_Entra, Minuto_Sale)
- Tarjetas_Partido (ID_Partido, COD_Equipo, CI_Jugador, Minuto_Tarjeta, Color_Tarjeta)
- Goles_Partido (ID_Partido, COD_Equipo, CI_Jugador, Minuto)
- Asisntecias_Partido (ID_Partido, COD_Equipo_Generante, CI_Jugador_Pasador, CI_Jugador_Definidor, Minuto)
- Penales_Partido (ID_Partido, COD_Equipo_Generante, CI_Jugador_Cometedor, CI_Jugador_Recibidor, Minuto)

Se quiere responder a preguntas puntuales de partidos, así como también a una estadística global de los mismos, tarjetas, córner, penales, incidencias importantes, qué jugadores fueron los más influyentes en determinados momentos de los partidos (Por ejemplo, del minuto 80 al 90), etc.

1. Identifique problemas que vea del modelo de datos.

Se identifica el problema de que la base de datos de la Federación tiene un exceso de tablas, es decir que hay mucha información dispersa, dado que alguna información del partido podría ser agrupada en una tabla sola con todos los eventos del partido. Esto dificulta al momento de analizar la información y generar el datawarehouse apropiado para responder las preguntas del negocio.

2. Diseñe un DWH para poder analizar la información que se pide.

Dimensiones

- DT_JUGADOR
 - SK_JUGADOR
 - ID_JUGADOR
 - NOMBRE
 - POSICION
- DT_TECNICO

- SK_TECNICO
 - ID_TECNICO
 - NOMBRE
- DT_ARBITRO
 - SK_ARBITRO
 - ID_ARBITRO
 - NOMBRE
- DT_EQUIPO
 - SK_EQUIPO
 - ID_EQUIPO
 - NOMBRE
 - SEDE
 - FECHA_FUNDACION
- DT_ESTADIO
 - SK_ESTADIO
 - ID_ESTADIO
 - NOMBRE

Hechos

- FT_PARTIDO
 - SK_TECNICO_VISITANTE
 - SK_TECNICO_LOCAL
 - SK_ARBITRO
 - SK_EQUIPO_VISITANTE
 - SK_EQUIPO_LOCAL
 - SK_ESTADIO
 - TOTAL_CAMBIOS
 - TOTAL_FALTAS
 - TOTAL_GOLES
 - TOTAL_TARJETAS
 - TOTAL_PENALES
 - TOTAL_LESIONADOS
- FT_GOL
 - SK_JUGADOR
 - SK_EQUIPO_JUGADOR
 - SK_EQUIPO_CONTRINCANTE
 - SK_ESTADIO
 - SK_TECNICO_JUGADOR
 - SK_TECNICO_CONTRINCANTE
 - SK_ARBITRO
 - MINUTO_GOL
 - POSICION_GOL
- FT_ASISTENCIA
 - SK_JUGADOR
 - SK_JUGADOR_ASISTIDO
 - SK_EQUIPO_JUGADOR
 - SK_EQUIPO_CONTRINCANTE
 - SK_ESTADIO
 - SK_ARBITRO

- SK_TECNICO_JUGADOR
 - SK_TECNICO_CONTRINCANTE
 - MINUTO_ASISTENCIA
 - POSICION_ASISTENCIA
- FT_TARJETA
 - SK_JUGADOR
 - SK_ARBITRO
 - SK_TECNICO_JUGADOR
 - SK_EQUIPO
 - SK_ESTADO
 - MINUTO
 - POSICION_JUGADOR
- FT_PENAL
 - SK_JUGADOR
 - SK_JUGADOR_GOLERO
 - SK_EQUIPO_JUGADOR
 - SK_EQUIPO_CONTRINCANTE
 - SK_ESTADIO
 - SK_TECNICO_JUGADOR
 - SK_TECNICO_CONTRINCANTE
 - SK_ARBITRO
 - MINUTO_PENAL
 - POSICION_PENAL