

## İlk aşama Veri Temizleme Ve Manipulasyon ;

Gerekli tablolar ve kolonlar ;

order\_details; order\_id , product\_id , unit\_price , quantity , discount

products; product\_id , product\_name

orders; order\_id , order\_date , employee\_id

employees; employee\_id , first\_name , Last\_name

## Analiz öncesi veri temizleme ve manipulasyonu aşamaları ;

### 1)Boş Değerlerin Kontrolü:NULL DEĞER YOK.

--order\_details tablosunda boş değerlerin kontrolü;

```
SELECT * FROM order_details
```

```
WHERE unit_price IS NULL OR quantity IS NULL OR discount IS NULL;
```

-- products tablosunda boş değerlerin kontrolü

```
SELECT * FROM products
```

```
WHERE product_name IS NULL;
```

-- orders tablosunda boş değerlerin kontrolü

```
SELECT * FROM orders
```

```
WHERE order_date IS NULL;
```

### 2)Veri Tiplerinin Kontrolü:VERİ TİPLERİ KULLNIMA UYGUN

Veri tiplerini kontrol etmek için veri tabanının şemasını kullanabiliriz

```
SELECT column_name, data_type
```

```
FROM information_schema.columns
```

```
WHERE table_name = 'order_details';
```

```
SELECT column_name, data_type
```

```
FROM information_schema.columns
```

```
WHERE table_name = 'products';
```

```
SELECT column_name, data_type
```

```
FROM information_schema.columns
```

```
WHERE table_name = 'orders';
```

## Senaryo 1: Satış Müdürü - En Çok Satılan Ürünler ve Müşteri Profili

**Senaryo:** Satış Müdürü, hangi ürünlerin en çok satıldığını ve bu ürünleri en çok hangi müşteri segmentlerinin satın aldığını bilmek istiyor. Bu bilgiler, gelecekteki pazarlama stratejilerini belirlemek için kullanılacak.

### Sorular:

- En çok satılan 10 ürün hangileridir?

Bu ürünleri en çok alan müşterilerin şirkete sağladığı değere göre (harcama mükhtarlarına göre ) hangi müşteri segmentleri satın alıyor?

-- En çok satılan 10 ürünü belirleyin ve bu ürünleri alan müşterilerin bilgilerini çekin

WITH top\_products AS (

SELECT

p.product\_id,

p.product\_name,

SUM(od.quantity) AS total\_quantity

FROM

order\_details od

JOIN products p ON od.product\_id = p.product\_id

GROUP BY

p.product\_id, p.product\_name

ORDER BY

total\_quantity DESC

LIMIT 10

),

customer\_purchases AS (

SELECT

c.customer\_id,

c.company\_name,

p.product\_id,

p.product\_name,

SUM(od.unit\_price \* od.quantity) AS total\_amount

FROM

customers c

```

JOIN orders o ON c.customer_id = o.customer_id

JOIN order_details od ON o.order_id = od.order_id

JOIN top_products p ON od.product_id = p.product_id

GROUP BY

    c.customer_id, c.company_name, p.product_id, p.product_name

),

total_spent AS (

    SELECT

        customer_id,

        company_name,

        SUM(total_amount) AS total_spent

    FROM

        customer_purchases

    GROUP BY

        customer_id, company_name

)

SELECT

    cp.customer_id,

    cp.company_name,

    cp.product_id,

    cp.product_name,

    cp.total_amount,

    ts.total_spent,

    CASE

        WHEN ts.total_spent >= (SELECT PERCENTILE_CONT(0.75) WITHIN GROUP (ORDER BY total_spent) FROM total_spent)
        THEN 'High Value'

        WHEN ts.total_spent >= (SELECT PERCENTILE_CONT(0.50) WITHIN GROUP (ORDER BY total_spent) FROM total_spent)
        THEN 'Mid Value'

        ELSE 'Low Value'

    END AS customer_segment

FROM

```

```
customer_purchases cp

JOIN total_spent ts ON cp.customer_id = ts.customer_id

ORDER BY

cp.product_id, cp.total_amount DESC;
```

## Senaryo 2: Depo Müdürü - Stok Seviyesi ve Yeniden Sipariş Noktası

**Senaryo:** Depo Müdürü, hangi ürünlerin stok seviyelerinin kritik düzeyde olduğunu ve yeniden sipariş verilmesi gerektiğini bilmek istiyor. Amaç, stok dışı kalma durumlarını önlemek ve tedarik zincirini optimize etmek.

-- Kritik stok seviyesindeki ürünleri ve yeniden sipariş miktarlarını belirlemek için SQL sorgusu

```
WITH critical_stock AS (

SELECT

    product_id,

    product_name,

    unit_in_stock,

    reorder_level,

    CASE

        WHEN unit_in_stock < reorder_level THEN reorder_level - unit_in_stock

        ELSE 0

    END AS reorder_quantity

FROM

    products

WHERE

    unit_in_stock <= reorder_level

    AND discontinued = 0

)

SELECT

    product_id,

    product_name,

    unit_in_stock,

    reorder_level,

    reorder_quantity

FROM
```

critical\_stock

ORDER BY

reorder\_quantity DESC;

### Senaryo 3: Finans Müdürü

**Senaryo:** Finans Müdürü Hangi tedarikçilerden yapılan alımlar en yüksek geliri gerektiğini bilmek istiyor. Amaç Daha karlı olan tedarikçilerden alım yaparak maliyetleri azaltmak. Ve Tedarikçi seçiminde maliyet ve getiri dengesini optimize etmek.

SELECT

s.company\_name,

SUM(od.unit\_price \* od.quantity \* (1 - od.discount)) AS total\_revenue

FROM

suppliers s

JOIN products p ON s.supplier\_id = p.supplier\_id

JOIN order\_details od ON p.product\_id = od.product\_id

JOIN orders o ON od.order\_id = o.order\_id

GROUP BY

s.company\_name

ORDER BY

total\_revenue DESC;

### Senaryo 4: İnsan Kaynakları Müdürü - Çalışan Performansı

**Senaryo:** İnsan Kaynakları Müdürü, satış temsilcilerinin motivasyonunu artırıcı ve genel şirket verimliliğini sağlamak amacıyla yeni bir ödül sistemi getirmek ister.( Toplam Satış Gelirleri, Sipariş Sayıları, Ortalama Sipariş Değerleri ana metrikler olarak belirlenmiştir.)

SELECT

e.employee\_id,

e.first\_name || ' ' || e.last\_name AS employee\_name,

SUM(od.unit\_price \* od.quantity \* (1 - od.discount)) AS total\_sales,

COUNT(o.order\_id) AS total\_orders,

AVG(od.unit\_price \* od.quantity \* (1 - od.discount)) AS avg\_order\_value

FROM

employees e

JOIN orders o ON e.employee\_id = o.employee\_id

```
JOIN order_details od ON o.order_id = od.order_id

WHERE

    e.title = 'Sales Representative'

GROUP BY

    e.employee_id, e.first_name, e.last_name

ORDER BY

    total_sales DESC;
```

son sipariş tarihlerini ve sipariş sayılarını hesaplayalım.

```
SELECT

    c.customer_id,

    c.company_name,

    COUNT(o.order_id) AS total_orders,

    MAX(o.order_date) AS last_order_date

FROM

    customers c

    LEFT JOIN orders o ON c.customer_id = o.customer_id

GROUP BY

    c.customer_id, c.company_name

ORDER BY

    last_order_date;
```

### Senaryo 5: Üst Yönetim Kurulu ;

**Senaryo:** genel iş stratejilerini belirlemek, müşteri ilişkilerini güçlendirmek ve gelir artışını sağlamak amacıyla 2 farklı yöntemle müşteri segmentasyonu yapmak istemişlerdir.(Rfm Analizi ve Clv Bazlı segmentasyonlar yapılabilmektedir.Rfm analizi python'da görselleştirecek Clv ise Power BI'de detaylı incelenecektir.)

CLV SEGMENTASYON SORGULARI SQL ;

#### 1. En Son Sipariş Tarihi Baz Alınarak Recency Hesaplama

Öncelikle, her müşteri için en son sipariş tarihine göre Recency (gün sayısı) hesaplanır:

```
SELECT

    customer_id,
```

```
    CURRENT_DATE - MAX(order_date) AS recency
FROM
    orders
GROUP BY
    customer_id;
```

## 2. Ortalama Sipariş Değeri (Average Order Value) Hesaplama

```
WITH avg_order_value AS (

    SELECT

        customer_id,

        AVG(od.unit_price * od.quantity * (1 - od.discount)) AS avg_order_value

    FROM

        orders o

    JOIN

        order_details od ON o.order_id = od.order_id

    GROUP BY

        customer_id

)
```

## 3. Alışveriş Sıklığı (Order Frequency) Hesaplama

```
WITH order_frequency AS (

    SELECT

        customer_id,

        COUNT(order_id) AS order_count

    FROM

        orders

    GROUP BY

        customer_id

)
```

#### 4. Müşteri Ömrü (Customer Lifespan) Hesaplama

WITH customer\_lifespan AS (

SELECT

customer\_id,

(MAX(order\_date) - MIN(order\_date)) / 365.0 AS customer\_lifespan\_years

FROM

orders

GROUP BY

customer\_id

)

#### CLV (Customer Lifetime Value) Hesaplama

WITH clv\_calculation AS (

SELECT

a.customer\_id,

a.avg\_order\_value \* f.order\_count \* COALESCE(l.customer\_lifespan\_years, 1) AS clv -- Müşteri ömrü en az 1 yıl olarak kabul edilir.

FROM

avg\_order\_value a

JOIN

order\_frequency f ON a.customer\_id = f.customer\_id

LEFT JOIN

customer\_lifespan l ON a.customer\_id = l.customer\_id

)

WITH clv\_segments AS (

SELECT

customer\_id,

CASE



```
        WHEN clv >= 10000 THEN 'Yüksek Değerli Müşteriler'

        WHEN clv BETWEEN 2000 AND 9999 THEN 'Orta Değerli Müşteriler'

        ELSE 'Düşük Değerli Müşteriler'

    END AS value_segment

FROM

    clv_calculation

)

CLV'ye Göre Müşteri Segmentasyonu;

WITH clv_segments AS (

    SELECT

        customer_id,

        CASE

            WHEN clv >= 10000 THEN 'Yüksek Değerli Müşteriler'

            WHEN clv BETWEEN 2000 AND 9999 THEN 'Orta Değerli Müşteriler'

            ELSE 'Düşük Değerli Müşteriler'

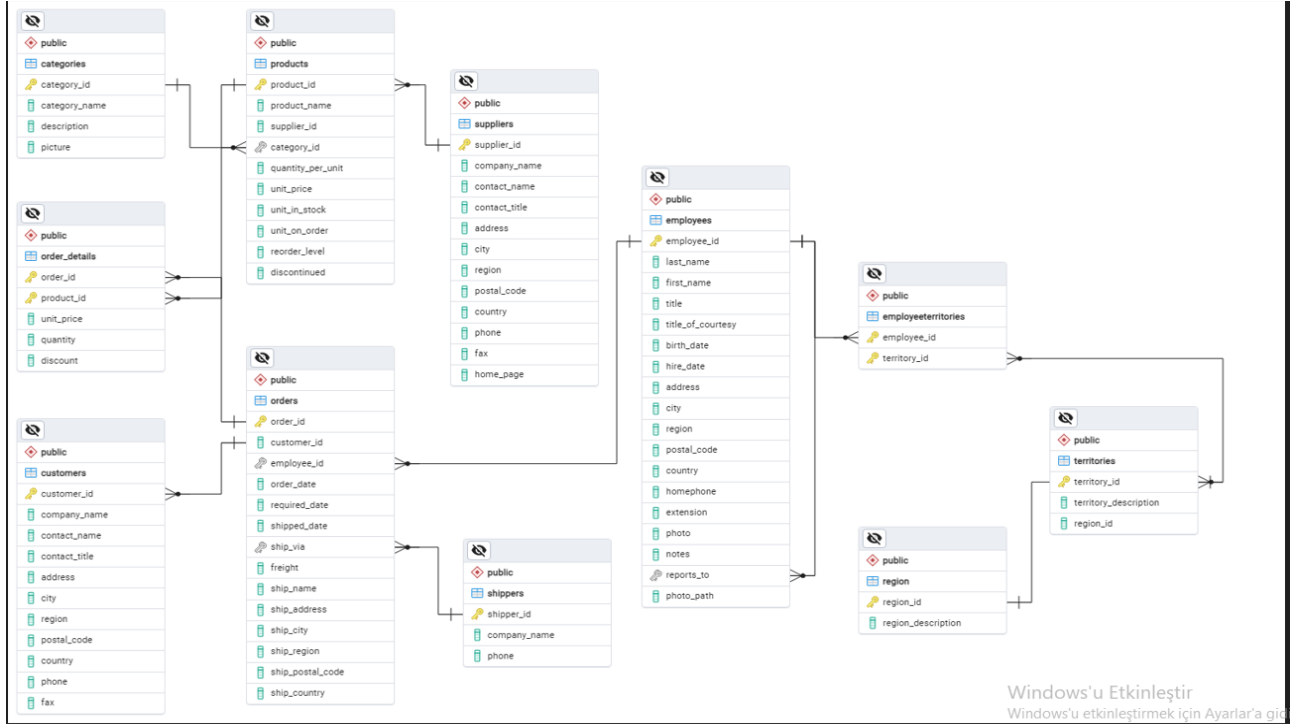
        END AS value_segment

    FROM

        clv_calculation

)
```

## Datasetiyle İlgili Bilgiler ;



### 1. Customers (Müşteriler)

- **customer\_id**: Müşteri kimlik numarası (Primary Key)
- **company\_name**: Şirket adı
- **contact\_name**: İrtibat kişisi adı
- **contact\_title**: İrtibat kişisi unvanı
- **address**: Adres
- **city**: Şehir
- **region**: Bölge
- **postal\_code**: Posta kodu
- **country**: Ülke
- **phone**: Telefon
- **fax**: Faks

### 2. Employees (Çalışanlar)

- **employee\_id**: Çalışan kimlik numarası (Primary Key)
- **last\_name**: Soyadı
- **first\_name**: Adı
- **title**: Unvan
- **title\_of\_courtesy**: Hitap şekli
- **birth\_date**: Doğum tarihi
- **hire\_date**: İşe başlama tarihi
- **address**: Adres
- **city**: Şehir
- **region**: Bölge
- **postal\_code**: Posta kodu
- **country**: Ülke
- **home\_phone**: Ev telefonu
- **extension**: Dahili numara

- **photo:** Fotoğraf
- **notes:** Notlar
- **reports\_to:** Rapor edilen kişi (Çalışan kimlik numarası)
- **photo\_path:** Fotoğraf yolu

### 3. Orders (Siparişler)

- **order\_id:** Sipariş kimlik numarası (Primary Key)
- **customer\_id:** Müşteri kimlik numarası (Foreign Key - Customers.customer\_id)
- **employee\_id:** Çalışan kimlik numarası (Foreign Key - Employees.employee\_id)
- **order\_date:** Sipariş tarihi
- **required\_date:** Gerekli tarih
- **shipped\_date:** Sevk tarihi
- **ship\_via:** Nakliyecı kimlik numarası (Foreign Key - Shippers.shipper\_id)
- **freight:** Nakliye ücreti
- **ship\_name:** Sevk adı
- **ship\_address:** Sevk adresi
- **ship\_city:** Sevk şehri
- **ship\_region:** Sevk bölgesi
- **ship\_postal\_code:** Sevk posta kodu
- **ship\_country:** Sevk ülkesi

### 4. Order Details (Sipariş Detayları)

- **order\_id:** Sipariş kimlik numarası (Primary Key, Foreign Key - Orders.order\_id)
- **product\_id:** Ürün kimlik numarası (Primary Key, Foreign Key - Products.product\_id)
- **unit\_price:** Birim fiyat
- **quantity:** Miktar
- **discount:** İndirim

### 5. Products (Ürünler)

- **product\_id:** Ürün kimlik numarası (Primary Key)
- **product\_name:** Ürün adı
- **supplier\_id:** Tedarikçi kimlik numarası (Foreign Key - Suppliers.supplier\_id)
- **category\_id:** Kategori kimlik numarası (Foreign Key - Categories.category\_id)
- **quantity\_per\_unit:** Birim başına miktar
- **unit\_price:** Birim fiyat
- **units\_in\_stock:** Stoktaki birim sayısı
- **units\_on\_order:** Sipariş edilen birim sayısı
- **reorder\_level:** Yeniden sipariş seviyesi
- **discontinued:** Devam etmiyor (0 = hayır, 1 = evet)

### 6. Suppliers (Tedarikçiler)

- **supplier\_id:** Tedarikçi kimlik numarası (Primary Key)
- **company\_name:** Şirket adı
- **contact\_name:** İrtibat kişisi adı
- **contact\_title:** İrtibat kişisi unvanı
- **address:** Adres
- **city:** Şehir
- **region:** Bölge
- **postal\_code:** Posta kodu
- **country:** Ülke
- **phone:** Telefon
- **fax:** Faks

- **home\_page:** Web sayfası

## 7. Categories (Kategoriler)

- **category\_id:** Kategori kimlik numarası (Primary Key)
- **category\_name:** Kategori adı
- **description:** Açıklama
- **picture:** Resim

## 8. Shippers (Nakliyeciler)

- **shipper\_id:** Nakliyecisi kimlik numarası (Primary Key)
- **company\_name:** Şirket adı
- **phone:** Telefon

## 9. Region (Bölgeler)

- **region\_id:** Bölge kimlik numarası (Primary Key)
- **region\_description:** Bölge açıklaması

## 10. Territories (Bölge Alanları)

- **territory\_id:** Bölge alanı kimlik numarası (Primary Key)
- **territory\_description:** Bölge alanı açıklaması
- **region\_id:** Bölge kimlik numarası (Foreign Key - Region.region\_id)

## 11. EmployeeTerritories (Çalışan Bölge Alanları)

- **employee\_id:** Çalışan kimlik numarası (Primary Key, Foreign Key - Employees.employee\_id)
- **territory\_id:** Bölge alanı kimlik numarası (Primary Key, Foreign Key - Territories.territory\_id)

## 12. USStates (ABD Eyaletleri)

- **state\_id:** Eyalet kimlik numarası (Primary Key)
- **state\_name:** Eyalet adı
- **state\_abbr:** Eyalet kısaltması
- **state\_region:** Eyalet bölgesi

## İlişkiler

- **Customers ve Orders:** Customers.customer\_id = Orders.customer\_id
- **Employees ve Orders:** Employees.employee\_id = Orders.employee\_id
- **Orders ve Order Details:** Orders.order\_id = Order Details.order\_id
- **Products ve Order Details:** Products.product\_id = Order Details.product\_id
- **Suppliers ve Products:** Suppliers.supplier\_id = Products.supplier\_id
- **Categories ve Products:** Categories.category\_id = Products.category\_id
- **Shippers ve Orders:** Shippers.shipper\_id = Orders.ship\_via
- **Region ve Territories:** Region.region\_id = Territories.region\_id
- **Employees ve EmployeeTerritories:** Employees.employee\_id = EmployeeTerritories.employee\_id
- **Territories ve EmployeeTerritories:** Territories.territory\_id = EmployeeTerritories.territory\_id