ERD for Database

metin, doküman, belge, sayı, numara, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

Sipariş Analizi

1. Aylık olarak order dağılımını inceleyiniz. Tarih verisi için order\_approved\_at kullanılmalıdır.

SELECT date\_trunc('month', order\_approved\_at)::date

AS order\_approved\_month, COUNT(order\_id) AS order\_count

FROM orders

WHERE order\_approved\_at IS NOT NULL

GROUP BY order\_approved\_month

ORDER BY order\_approved\_month ASC

2. Aylık olarak order status kırılımında order sayılarını inceleyiniz. Sorgu sonucunda çıkan outputu excel ile görselleştiriniz. Dramatik bir düşüşün ya da yükselişin olduğu aylar var mı? Veriyi inceleyerek yorumlayınız.

SELECT date\_trunc('month',order\_approved\_at)::date AS order\_approved\_month,

order\_status, COUNT(order\_id) AS order\_count

FROM orders

WHERE order\_approved\_at IS NOT NULL

GROUP BY 1,2

ORDER BY 1,2



--Aylık bazda order statüs kırılımı grafiği incelendiğinde en çok siparişin 2017 Kasım ayında verildiği gözlemlenmektedir. Sipariş sayılarının 2017 Kasım ayında en yüksek seviyede gözlemlenmesi, her yıl dünya çapında birçok marka tarafından da uygulanan black friday indirimleri ile açıklanabilir. Order sayılarının en düşük olduğu dönemler incelendiğinde ise göze çarpan üç dönem vardır. Bu dönemler 2016 yılının Eylül ve Aralık ayları ile 2018 yılının Eylül ayı olarak öne çıkmaktadır. Bu düşüşlerin sebepleri araştırılırken söz konusu alışveriş platformunun faaliyet gösterdiği ülkenin ekonomik ve politik durumu göz önünde bulundurulmalıdır ki 2018 Eylül ayındaki sipariş sayılarında gözlemlenen bu ciddi düşüş aynı dönemde Brezilya’da artış gösteren işsizlik sayılarıyla açılanabilir.

--Order statüs kırılımında bir analiz yapıldığında ise öncelikle teslimatı gerçekleşmiş siparişler incelendiğinde en çok teslimatın gerçekleştiği dönemin 2017 Kasım ayı olduğu dikkat çekmektedir.

Ancak grafiğe genel anlamda bakıldığında en çok göze çarpan durum geçmiş dönemlere ait birçok siparişin teslimatının gerçekleşmemiş olmasıdır. Örneğin 2017 yılının Şubat ve Nisan aylarında onaylanmış ancak bunun dışında teslimata dair bir sonraki aşamaya geçilmemiş 2 adet sipariş olduğu dikkat çekmektedir. Bu durum bu e ticaret firmasına ilişkin teslimat sürecinin aşamaları olan paketleme, taşıma, kargo gibi süreçlerle alakalı birtakım aksaklıkların yaşadığının bir göstergesi olabilir.

3.Ürün kategorisi kırılımında sipariş sayılarını inceleyiniz. Özel günlerde öne çıkan kategoriler nelerdir?

--Yılbaşı döneminde öne çıkan kategoriler incelenmek istendiğinde:

SELECT o.order\_approved\_at::date, p.product\_category\_name,eng\_product\_name.product\_category\_name\_english,count(o.order\_id) AS order\_count

FROM products AS p

LEFT JOIN product\_category\_name\_translation AS eng\_product\_name

ON p.product\_category\_name = eng\_product\_name.product\_category\_name

LEFT JOIN order\_items AS ot

ON p.product\_id =ot.product\_id

LEFT JOIN orders AS o

ON ot.order\_id = o.order\_id

WHERE

o.order\_approved\_at IS NOT NULL

AND p.product\_category\_name IS NOT NULL

AND EXTRACT(month from o.order\_approved\_at ) = 12

AND EXTRACT(day from o.order\_approved\_at ) BETWEEN 25 AND 31

GROUP BY 1,2,3

ORDER BY order\_count DESC

LIMIT 10

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

--Yılbaşı döneminde öne çıkan kategorileri incelemek için Aralık ayının son 6 günlük verileri dikkate alındığında bed\_bath\_table kategorisi öne çıkmaktadır bu kategoriyi ise ikinci olarak furniture\_decor kategorisi izlemektedir. Furniture\_decor kategorisini ise health\_beauty garden\_tools, telephony ve sports\_leisure kategorileri izlemektedir.

--Sevgililer gününde öne çıkan kategoriler incelenmek istendiğinde:

SELECT o.order\_approved\_at::date, p.product\_category\_name,eng\_product\_name.product\_category\_name\_english,count(o.order\_id) AS order\_count

FROM products AS p

LEFT JOIN product\_category\_name\_translation AS eng\_product\_name

ON p.product\_category\_name = eng\_product\_name.product\_category\_name

LEFT JOIN order\_items AS ot

ON p.product\_id =ot.product\_id

LEFT JOIN orders AS o

ON ot.order\_id = o.order\_id

WHERE

o.order\_approved\_at IS NOT NULL

AND p.product\_category\_name IS NOT NULL

AND EXTRACT(month from o.order\_approved\_at ) = 2

AND EXTRACT(day from o.order\_approved\_at ) BETWEEN 7 AND 14

GROUP BY 1,2,3

ORDER BY order\_count DESC

LIMIT 10

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

--Sevgililer günü için zirve yapan kategorileri incelemek için ise 1 haftalık verilere bakıldığında yine öne çıkan kategorinin bed\_bath\_table kategorisi olduğu dikkat çekmekte. Bu kategoriyi ise computers\_accessories, sports\_leisure, health\_beauty kategorileri izlemektedir.

--Black friday döneminde zirve yapan kategoriler:

SELECT o.order\_approved\_at::date, p.product\_category\_name,eng\_product\_name.product\_category\_name\_english,count(o.order\_id) AS order\_count

FROM products AS p

LEFT JOIN product\_category\_name\_translation AS eng\_product\_name

ON p.product\_category\_name = eng\_product\_name.product\_category\_name

LEFT JOIN order\_items AS ot

ON p.product\_id =ot.product\_id

LEFT JOIN orders AS o

ON ot.order\_id = o.order\_id

WHERE

o.order\_approved\_at IS NOT NULL

AND p.product\_category\_name IS NOT NULL

AND EXTRACT(month from o.order\_approved\_at ) = 11

AND EXTRACT(day from o.order\_approved\_at ) BETWEEN 25 AND 27

GROUP BY 1,2,3

ORDER BY order\_count DESC

LIMIT 10

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

--Black Friday için Kasım ayının son Cuma verileri incelendiğinde ise yine zirve yapan kategori bed\_bath\_table kategorisi olmuştur ve bunu furniture\_decor kategorisi izlemiştir.

--Brezilya’da Dia do Consumidor (Tüketicinin Günü)olarak bilinen bu günde her yıl 15 Mart tarihinde tüketicilere özel indirimler uygulanan bir gündür.

metin, ekran görüntüsü, sayı, numara, yazı tipi içeren bir resim

Açıklama otomatik olarak oluşturuldu

Dia do Consumidor verileri incelendiğinde ise bed\_bath\_table kategorisi yine öne çıkan kategori olmuştur.

4. Haftanın günleri ve ay günleri bazında order sayılarını inceleyiniz. Yazdığınız sorgunun outputu ile excel’de bir görsel oluşturup yorumlayınız.

--Haftanın Günleri

SELECT

CASE

WHEN EXTRACT(DOW FROM order\_approved\_at) = 0 THEN 'Pazar'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 1 THEN 'Pazartesi'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 2 THEN 'Salı'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 3 THEN 'Çarşamba'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 4 THEN 'Perşembe'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 5 THEN 'Cuma'

WHEN EXTRACT(DOW FROM order\_approved\_at) = 6 THEN 'Cumartesi'

END AS day\_of\_week,

COUNT(order\_id) AS order\_count

FROM orders

WHERE order\_approved\_at IS NOT NULL

GROUP BY 1

ORDER BY 1 DESC

metin, ekran görüntüsü, yazı tipi, sayı, numara içeren bir resim

Açıklama otomatik olarak oluşturuldu

--Ayın Günleri

SELECT

EXTRACT(DAY FROM order\_approved\_at) AS day\_of\_month,

COUNT(order\_id) AS order\_count

FROM orders

WHERE order\_approved\_at IS NOT NULL

GROUP BY 1

ORDER BY 1

Müşteri Analizi

1. Hangi şehirlerdeki müşteriler daha çok alışveriş yapıyor? Müşterinin şehrini en çok sipariş verdiği şehir olarak belirleyip analizi ona göre yapınız.

--en çok sipariş verilen şehir sıralaması

WITH CustomerOrderAmounts AS (

SELECT

c.customer\_city,

COUNT(o.order\_id) AS order\_amount

FROM

customers AS c

LEFT JOIN

orders AS o ON c.customer\_id = o.customer\_id

GROUP BY

c.customer\_city)

SELECT

customer\_city,

SUM(order\_amount) AS total\_order\_amount

FROM CustomerOrderAmounts

GROUP BY customer\_city

ORDER By total\_order\_amount DESC

--müşterilerin Şehri

WITH CustomerOrderAmounts AS (

SELECT

c.customer\_unique\_id,

c.customer\_city,

COUNT(o.order\_id) AS order\_amount

FROM customers c

LEFT JOIN

orders AS o ON c.customer\_id = o.customer\_id

GROUP BY c.customer\_unique\_id, c.customer\_city

),

RankedCities AS (

SELECT

coa.customer\_unique\_id,

coa.customer\_city,

RANK() OVER (PARTITION BY coa.customer\_unique\_id ORDER BY coa.order\_amount DESC) AS city\_rank

FROM CustomerOrderAmounts AS coa

)

SELECT

rc.customer\_unique\_id,

rc.customer\_city,

CASE WHEN rc.city\_rank = 1 THEN 'En Çok Sipariş Verilen Şehir'

WHEN rc.city\_rank IS NULL THEN 'Sipariş Verilmemiş'

ELSE 'Diğer'

END AS city\_status

FROM

RankedCities AS rc

Satıcı Analizi

1. Siparişleri en hızlı şekilde müşterilere ulaştıran satıcılar kimlerdir? Top 5 getiriniz.

WITH SellerOrderDelivery AS (

SELECT

oi.seller\_id,

COUNT(DISTINCT o.order\_id) AS order\_count,

ROUND(AVG(EXTRACT(EPOCH FROM o.order\_delivered\_customer\_date - o.order\_purchase\_date)::numeric) / 3600, 2) AS avg\_delivery\_hours

FROM order\_items oi

LEFT JOIN orders AS o

ON oi.order\_id = o.order\_id

WHERE o.order\_delivered\_customer\_date IS NOT NULL

GROUP BY oi.seller\_id

)

SELECT

s.seller\_id,

sod.order\_count,

sod.avg\_delivery\_hours::numeric AS rounded\_avg\_delivery\_hours

FROM SellerOrderDelivery sod

LEFT JOIN sellers AS s

ON sod.seller\_id = s.seller\_id

ORDER BY 3 ASC

limit 5

--Bu sorgu ile order\_purchase\_date ile order\_delivered\_customer\_date arasındaki sürenin ortalaması alınarak siparişi en hızlı şekilde müşterilere ulaştıran ilk 5 satıcı listelenmektedir.

--Bu satıcıların order sayıları ile ürünlerindeki yorumlar ve puanlamaları inceleyiniz ve yorumlayınız.

WITH seller\_rates AS (

SELECT

s.seller\_id,

COUNT(DISTINCT oi.order\_id) AS order\_count,

COUNT(DISTINCT reviews\_order.review\_comment\_message) AS order\_comment,

ROUND(AVG(reviews\_order.review\_score), 2) AS avg\_rating

FROM order\_reviews AS reviews\_order

LEFT JOIN order\_items AS oi ON oi.order\_id = reviews\_order.order\_id

LEFT JOIN sellers AS s ON s.seller\_id = oi.seller\_id

WHERE reviews\_order.review\_score IS NOT NULL

GROUP BY s.seller\_id

),

seller\_rank AS (

SELECT

seller\_id,

order\_count,

order\_comment,

avg\_rating,

RANK() OVER (ORDER BY avg\_rating DESC) AS avg\_rating\_rank,

RANK() OVER (ORDER BY order\_count DESC) AS order\_count\_rank

FROM seller\_rates

)

SELECT

seller\_id,

order\_count,

order\_comment,

avg\_rating,

avg\_rating\_rank,

order\_count\_rank

FROM seller\_rank

WHERE seller\_id IS NOT NULL

--satıcıları sipariş ve yorum sayıları ile değerlendirme puanları arasındaki ilişkiye göre değerlendirmek için Rank fonksiyonundan faydalanıldı. Sorgu sonucu elde edilen çıktılar sonucunda düşük Rank değerine sahip satıcıların daha yüksek bir performansa sahip olduğunu gösterir. Örneğin ilk sırada yer alan satıcının order\_count\_rank değeri 1 olarak belirlenmiştir bu satıcı sipariş sayısı bakımından yüksek bir performansa sahip olarak değerlendirilir ancak avg\_rating\_rank değeri 2065 olarak atanan bu satıcı rating bakımından iyi bir performans gösterememiştir.

2. Hangi satıcılar daha fazla kategoriye ait ürün satışı yapmaktadır? Fazla kategoriye sahip satıcıların order sayıları da fazla mı?

SELECT

s.seller\_id,

COUNT(DISTINCT p.product\_category\_name) AS category\_count,

COUNT(DISTINCT oi.order\_id) AS order\_count

FROM sellers AS s

INNER JOIN order\_items AS oi

ON s.seller\_id = oi.seller\_id

INNER JOIN products AS p

ON oi.product\_id = p.product\_id

WHERE p.product\_category\_name IS NOT NULL

GROUP BY s.seller\_id

ORDER BY category\_count DESC

--Bu sorgu sonucunda elde edilen sonuçlara göre satıcıların sahip olduğu kategori sayısı ve order sayıları karşılaştırıldığında bir satıcının daha az kategoride ürün satışı yapıyor olması order sayılarını doğrudan etkileyen bir faktör değildir. Daha az kategoride satışı olan satıcıların, daha fazla kategoride satışa sahip olan satıcılara göre daha fazla order sayısına sahip oldukları gözlemlenmektedir. Bu daha az kategoride ürün satışı yapan satıcıların belli başlı birkaç kategoriye yönelerek bu kategorilerde uzmanlaşmalarıyla açıklanabilir.

Payment Analizi

1. Ödeme yaparken taksit sayısı fazla olan kullanıcılar en çok hangi bölgede yaşamaktadır? Bu çıktıyı yorumlayınız.

SELECT

c.customer\_city,

SUM(CASE WHEN p.payment\_installments IS NOT NULL THEN 1 ELSE 0 END) AS total\_installments

FROM customers AS c

LEFT JOIN orders AS o

ON c.customer\_id = o.customer\_id

LEFT JOIN order\_payments AS p ON o.order\_id = p.order\_id

GROUP BY c.customer\_city

ORDER BY total\_installments DESC

--Bu sorgu ile customer\_city ve payment\_installments değerlerine göre gruplandırılarak ve installment\_count değerine göre sıralama yapılarak, hangi şehirde kaç taksit yapıldığını gösteren bir liste elde edilmiştir ve azalan bir sıralamaya göre listelenmiştir.

--Bu sorgudan elde edilen verilerden yola çıkarak customer\_city bazında taksit sayısının en fazla görüldüğü şehirler belirlenerek o şehirdeki müşterilerin ödeme alışkanlıklarıyla ilgili analizler yapılabilir.

Daha etkili bir pazarlama stratejisi oluşturmak için taksit sayısı ve customer\_city bazında yapılan ödemeler analiz edilerek, farklı müşteri segmentlerini belirlenebilir ve bu segmentlere yönelik farklı stratejiler uygulanabilir

2. Ödeme tipine göre başarılı order sayısı ve toplam başarılı ödeme tutarını hesaplayınız. En çok kullanılan ödeme tipinden en az olana göre sıralayınız.

WITH PaymentStatistics AS (

SELECT

p.payment\_type,

COUNT(DISTINCT o.order\_id) AS order\_count,

SUM(CAST(p.payment\_value AS NUMERIC)) AS total\_payment\_amount

FROM

order\_payments AS p

LEFT JOIN

orders AS o

ON p.order\_id = o.order\_id

GROUP BY

p.payment\_type

)

SELECT

payment\_type,

order\_count,

total\_payment\_amount

FROM

PaymentStatistics

ORDER BY

order\_count DESC, total\_payment\_amount DESC

--Bu sorgu ile her bir ödeme tipine ait sipariş sayısı ve toplam ödeme tutarı, sipariş sayısına göre büyükten küçüğe ve ardından toplam ödeme tutarına göre büyükten küçüğe sıralanarak listelenmiştir

Bu sorgudan çıkan sonuçlara göre en çok tercih edilen ödeme yöntemi credit card olarak belirlendiği gözlemlenmektedir.

3. Tek çekimde ve taksitle ödenen siparişlerin kategori bazlı analizini yapınız. En çok hangi kategorilerde taksitle ödeme kullanılmaktadır?

WITH PaymentDetails AS (

SELECT

o.order\_id,

p.payment\_type,

p.payment\_installments,

op.product\_id,

pr.product\_category\_name

FROM

orders AS o

LEFT JOIN

order\_payments AS p ON o.order\_id = p.order\_id

LEFT JOIN

order\_items AS op ON o.order\_id = op.order\_id

LEFT JOIN

products AS pr ON op.product\_id = pr.product\_id

)

SELECT

product\_category\_name,

payment\_type,

payment\_installments,

COUNT(DISTINCT order\_id) AS order\_count,

CASE

WHEN payment\_installments = 1 THEN 'Tek Çekim'

WHEN payment\_installments > 1 THEN 'Taksitli'

END AS payment\_category

FROM

PaymentDetails

WHERE

payment\_type IN ('credit\_card', 'voucher', 'debit\_card')

OR payment\_installments > 1

GROUP BY

product\_category\_name,

payment\_type,

payment\_installments

ORDER BY

payment\_installments DESC

--Bu sorgu ile sonuçlar ürün kategorisi, ödeme tipi ve taksit sayısına göre gruplanır ve taksit sayısına göre büyükten küçüğe sıralanır. Sorgu sonucunda elde edilen veriler doğrultusunda en çok taksitlendirmenin yapıldığı ürün kategorisinin otomotiv kategorisi olduğu gözlemlenmektedir.

RFM Analizi

--Recency Hesaplama

SELECT

customer\_id,

MAX(purchase\_date):: date AS last\_purchase\_date,

'2011-12-09'::date - MAX(purchase\_date)::date AS recency

INTO recency

FROM commerce\_data

WHERE customer\_id IS NOT NULL

GROUP BY customer\_id

-- Frequency Hesaplama

SELECT

customer\_id,

COUNT(DISTINCT invoice\_no) AS frequency

INTO frequency

FROM commerce\_data

GROUP BY customer\_id

-- Monetary Hesaplama

SELECT

customer\_id,

SUM(quantity \* CAST(unitprice AS numeric)) AS monetary

INTO monetary

FROM commerce\_data

GROUP BY customer\_id

-- RFM Skorları Oluşturma

SELECT

r.customer\_id,

NTILE(5) OVER (ORDER BY r.recency DESC) AS r\_score,

NTILE(5) OVER (ORDER BY f.frequency DESC) AS f\_score,

NTILE(5) OVER (ORDER BY m.monetary DESC) AS m\_score

INTO rfm\_scores

FROM recency AS r

LEFT JOIN frequency AS f

ON r.customer\_id = f.customer\_id

LEFT JOIN monetary AS m

ON r.customer\_id = m.customer\_id

-- RFM Skorlarını Birleştirme ve Segmentasyon

SELECT

rfm.customer\_id,

rfm.r\_score,

rfm.f\_score,

rfm.m\_score,

(rfm.r\_score::text || rfm.f\_score::text || rfm.m\_score::text) AS rfm\_score

FROM temp\_rfm\_scores AS rfm

LEFT JOIN commerce\_data AS cd

ON rfm.customer\_id = cd.customer\_id

SELECT

customer\_id,

r\_score,

f\_score,

m\_score,

CASE

WHEN r\_score >= 4 AND f\_score >= 4 AND m\_score >= 4 THEN 'VIP Müşteri'

WHEN r\_score >= 3 AND f\_score >= 3 AND m\_score >= 3 THEN 'Sadık Müşteri'

WHEN r\_score >= 2 AND f\_score >= 2 AND m\_score >= 2 THEN 'Potansiyel Müşteri'

ELSE 'Normal Müşteri'

END AS segment

FROM rfm\_scores

ORDER BY segment DESC

--RFM analizi için öncelikle müşterilerin recenc), frequency ve monetary değerlerinin hesaplandığı bir tablo oluştururldu. Recency hesaplamak için son sipariş tarihi ile müşterilerin ilk sipariş tarihi arasındaki fark listelenerek bir lisansı tablosu oluşturuldu. Frequency için her bir müşteri için invoice no saydırılarak sipariş sayıları saptandı. Monetary için ise unit price toplamı alınarak her bir müşteri için toplam ödeme tutarı listelenmiştir. Daha sonra bu tablolardaki RFM çıktıları NTILE fonksiyonu ile 5 gruba bölündü ve bir skor tablosu oluşturuldu oluşturulan bu skorlar customer id ile listelendi. Son olarak ise müşteriler skorlarına göre gruplandırıldı. Skoru 4 ve üzeri olan müşteriler VIP müşteri, 3 olan müşteriler sadık müşteri, 2 olan müşteriler potansiyel müşteri, geriye kalanlar ise normal müşteri grubuna dahil edilecek şekilde sorgu yazıldı. RFM analizi sonucunda elde edilen bu sorguları ile firmalar her bir müşteri segmenti için ayrı pazarlama stratejileri uygulayabilir. VIP ve sadık müşteri kategorilerindeki müşteriler için onların bu kategoride kalmasını sağlayacak kampanyalar uygulayabilir iken potansiyel ve normal müşteri kategorisindeki müşteriler için daha üst segmentlere çıkabilecekleri kampanyalar uygulayabilirler.