##### **Question 1 :**

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

SELECT

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

COUNT(order\_id) AS total\_orders

FROM

orders

WHERE

order\_approved\_at IS NOT NULL

GROUP BY

month

ORDER BY

month;

Aylık olarak sipariş verilerine bakıldığında 2016 yılında muhtemelen yıl sonu enflasyonun artışı ve maaşlardaki vergisel oranların artması nedeniyle siparişler düşme eğilimindeymiş. 2017 Ocak ile muhtemelen kişilerin revize edilerek yükselen maaşları ile sipariş sayılarında bir artış gözlenmeye başlamış. Yaz ve son bahar dönemlerinde de sürekli artış ile devam eden siparişler muhtemel black Friday indirimleri sebebebiyle kasım ayına en yüksek seviyesine ulaşmış gözükmektedir.Sonraki dönemlerde tekrar düşüş ile belirli bir seyir yakalanmış, muhtemel kadınlar günü, sevgileler günü gibi sebeplerle normal seyirlerde devam etmiş gözükmektedir.

##### **Question 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 TO\_CHAR(order\_approved\_at, 'YYYY,MM') AS monthly\_order,

COUNT(order\_id) AS order\_count,

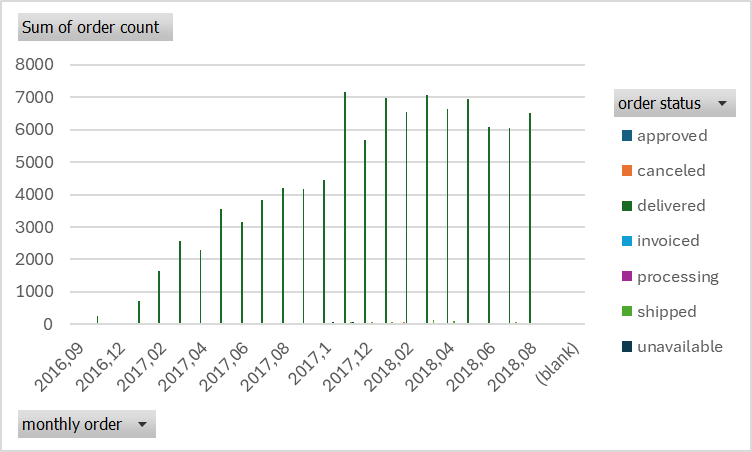
order\_status

FROM orders

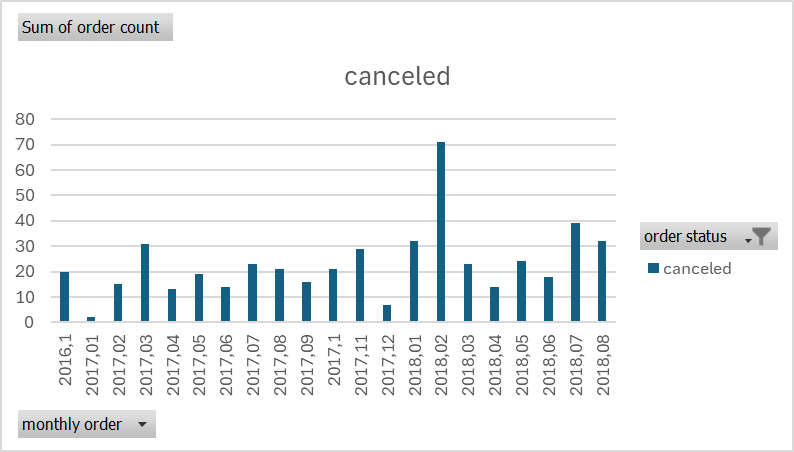
WHERE order\_approved\_at IS NOT NULL

GROUP BY TO\_CHAR(order\_approved\_at, 'YYYY,MM'), order\_status

ORDER BY order\_status, monthly\_order ASC;

****

Genel anlamda en yüksek ulaştrılan sipariş 2017 kasım ayında gözükmektedir. Bu artışın dünya genelinde black Friday indirimleri sebebiyle olduğunu düşünmekteyim. Sonrasında yılbaşı, sevgililer günü, kadınlar günü vb özel günler sebebiyle ulaştırılan sipariş sayıları genel olarak yüksek seviyelerde devam etmiş.



Şubat ayında cancel sayılarında birden artış olmuş, bu durum sevgililer günü için verilen fakat sonrasında vazgeçilen siparişlerden kaynaklı olabilir. Diğer aylarda kayda değer aşırı artış gözükmemektedir.

##### **Question 3 :**

**-Ürün kategorisi kırılımında sipariş sayılarını inceleyiniz. Özel günlerde öne çıkan kategoriler nelerdir? Örneğin yılbaşı, sevgililer günü…**

WITH order\_count AS (

SELECT

TO\_CHAR(order\_approved\_at, 'YYYY-MM') AS order\_date,

product\_category\_name,

COUNT(DISTINCT o.order\_id) AS orderCount

FROM

order\_items o

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

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

WHERE

order\_approved\_at IS NOT NULL

GROUP BY

1, 2

)

SELECT

order\_date,

orderCount,

category\_name\_english

FROM

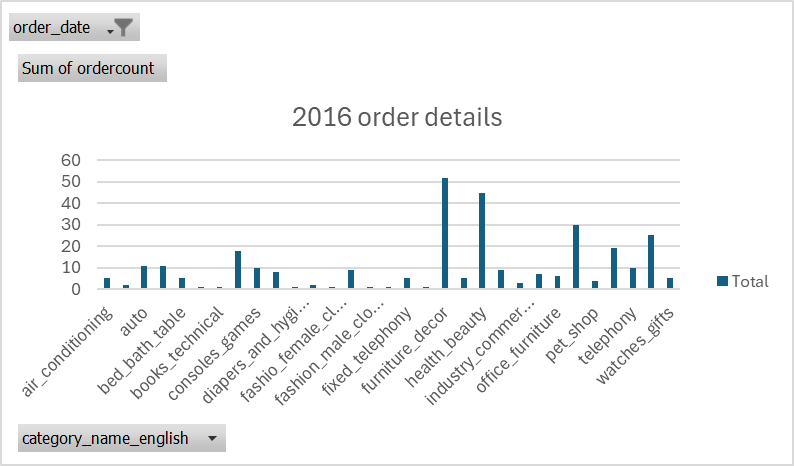
order\_count oc

INNER JOIN translation t ON t.category\_name = oc.product\_category\_name

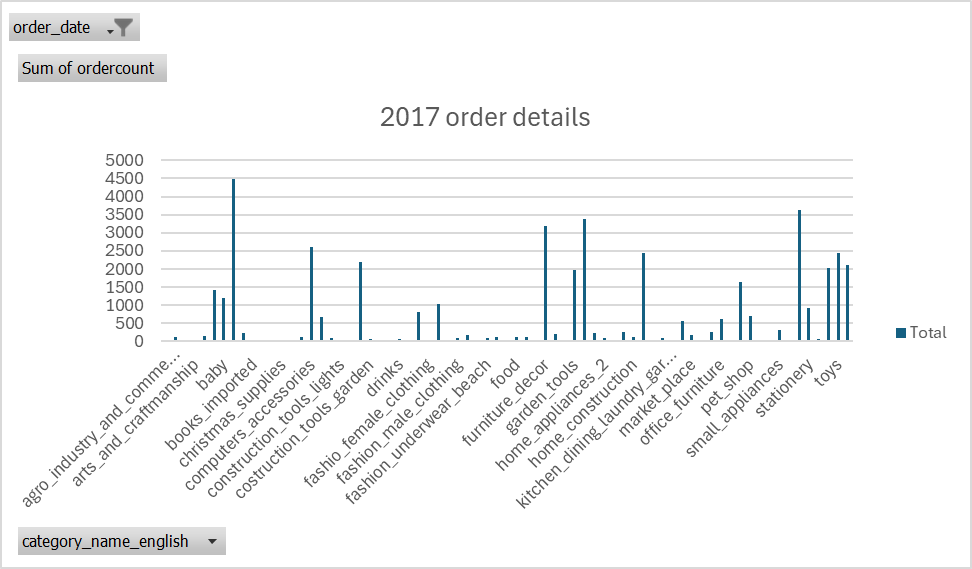
ORDER BY

order\_date,

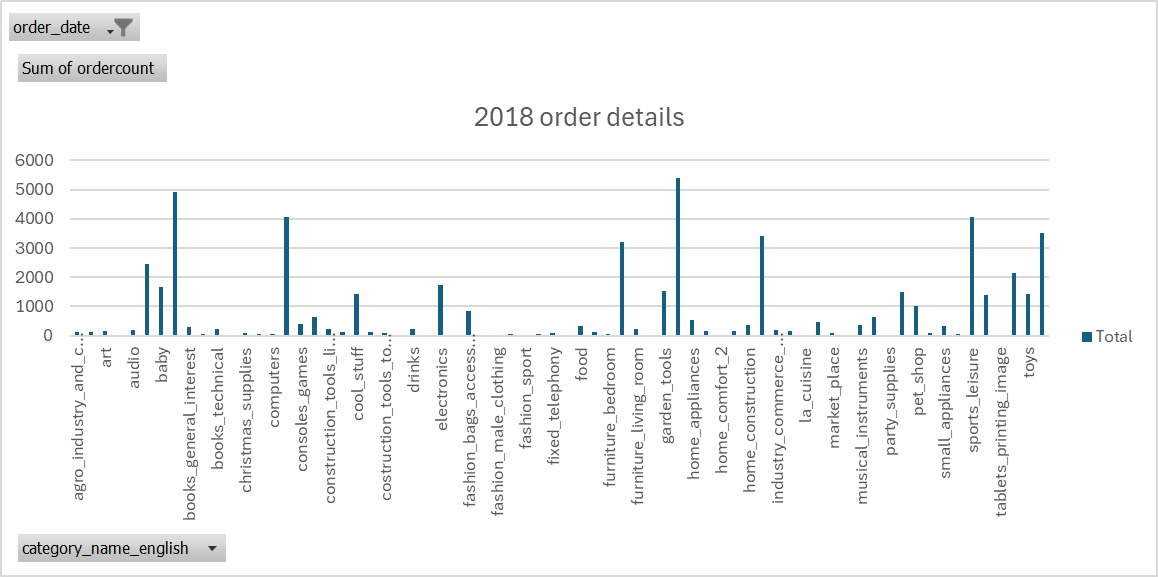
orderCount DESC;



2016 yılı itibayle en çok furnituredecor ve health beauty, perfumery ürünlerinde yüksek satışlar olmuş. Genel anlamda infulencer sektörlerindeki reklam kampanyaları sağlık güzellik ürünlerinde artışa sebep olmuş olabilir ayrıca mobilya alanında bir kampanya yapılmış olabilir.



2017 yılında en çok satışı özellikle temmuz ayı sonrası bed bathtable kategorisinde olmuş, bunu akabinde sports leisure health beauty ve furnıture decor ürünleri takip etmiş.



2018 yılında health beauty ürünleri yüksek satışları devam etmiş, genel anlamda bu sektörün instagram ve youtube gibi reklamlarla yükselişe geçtiğini düşünüyorum Tüketici tercihleri sürekli olarak değişmiş fakat health beautiy ürün satışları sürekli devam etmiş.

##### **Question 4 :**

**-Haftanın günleri(pazartesi, perşembe, ….) ve ay günleri (ayın 1’i,2’si gibi) bazında order sayılarını inceleyiniz. Yazdığınız sorgunun outputu ile excel’de bir görsel oluşturup yorumlayınız**

WITH day\_based AS (

SELECT

TO\_CHAR(order\_purchase\_timestamp, 'Day') as days,

COUNT(DISTINCT order\_id) as orderCount

FROM

orders

WHERE

order\_status != 'canceled' AND order\_status != 'unavailable'

GROUP BY 1

),

monthday\_based AS (

SELECT

EXTRACT(DAY FROM order\_purchase\_timestamp) as monthday,

COUNT(order\_id) as orderCount

FROM

orders

WHERE

order\_status != 'canceled' AND order\_status != 'unavailable'

GROUP BY 1

)

SELECT \* FROM day\_based WHERE days IS NOT NULL;

En yüksek sipariş Pazartesi günleri verilmiş. En düşük sayı ise Cumartesi gözükmekte. Sebebinin Cumartesi-Pazar günleri kişilerin gezmeyi ya da dinlenmeyi tercih etmeleri diğer günlerde alış veriş yaptıklarını düşüneiliriz.

#### **Case 2 : Müşteri Analizi**

##### **Question 1 :**

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

WITH city\_order\_counts AS (

SELECT

c.customer\_city,

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

FROM

customers c

JOIN

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

GROUP BY

c.customer\_city

)

SELECT

customer\_city,

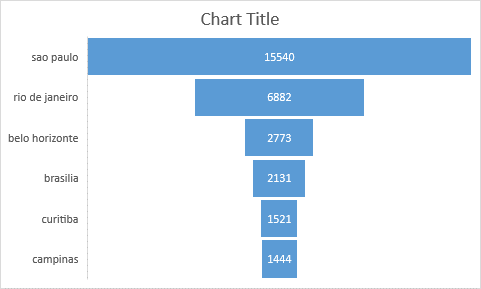
order\_count

FROM

city\_order\_counts

ORDER BY

order\_count DESC;



En yüksek siparişler büyük şehilerden nüfusun yüksek olduğu alanlardan alınmış.Muhtemelen kurumsal firmaların, plazaların ve büyük işyerlerinin olduğu kozmopolit şehirlerde ekonomik seviyesi yüksek olan kişilerin siparişleridir.

#### **Case 3: Satıcı Analizi**

##### **Question 1 :**

**-Siparişleri en hızlı şekilde müşterilere ulaştıran satıcılar kimlerdir? Top 5 getiriniz. Bu satıcıların order sayıları ile ürünlerindeki yorumlar ve puanlamaları inceleyiniz ve yorumlayınız.**

WITH DeliveryTimes AS (

SELECT

s.seller\_id,

AVG(EXTRACT(EPOCH FROM (ord.order\_delivered\_customer\_date - ord.order\_purchase\_timestamp)) / 86400) AS avg\_delivery\_time\_days,

COUNT(ord.order\_id) AS order\_count

FROM

sellers s

JOIN order\_items o ON s.seller\_id = o.seller\_id

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

WHERE

ord.order\_status = 'delivered'

GROUP BY

s.seller\_id

)

SELECT

dt.seller\_id,

dt.avg\_delivery\_time\_days,

dt.order\_count,

AVG(rev.review\_score) AS avg\_review\_score,

COUNT(rev.review\_id) AS review\_count

FROM

DeliveryTimes dt

JOIN order\_items oi ON dt.seller\_id = oi.seller\_id

JOIN reviews rev ON oi.product\_id = rev.product\_id

GROUP BY

dt.seller\_id, dt.avg\_delivery\_time\_days, dt.order\_count

ORDER BY

dt.avg\_delivery\_time\_days ASC

LIMIT 5;

A screenshot of a number

Description automatically generated

İlk sıradaki satıcı en hızlı teslimatı yapmış, yorumları diğerlerine göre düşük kalmış sebebi ürün kalitesi vs olabilir.Genel anlamda diğer satıcıların da teslimat süreleri ve yorumları iyi gözükmekte.

##### **Question 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 unique\_categories,

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

FROM

sellers s

JOIN order\_items oi ON s.seller\_id = oi.seller\_id

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

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

GROUP BY

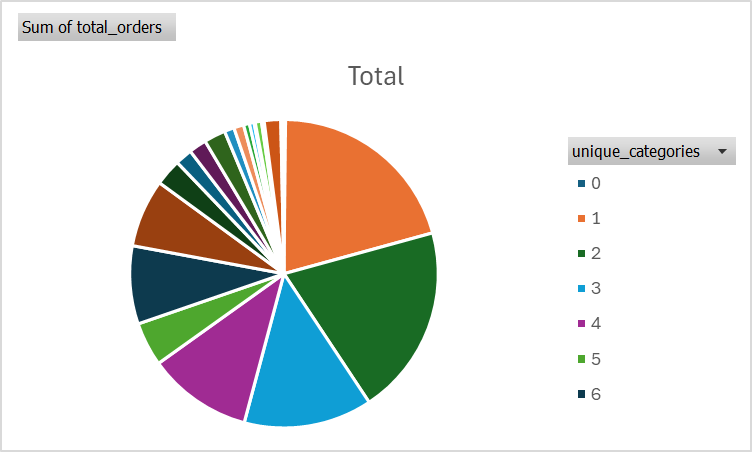
s.seller\_id

ORDER BY

unique\_categories DESC;

A screenshot of a computer

Description automatically generated



Sorgu sonucuna göre fazla kategorisi olan ama çok az satış yapan satıcılar mevcut.Yani kategorinin fazla olması satış stratejisinde etkili değildir. Fazla ve yüksek fiyattan mal satmaya çalışan pazarda rağbet gören ürünleri belirlemete çalışan üretici olmayan satıcılar olabilir. Öte yandan yalnızca bir ürün üzerinde uzmanlaşmış ve üretici olan, uygun fiyattan ürün satışı yapan satışcıların olduğunu da söyleyebiliriz.

#### **Case 4 : Payment Analizi**

##### **Question 1 :**

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

WITH tablo AS (

SELECT

customer\_city,

customer\_state,

COUNT(DISTINCT c.customer\_id) AS CUSTOMER\_COUNT

FROM

payments p

JOIN

orders AS ord ON p.order\_id = ord.order\_id

JOIN

customers AS c ON ord.customer\_id = c.customer\_id

WHERE

p.payment\_type = 'credit\_card' AND payment\_installments > 2

GROUP BY

customer\_city,

customer\_state

)

SELECT

customer\_city,

customer\_state,

CUSTOMER\_COUNT

FROM

tablo

WHERE

CUSTOMER\_COUNT > 1

ORDER BY

CUSTOMER\_COUNT DESC;

A screenshot of a computer

Description automatically generated

En çok sipariş verenler sao psulo, rio dejeneiro gibi şehirlerde gözükmektedir.Nufus sayısı olarakta yüksek olan bu şehirlerde muhtemelen çalışan insan sayısı da yüksek olduğundan insanlar daha çok alışveriş yapıyor olabilirler.Tüketimin artması da haliyle taksitli alışveriş yapılmasını beraberinde getirmiş.

##### **Question 2 :**

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

WITH tablo AS (

SELECT

COUNT(DISTINCT o.order\_status) AS ord\_status,

p.payment\_type,

o.order\_id,

COUNT(DISTINCT p.payment\_value) AS distinct\_payment\_values\_count

FROM

orders o

JOIN

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

WHERE

o.order\_status = 'delivered'

GROUP BY

p.payment\_type, o.order\_id

)

SELECT

COUNT(ord\_status) AS total\_delivered\_orders,

SUM(distinct\_payment\_values\_count) AS total\_distinct\_payment\_values,

payment\_type

FROM

tablo

GROUP BY

payment\_type

ORDER BY

total\_delivered\_orders DESC;

A screenshot of a computer

Description automatically generated

##### **Question 3 :**

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

SELECT

t.category\_name\_english,

COUNT(DISTINCT CASE WHEN p.payment\_installments = 1 THEN o.order\_id END) AS tek\_cekim\_siparis\_sayisi,

COUNT(DISTINCT CASE WHEN p.payment\_installments > 1 THEN o.order\_id END) AS taksitli\_siparis\_sayisi,

COALESCE(ROUND(SUM(CASE WHEN p.payment\_installments = 1 THEN p.payment\_value ELSE 0 END)::numeric, 2), 0) AS rounded\_tek\_cekim\_toplam\_odeme,

COALESCE(ROUND(SUM(CASE WHEN p.payment\_installments > 1 THEN p.payment\_value ELSE 0 END)::numeric, 2), 0) AS rounded\_taksitli\_toplam\_odeme

FROM

order\_items o

JOIN

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

JOIN

products pr ON o.product\_id = pr.product\_id

JOIN

translation t ON pr.product\_category\_name = t.category\_name

WHERE

p.payment\_installments IN (1, 2)

GROUP BY

t.category\_name\_english order by tek\_cekim\_siparis\_sayisi desc

A screenshot of a computer

Description automatically generated

Sorgu çıktımıza göre en çok tek çekim ile alış verişler sağlık-güzellik, spor, bilgisayar aksesuarları,yatak-banyo-masa, moniblya gibi alanlarda yapılmış.Genel anlamda tek çekim ödeme yapılan ürünler tüketicinin nispeten uygun fiyatlı ve tek seferde ödeyebileceğini düşündüğü ürünler olmuş.

#### **Case 5 : RFM Analizi**

**Aşağıdaki e\_commerce\_data\_.csv doyasındaki veri setini kullanarak RFM analizi yapınız.   
Recency hesaplarken bugünün tarihi değil en son sipariş tarihini baz alınız.**

WITH rfm\_tablo AS (

SELECT

customer\_id,

invoiceno,

quantity,

invoicedate,

unitprice

FROM

rfm

WHERE

customer\_id IS NOT NULL

AND invoiceno IS NOT NULL

AND invoiceno NOT LIKE 'C%'

AND quantity > 0

AND unitprice > 0

),

date AS (

SELECT

customer\_id,

MAX(invoicedate) AS max\_date

FROM

rfm\_tablo

GROUP BY 1

),

recency AS (

SELECT

customer\_id,

max\_date,

('2011-12-09' - max\_date::DATE) AS recency

FROM

date

),

frequency AS (

SELECT

customer\_id,

COUNT(DISTINCT invoiceno) AS frequency

FROM

rfm\_tablo

GROUP BY 1

),

monetary AS (

SELECT

customer\_id,

ROUND(SUM(quantity \* unitprice)::NUMERIC, 2) AS monetary

FROM

rfm\_tablo

GROUP BY customer\_id

),

scores AS (

SELECT

r.customer\_id,

r.recency,

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

f.frequency,

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

m.monetary,

NTILE(5) OVER (ORDER BY monetary) AS monetary\_score

FROM

recency r

JOIN

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

JOIN

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

),

merge\_tables AS (

SELECT

customer\_id,

recency\_score,

(frequency\_score + monetary\_score) AS monefreq\_score

FROM

scores

),

rfm\_Score AS (

SELECT

customer\_id,

recency\_score,

NTILE(5) OVER (ORDER BY monefreq\_score) AS mone\_fre\_score

FROM

merge\_tables

),

rfm\_segmentaion AS (

SELECT

customer\_id,

recency\_score,

mone\_fre\_score,

CASE

WHEN recency\_score >= 4 AND mone\_fre\_score = 5 THEN 'goldclass'

WHEN recency\_score >= 4 AND mone\_fre\_score >= 3 THEN 'Loyals'

WHEN recency\_score > 3 AND mone\_fre\_score <= 4 THEN 'Newcomers'

WHEN recency\_score = 3 AND mone\_fre\_score >= 4 THEN 'risky'

WHEN (recency\_score <= 3 AND recency\_score > 1) AND mone\_fre\_score <= 3 THEN 'about to sleep'

WHEN recency\_score <= 2 AND (mone\_fre\_score >= 1 AND mone\_fre\_score <= 3) THEN 'Lost'

ELSE 'Need to Be Regained'

END AS segment

FROM

rfm\_Score

)

SELECT

segment,

COUNT(DISTINCT customer\_id) AS customer\_count

FROM

rfm\_segmentaion

GROUP BY

segment

ORDER BY

customer\_count DESC;

