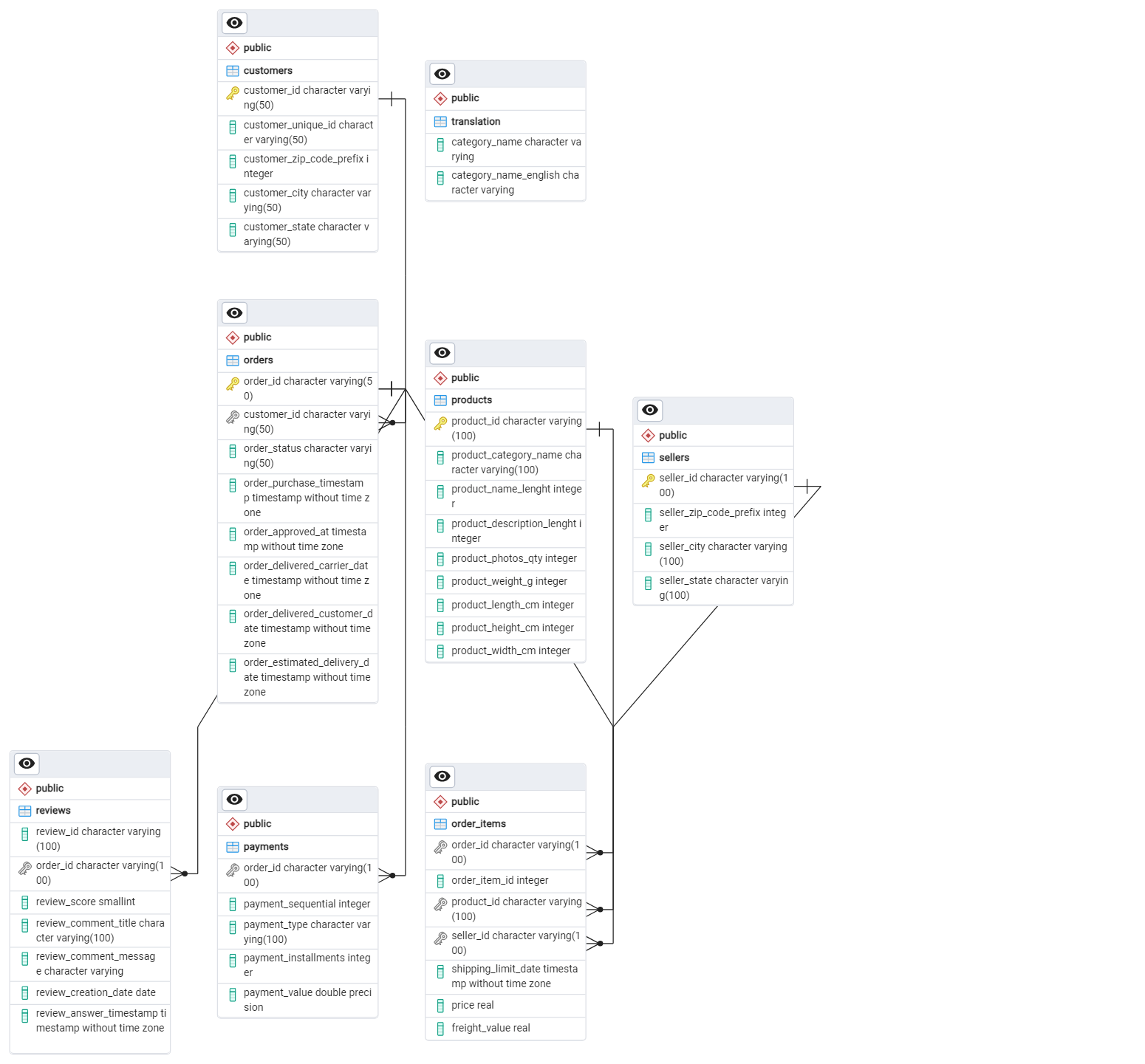
### SQL PROJE

Verilen linkteki veri setinden veri tabanını oluşturarak ERD oluşturunuz ve görsel olarak ekleyiniz. Geolocation tablosu projede kullanılmayacaktır.



#### 

#### **Örnek Case Çözümü**

-Aylık bazda toplam ödeme tutarı analizi yapıp excelde görselleştiriniz. Datayı yorumlayınız.

**SQL Query:**

**SELECT**

**(date\_trunc('month',o.order\_approved\_at))::date AS payment\_month,**

**(sum(payment\_value))::integer AS total\_payment\_value**

**FROM payments AS p**

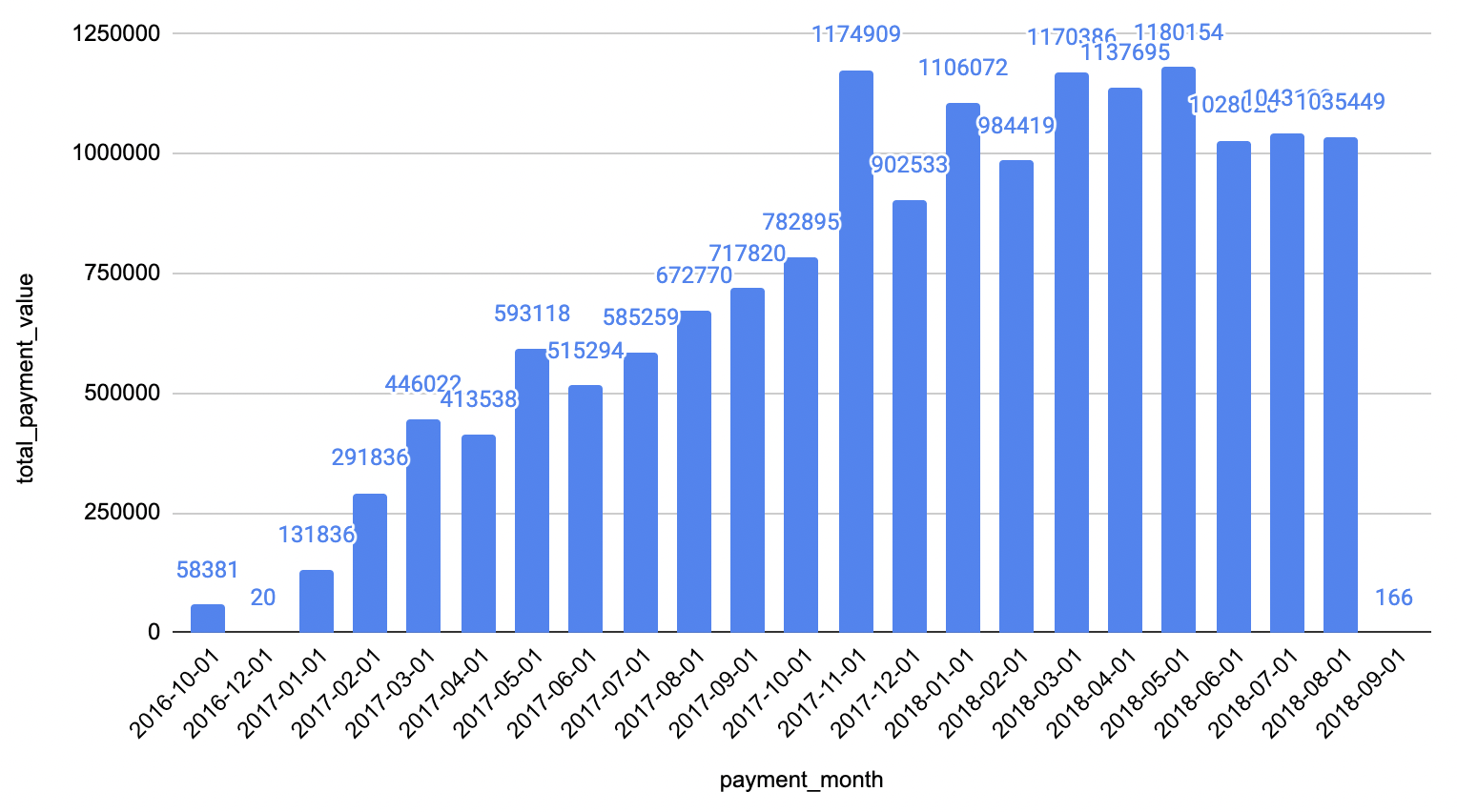
**LEFT JOIN orders AS o**

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

**WHERE o.order\_approved\_at IS NOT NULL**

**GROUP BY 1**

**ORDER BY payment\_month**



Çıktıyı incelediğimizde order sayılarında aylar içinde düzenli bir artış görüyoruz. Fakat Kasım 2017’de dramatik bir artış söz konusu, bunun sebebi ilgili ayda yapılan bir kampanya,özel bir gün (örneğin Türkiye’de bayram dönemlerinde maaşlara ek promosyon yattığında insanların daha fazla alışveriş yapması gibi) ya da tüm dünyada Kasım ayında uygulanan black friday etkili olmuş olabilir.

2018 yılında da Ocak ayında bir artış söz konusu, bunun sebebi yeni yılda yapılan maaş artışı ya da noel olabilir.

Elimizdeki veri ile yıl bazlı mevsim değerlendirmesi yapmak zor çünkü 2016 yılına ait veri çok az, 2017’de sezondan bağımsız bir artış görünüyor, 2018 yılında ise kış ve bahar ayları hemen hemen aynı verimlilikte geçmiş gibi görünüyor. Sezon incelemesi yapmak için verimiz pek yeterli değil.

#### 

#### **Case 1 : Sipariş Analizi**

##### Question 1 :

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

**SELECT TO\_CHAR(order\_approved\_at, 'YYYY-MM') AS month\_year, COUNT(order\_id) AS total\_orders**

**FROM orders**

**GROUP BY month\_year**

**ORDER BY month\_year;**

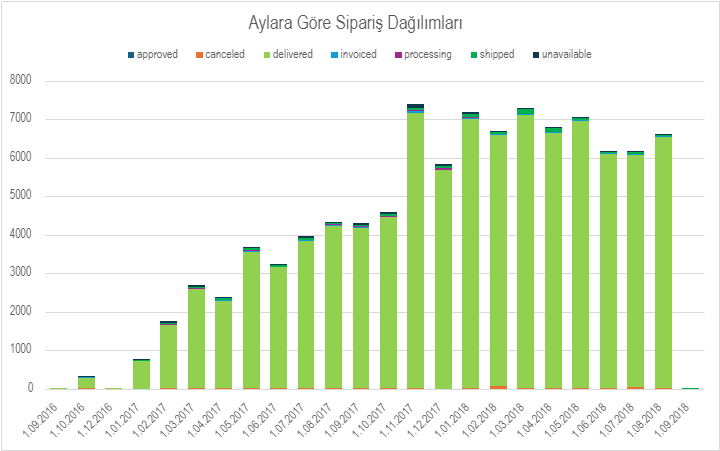
##### Question 2 :

-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 month\_year, order\_status, COUNT(order\_id) AS total\_orders**

**FROM orders**

**GROUP BY month\_year, order\_status**

**ORDER BY month\_year, order\_status;**

2016'da sipariş sayısı oldukça düşükken, 2017 boyunca ve 2018 başlarında sipariş sayısında belirgin bir artış görülüyor. Bu durum, e-ticaret platformunun büyüdüğüne ya da kullanıcıların alışveriş alışkanlıklarının değiştiğine işaret edebilir.

Kasım 2017’deki zirve, büyük ihtimalle özel indirim dönemlerinin etkisi ile açıklanabilir. Benzer bir yükseliş, Ocak 2018'de yeni yıl alışverişlerine bağlı olarak gözlemleniyor.

##### Question 3 :

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

**WITH ranked\_orders AS (**

**SELECT t.category\_name\_english, TO\_CHAR(o.order\_approved\_at, 'DD-MM-YYYY') AS date, COUNT(oi.order\_id) AS total\_orders,**

**ROW\_NUMBER() OVER (PARTITION BY TO\_CHAR(o.order\_approved\_at, 'DD-MM-YYYY') ORDER BY COUNT(oi.order\_id) DESC) AS rank**

**FROM products p**

**JOIN order\_items oi ON p.product\_id = oi.product\_id**

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

**JOIN translation t ON p.product\_category\_name = t.category\_name**

**WHERE o.order\_approved\_at IS NOT NULL**

**AND (**

**(EXTRACT(MONTH FROM o.order\_approved\_at) = 2 AND EXTRACT(DAY FROM o.order\_approved\_at) = 14 AND EXTRACT(YEAR FROM o.order\_approved\_at) IN (2017, 2018)) -- Valentine's Day**

**OR (EXTRACT(MONTH FROM o.order\_approved\_at) = 12 AND EXTRACT(DAY FROM o.order\_approved\_at) = 31 AND EXTRACT(YEAR FROM o.order\_approved\_at) = 2017) -- New Year's Eve**

**OR (EXTRACT(MONTH FROM o.order\_approved\_at) = 5 AND EXTRACT(DAY FROM o.order\_approved\_at) = 14 AND EXTRACT(YEAR FROM o.order\_approved\_at) = 2017) -- Mother's Day 2017**

**OR (EXTRACT(MONTH FROM o.order\_approved\_at) = 5 AND EXTRACT(DAY FROM o.order\_approved\_at) = 13 AND EXTRACT(YEAR FROM o.order\_approved\_at) = 2018) -- Mother's Day 2018**

**OR (EXTRACT(MONTH FROM o.order\_approved\_at) = 6 AND EXTRACT(DAY FROM o.order\_approved\_at) = 18 AND EXTRACT(YEAR FROM o.order\_approved\_at) = 2017) -- Father's Day 2017**

**OR (EXTRACT(MONTH FROM o.order\_approved\_at) = 6 AND EXTRACT(DAY FROM o.order\_approved\_at) = 17 AND EXTRACT(YEAR FROM o.order\_approved\_at) = 2018) -- Father's Day 2018**

**)**

**GROUP BY t.category\_name\_english, date**

**)**

**SELECT category\_name\_english, date, total\_orders,**

**CASE**

**WHEN date = '14-02-2017' THEN 'Valentine''s Day'**

**WHEN date = '14-02-2018' THEN 'Valentine''s Day'**

**WHEN date = '31-12-2017' THEN 'New Year''s Eve'**

**WHEN date = '14-05-2017' THEN 'Mother''s Day'**

**WHEN date = '13-05-2018' THEN 'Mother''s Day'**

**WHEN date = '18-06-2017' THEN 'Father''s Day'**

**WHEN date = '17-06-2018' THEN 'Father''s Day'**

**ELSE 'Other'**

**END AS special\_day**

**FROM ranked\_orders**

**WHERE rank <= 3**

**ORDER BY TO\_DATE(date, 'DD-MM-YYYY') ASC, total\_orders DESC;**

Her özel gün, tüketici tercihleri açısından belirgin bir desen göstermektedir. Sevgililer Günü'nde ev ve dekorasyon ile ilgili ürünler popülerken, Anneler Günü'nde kişisel bakım ürünleri öne çıkmaktadır. Babalar Günü'nde ise saat gibi aksesuarlar tercih edilmektedir. Bu bulgular, özel günlerde en çok tercih edilen ürün kategorileri için hedefli pazarlama kampanyaları veya promosyonlar planlamada yardımcı olabilir, bu sayede satışların maksimum seviyeye çıkartılması sağlanabilir.

##### 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.

**--Haftanın günleri**

**SELECT**

**TO\_CHAR(order\_approved\_at, 'Day') AS day\_of\_week,**

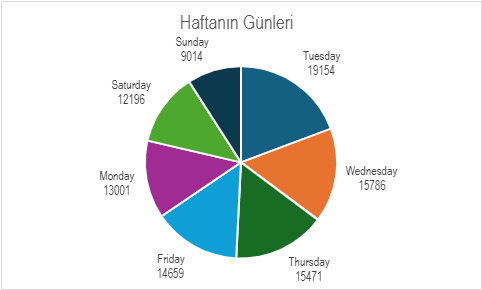
**COUNT(order\_id) AS total\_orders**

**FROM orders**

**WHERE order\_approved\_at IS NOT NULL**

**GROUP BY 1**

**ORDER BY 2 DESC;**

****

Salı gününün en yoğun gün olarak öne çıkması, alışveriş siteleri için kampanyaların ve promosyonların bu günlerde yoğunlaştırılabileceğini gösteriyor. Ayrıca, haftanın diğer yoğun günleri olan Çarşamba ve Perşembe de dikkate alınabilir. Hafta sonu özellikle Pazar gününün daha düşük sipariş aldığı göz önüne alındığında, hafta sonu alışveriş teşviklerini artıracak kampanyalar planlanabilir.

**--Ay günleri**

**SELECT**

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

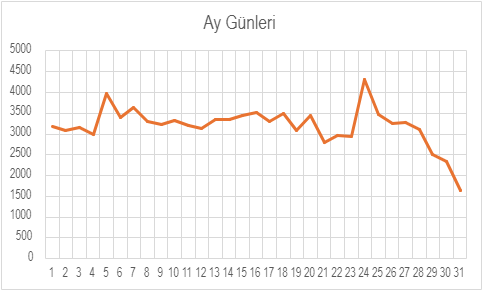
**COUNT(order\_id) AS total\_orders**

**FROM orders**

**WHERE order\_approved\_at IS NOT NULL**

**GROUP BY 1**

**ORDER BY 1;**



Ayın 23-27 arasındaki dönemin yoğun sipariş aldığı göz önüne alındığında, e-ticaret kampanyaları bu günlerde yoğunlaştırılabilir.

Ay sonundaki düşüş, tüketicilerin bütçelerinin ayın sonuna doğru daraldığını ve harcamalarını azaltma eğiliminde olduklarını göstermektedir. Bu yüzden, ayın son günlerine özel indirimler veya fırsatlar sunmak, talebi artırmak için etkili bir strateji olabilir.

#### **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.

**Örneğin; Sibel Çanakkale’den 3, Muğla’dan 8 ve İstanbul’dan 10 sipariş olmak üzere 3 farklı şehirden sipariş veriyor. Sibel’in şehrini en çok sipariş verdiği şehir olan İstanbul olarak seçmelisiniz ve Sibel’in yaptığı siparişleri İstanbul’dan 21 sipariş vermiş şekilde görünmelidir.**

**WITH customer\_city\_orders AS (**

**SELECT c.customer\_id, 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\_id, c.customer\_city**

**),**

**customer\_primary\_city AS (**

**SELECT customer\_id, customer\_city, order\_count,**

**ROW\_NUMBER() OVER (PARTITION BY customer\_id ORDER BY order\_count DESC) AS city\_rank**

**FROM customer\_city\_orders**

**),**

**primary\_city\_orders AS (**

**SELECT cpc.customer\_id, cpc.customer\_city AS primary\_city, cpc.order\_count**

**FROM customer\_primary\_city cpc**

**WHERE cpc.city\_rank = 1**

**)**

**SELECT primary\_city, SUM(order\_count) AS total\_orders**

**FROM primary\_city\_orders**

**GROUP BY primary\_city**

**ORDER BY total\_orders DESC;**

#### **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.

**SELECT oi.seller\_id,**

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

**AVG(o.order\_delivered\_customer\_date - o.order\_approved\_at) AS avg\_delivery\_time\_hours,**

**ROUND(AVG(r.review\_score), 2) AS avg\_review\_score**

**FROM orders o**

**JOIN order\_items oi ON o.order\_id = oi.order\_id**

**LEFT JOIN reviews r ON o.order\_id = r.order\_id**

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

**AND o.order\_approved\_at IS NOT NULL**

**AND o.order\_status = 'delivered'**

**GROUP BY oi.seller\_id**

**ORDER BY avg\_delivery\_time\_hours ASC**

**LIMIT 5;**

Top 5 getirdiğimizde en hızlı teslimat yapan satıcıların 1’er sipariş aldıklarını görüyoruz, elimizdeki bu verilerle hızlı teslimatın, müşteri memnuniyeti üzerinde olumlu bir etki yarattığını, zira en hızlı teslimat yapan bu satıcılar genellikle yüksek puanlar aldığını gözlemliyoruz. Ancak, daha fazla veri ile ortalama teslimat süreleri ve puanlar arasında daha güvenilir bir ilişki kurulabilir.

##### Question 2 :

-Hangi satıcılar daha fazla kategoriye ait ürün satışı yapmaktadır?

Fazla kategoriye sahip satıcıların order sayıları da fazla mı?

**WITH seller\_category\_orders AS (**

**SELECT oi.seller\_id,**

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

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

**FROM order\_items oi**

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

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

**GROUP BY 1**

**)**

**SELECT category\_count,**

**COUNT(seller\_id) AS num\_sellers,**

**ROUND(AVG(order\_count), 2) AS avg\_order\_count,**

**MAX(order\_count) AS max\_order\_count,**

**MIN(order\_count) AS min\_order\_count**

**FROM seller\_category\_orders**

**GROUP BY 1**

**ORDER BY 1 DESC;**

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

##### Question 1 :

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

**SELECT c.customer\_state,**

**ROUND(AVG(p.payment\_installments), 2) AS avg\_installments**

**FROM payments p**

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

**JOIN customers c ON o.customer\_id = c.customer\_id**

**WHERE p.payment\_type = 'credit\_card'**

**GROUP BY c.customer\_state**

**ORDER BY avg\_installments DESC;**

Bu analiz, taksitli ödeme eğilimlerinin bölgesel olarak farklılık gösterdiğini ortaya koyuyor. Özellikle PB, SE ve AC gibi bölgelerde yüksek taksit kullanımı öne çıkarken, daha büyük şehirlerde daha düşük taksit eğilimleri gözlemleniyor.

##### Question 2 :

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

**SELECT p.payment\_type,**

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

**ROUND(SUM(p.payment\_value)::numeric, 2) AS total\_payment\_value**

**FROM payments p**

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

**WHERE o.order\_status = 'delivered'**

**GROUP BY p.payment\_type**

**ORDER BY order\_count DESC;**

##### Question 3 :

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

**SELECT p.product\_category\_name,**

**COUNT(CASE WHEN pay.payment\_installments = 1 THEN 1 END) AS single\_payment\_count,**

**COUNT(CASE WHEN pay.payment\_installments > 1 AND pay.payment\_type = 'credit\_card' THEN 1 END) AS installment\_payment\_count,**

**ROUND(100.0 \* COUNT(CASE WHEN pay.payment\_installments > 1 AND pay.payment\_type = 'credit\_card' THEN 1 END) / COUNT(\*), 2) AS installment\_percentage**

**FROM payments pay**

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

**JOIN order\_items oi ON o.order\_id = oi.order\_id**

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

**WHERE o.order\_status = 'delivered'**

**GROUP BY p.product\_category\_name**

**ORDER BY installment\_percentage DESC;**

Taksitli ödeme kullanımı, ürünün maliyeti ile doğrudan ilişkilidir. Pcs (bilgisayarlar),

la\_cuisine (mutfak eşyaları) ve moveis\_colchao\_e\_estofado (yatak ve döşeme mobilyaları) gibi yüksek maliyetli ürün kategorilerinde taksitli ödeme oranı daha yüksek olurken, günlük tüketim ürünlerinde tek çekim daha baskındır.

#### **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.

**--Recency**

**SELECT MAX(invoicedate) AS last\_order\_date FROM rfm;**

**--2011-12-09 12:50:00**

**SELECT customer\_id,**

**EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) AS recency**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id;**

**--Frequency**

**SELECT customer\_id,**

**COUNT(DISTINCT invoiceno) AS frequency**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id;**

**--Monetary**

**SELECT customer\_id,**

**ROUND(SUM(quantity \* unitprinbvgdrvsece)::numeric, 2) AS monetary**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id;**

**--Complete RFM Table**

**SELECT customer\_id,**

**EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) AS recency,**

**COUNT(DISTINCT invoiceno) AS frequency,**

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

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id;**

**--Assigning RFM Scores**

**SELECT customer\_id,**

**EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) AS recency,**

**COUNT(DISTINCT invoiceno) AS frequency,**

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

**NTILE(5) OVER (ORDER BY EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) ASC) AS recency\_score,**

**NTILE(5) OVER (ORDER BY COUNT(DISTINCT invoiceno) DESC) AS frequency\_score,**

**NTILE(5) OVER (ORDER BY ROUND(SUM(quantity \* unitprice)::numeric, 2) DESC) AS monetary\_score**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id;**

**--Combined RFM Score**

**SELECT customer\_id, recency, frequency, monetary, recency\_score, frequency\_score, monetary\_score,**

**CONCAT(recency\_score, frequency\_score, monetary\_score) AS rfm\_score**

**FROM (**

**SELECT customer\_id,**

**EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) AS recency,**

**COUNT(DISTINCT invoiceno) AS frequency,**

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

**NTILE(5) OVER (ORDER BY EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) ASC) AS recency\_score,**

**NTILE(5) OVER (ORDER BY COUNT(DISTINCT invoiceno) DESC) AS frequency\_score,**

**NTILE(5) OVER (ORDER BY ROUND(SUM(quantity \* unitprice)::numeric, 2) DESC) AS monetary\_score**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id**

**) AS rfm\_scored;**

**--Customer Segmentation**

**SELECT**

**customer\_id,**

**recency,**

**frequency,**

**monetary,**

**recency\_score,**

**frequency\_score,**

**monetary\_score,**

**CONCAT(recency\_score, frequency\_score, monetary\_score) AS rfm\_score,**

**CASE**

**WHEN recency\_score = 5 AND frequency\_score >= 4 AND monetary\_score >= 4 THEN 'Champions'**

**WHEN recency\_score >= 4 AND frequency\_score >= 3 AND monetary\_score >= 3 THEN 'Loyal Customers'**

**WHEN recency\_score <= 2 AND frequency\_score >= 3 AND monetary\_score >= 3 THEN 'At Risk'**

**WHEN recency\_score >= 3 AND frequency\_score <= 2 AND monetary\_score <= 3 THEN 'New Customers'**

**WHEN recency\_score <= 3 AND frequency\_score <= 3 AND monetary\_score <= 3 THEN 'Lost Customers'**

**ELSE 'General Customers'**

**END AS customer\_segment**

**FROM (**

**SELECT customer\_id,**

**EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) AS recency,**

**COUNT(DISTINCT invoiceno) AS frequency,**

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

**NTILE(5) OVER (ORDER BY EXTRACT(DAY FROM AGE('2011-12-09', MAX(invoicedate))) ASC) AS recency\_score,**

**NTILE(5) OVER (ORDER BY COUNT(DISTINCT invoiceno) DESC) AS frequency\_score,**

**NTILE(5) OVER (ORDER BY ROUND(SUM(quantity \* unitprice)::numeric, 2) DESC) AS monetary\_score**

**FROM rfm**

**WHERE customer\_id IS NOT NULL**

**GROUP BY customer\_id**

**) AS rfm\_scored;**

