

Akbank Teknoloji Okuryazarlığı Proje Ödevi

Bu doküman, users.csv ve transactions.csv veri setleriyle ödevde istenen analiz ve görselleştirmeleri hem Python/Colab hem de Power BI üzerinde adım adım nasıl yaptığımı anlatır.

Kodun genel yapısı

Kod bir akış izliyor. önce ayarlar. sonra veriyi okuma ve temizleme. ardından tabloları birleştirme. en sonda grafikler ve kısa özetler. hepsi aynı notebook içinde.

- adımlar: **kurulum** → **veri yükleme/temizlik** → **birleştirme** → **analiz/grafikler** → **özet**
- ortak anahtar: **User** (iki tabloyu bununla bağlıyoruz)

	User	Card	Year	Month	Day	Time	Amount	Use Chip	Merchant Name	Merchant City	State	Zipcode	Latitude	Longitude	Per Capita Income	Yearly Income	Total Debt	FICO Score	Num Credit Cards	IncomeNumeric
0	102	0	2002	9	1	2025-08-18 06:21:00	2.08	Swipe Transaction	3527213246127876953	La Verne	IL	60153	41.88	-87.84	\$15451	\$22158	\$19101	681	5	22158
1	435	0	2002	9	1	2025-08-18 06:42:00	5.59	Swipe Transaction	-727612092139916043	Monterey Park	OH	44102	41.47	-81.67	\$14737	\$30049	\$80474	700	1	30049
2	348	0	2002	9	2	2025-08-18 06:22:00	132.12	Swipe Transaction	-727612092139916043	Monterey Park	IL	60616	41.83	-87.68	\$25365	\$41793	\$1141	730	5	41793
3	270	0	2002	9	2	2025-08-18 17:45:00	60.91	Swipe Transaction	3414527459579106770	Monterey Park	NY	14817	42.38	-76.39	\$20159	\$41106	\$106963	762	3	41106
4	106	0	2002	9	3	2025-08-18 06:23:00	118.74	Swipe Transaction	5817218446178736267	La Verne	NC	27502	35.72	-78.84	\$32485	\$66230	\$90248	742	3	66230

Hangi kütüphaneler kullanıldı?

- **pandas**: csv okuma, kolon seçme, veri dönüştürme, gruptama, birleştirme.
- **numpy**: boş değer ve sayısal yardımcılar.
- **matplotlib**: grafik çizme (bar, line, scatter, pie/donut).
- **seaborn** (isteğe bağlı): daha hoş bir tema vermek için.

kısaca: pandas veri, matplotlib grafik; seaborn sadece görünüm.

Yardımcı fonksiyonlar

- **pick_column(df, candidates)**
Kolon adı farklı yazılsa bile doğru sütunu bulur. örnek: *User*, *UserID*, *user_id* → hepsi “User” olarak yakalanır. kod böylece farklı csv sürümlerinde de çalışır.
- **to_numeric_money(s)**
Para metnini sayıya çevirir. \$, €, £, , gibi işaretleri siler. ondalık ayracı sorunlarını düzeltir. sayı yapamazsa **NaN** yapar.

bu iki fonksiyon kodu esnek ve dayanıklı yapar.

Veri yükleme ve hazırlık

- **CSV okuma:** users.csv ve transactions.csv dosyaları /content'ten okunur.
- **Kolon adları:** baştaki/sondaki boşluklar temizlenir.
- **Tutar (Amount):** para metni sayıya çevrilir.
- **Zaman (Time):** tarih-saat olarak okunur; **Hour** sütunu üretilir (0–23).
- **Gelir:** Yearly Income - Person metninden **IncomeNumeric** oluşturulur.

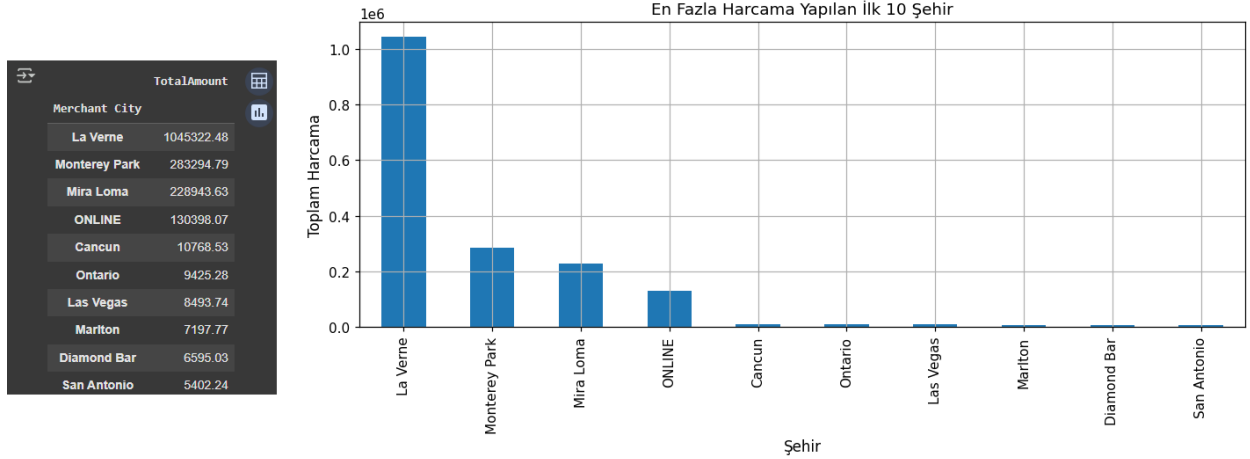
Birleştirme

- iki tablo **User** alanı üzerinden **left merge** ile birleştirilir.
- sonuç DataFrame analiz için kullanılır.

Analiz ve grafikler

1) En çok harcama yapılan şehirler (Top 10, bar)

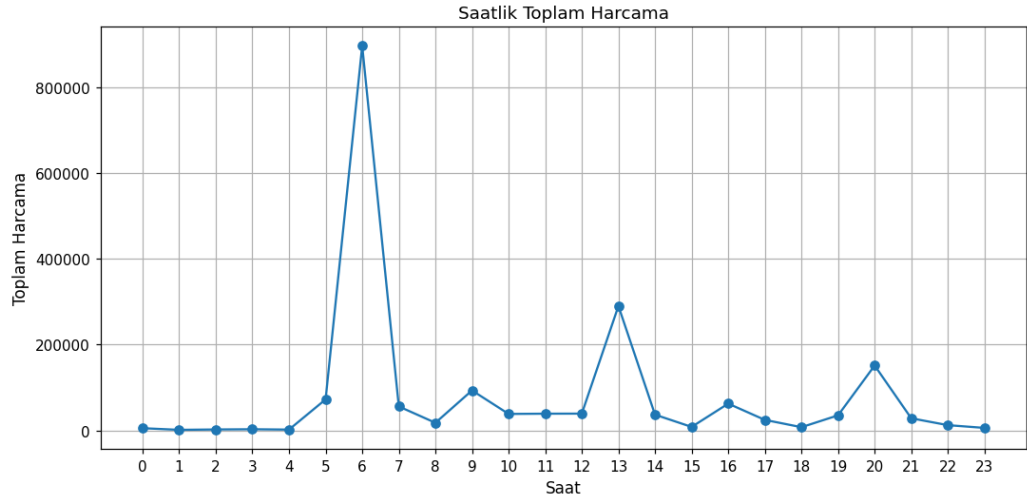
- şehir bazında **Amount toplamı** alınır.
- en yüksek 10 şehir bar grafikte gösterilir.



2) Saatlik harcama (line)

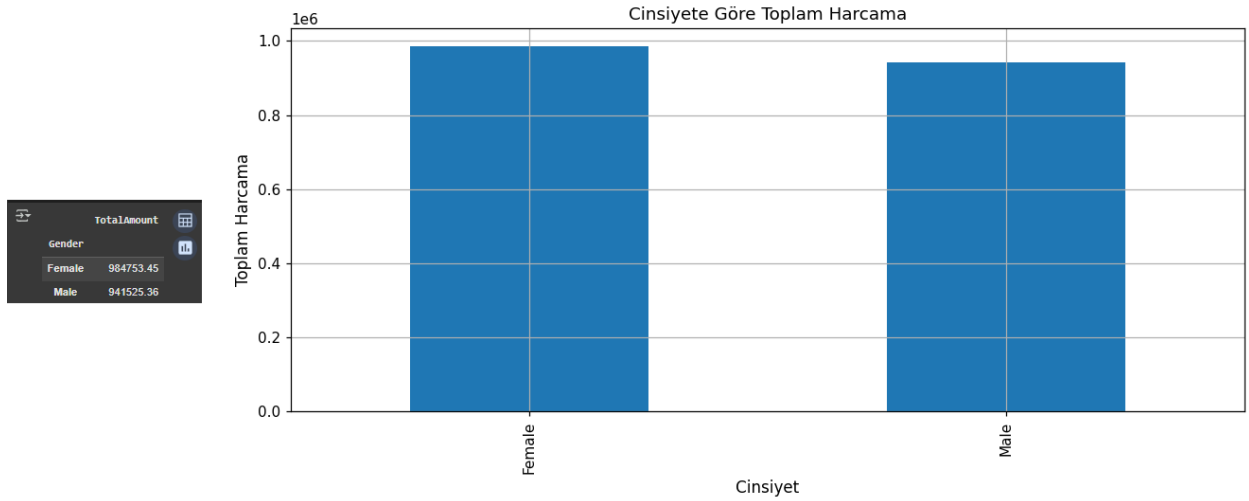
- **Hour** bazında **Amount toplamı** çıkarılır.
- 0–23 saat aralığı çizgi grafikte gösterilir.

Hour	TotalAmount
0	5001.90
1	1177.99
2	2042.49
3	2560.96
4	1753.44
5	72221.95
6	897523.69
7	55786.57
8	17614.61
9	93127.71
10	38305.72
11	38894.18
12	39054.18
13	290185.72
14	36560.55
15	8017.31
16	62383.46
17	24424.23
18	7373.05
19	34985.01
20	151329.55
21	28145.99
22	12124.46
23	5684.09



3) Cinsiyete göre harcama (bar)

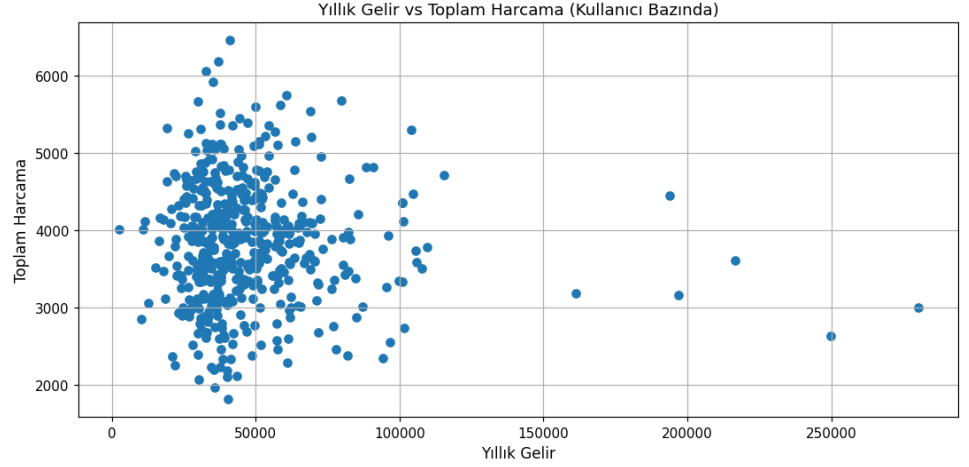
- **Gender** bazında **Amount toplamı** hesaplanır.
- bar grafikte karşılaştırılır.



4) Gelir vs harcama (scatter)

- kullanıcı bazında toplam harcama (**TotalSpend**) hesaplanır.
- IncomeNumeric ile eşleştirilir.
- scatter grafikte **X=gelir**, **Y=harcama** olarak çizilir.
- basit korelasyon değeri yazdırılır.

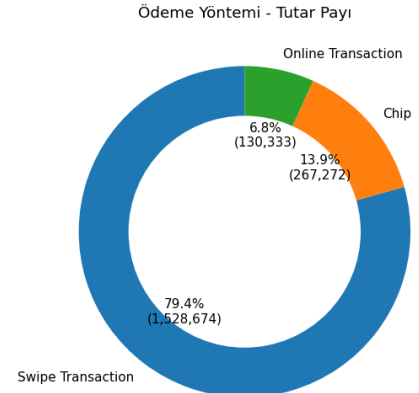
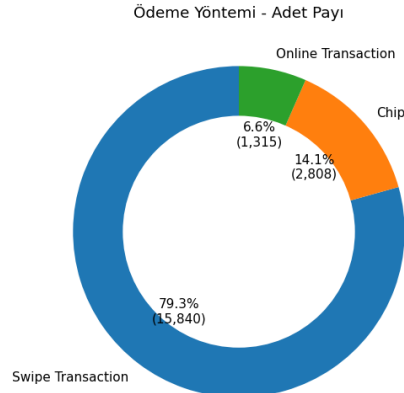
User	TotalSpend	IncomeNumeric
0	0	4140.96
1	1	3359.81
2	2	3367.61
3	3	2636.14
4	4	3782.87



5) Ödeme yöntemi: Chip vs Swipe (donut)

- Use Chip alanı normalize edilir → **PayMethod** (Chip/Swipe/Unknown).
- iki donut önerilir: **Adet payı** ve **Tutar payı**.

PayMethod	Count
Swipe Transaction	15540
Chip	2808
Online Transaction	1315
TotalAmount	PayMethod
Swipe Transaction	1528674.00
Chip	267272.11
Online Transaction	130332.70



Veri temizliği ve hazırlık

Önce iki dosyada baştaki-sondaki boşlukları ve görünmez karakterleri temizledim, kolon adlarını sadeleştirdim. Parasal alanlarda \$ / € / ₺ ve binlik ayrıçları sildim; ondalık ayracı tek biçime getirip sayıya çevirdim. Dönüşmeyen değerler NaN bırakıldı ki akış kesilmesin. Yıl-Ay-Gün ve Saat bilgilerini tek bir **datetime** alanında birleştirip buradan **Hour (0-23)** ve gerekirse **gün adı** gibi türevleri ürettim. Kategorileri standartlaştırdım: **Gender** → **Male/Female**, **Use Chip** → **Chip/Swipe/Online**; **Merchant City** üzerinde trim/capitalize uyguladım. **User** anahtarının tipi iki tabloda aynı olacak şekilde ayarlandı; kopya ve bariz hatalı satırlar elendi; **Zip/Zipcode** metin tutuldu ki baştaki sıfırlar kaybolmasın. Negatif tutarları

“iade/chargeback” olarak işaretledim ve analize alıp almamayı bilinçli bir kararla yönettim. Son olarak birleştirme sonrası eşleşme oranını, boşluk sayılarını ve toplamaları kontrol ederek veriyi “analize hazır” hale getirdim.

Kısa özet / çıkarım

Harcama dağılımında şehirler arasında **X** öne çıkıyor; onu **Y** ve **Z** izliyor. Bu sonuç, alışverişin belirli lokasyonlarda yoğunlaştığını gösteriyor. Zaman kırılımında harcamalar gün içinde eşit dağılmıyor; zirve **18:00–21:00** aralığında. Cinsiyete göre toplam harcamada **Male/Female** tarafı bir miktar daha yüksek; fark büyük değilse kategori ve kullanıcı sayısı etkisi ayrıca not edilebilir. Gelir–harcama ilişkisinde korelasyon **zayıf/orta/güçlü** düzeyde; yani gelir arttıkça harcama da aynı yönde **artıyor** ya da ilişki **çok net değil** (eldeki değere göre yorumlanır). Ödeme yönteminde **Swipe Transaction** hem adet hem tutarda en büyük paya sahip; **Chip** ikinci, **Online Transaction** daha düşük payda kalıyor. Adet ve tutar paylarının birbirine yakın olup olmadığı da kısaca belirtilir.

Dayanıklılık ve pratik notlar

Kolon adları her zaman aynı gelmeyebileceği için `pick_column` benzer isimleri tarayıp doğru sütunu bularak akışın bozulmasını önler. Para ve tarih alanlarında beklenmedik karakterler veya formatlar çıktığında değerler otomatik olarak **NaN** olur ve analiz güvenle devam eder. Saat kolonu yoksa saat grafiği otomatik atlanır ve bilgi mesajı verilir. Şehir bazlı grafikte **Merchant City**, harcama toplamında **Amount** kullanılmalıdır. Scatter grafiğinde her nokta **bir kullanıcıyı** temsil eder; bu yüzden gelir eksene **özetlemeden** (ham değer) konur, harcama ise kullanıcı bazında toplanarak kullanılır. Bu kurallarla metin, tablosuz da net ve tutarlı bir özet sunar.