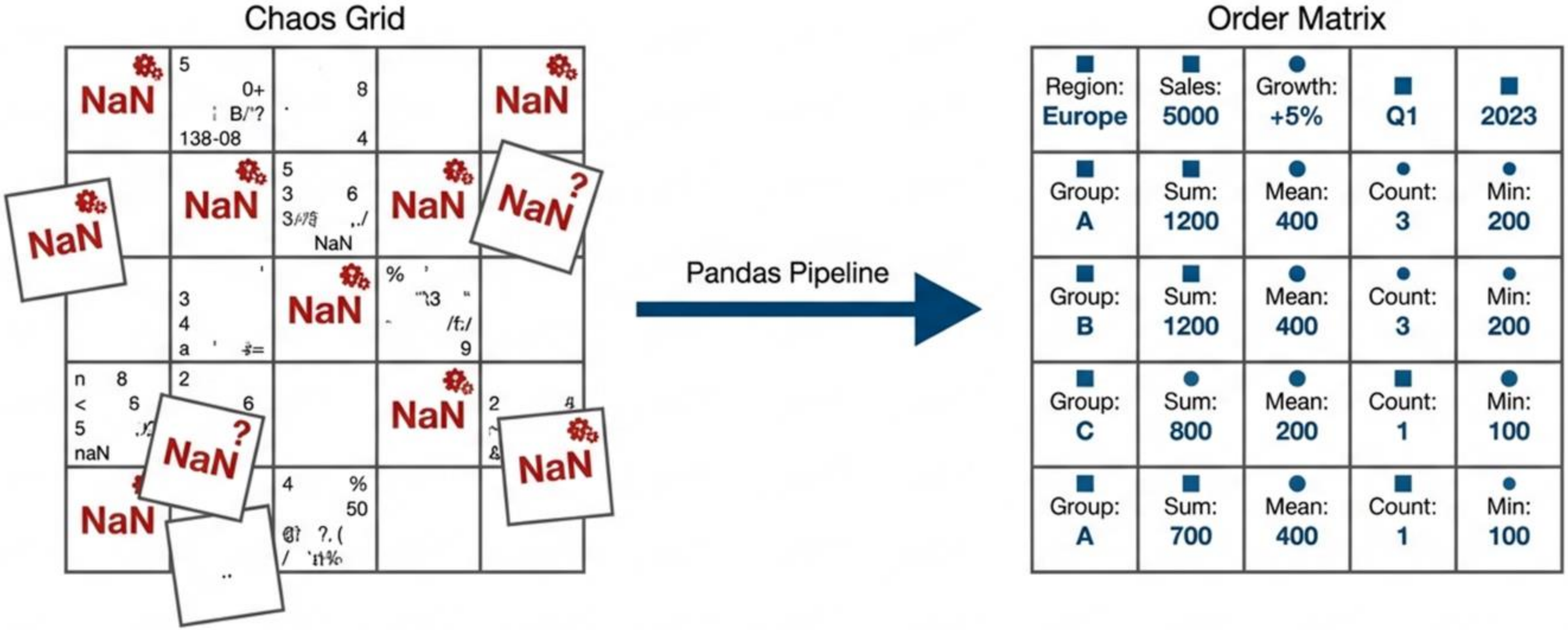
Pandas-2: Temizleme ve Özetleme

Gerçek Veri Dünyası: NaN → Groupby/Pivot → Merge/Join



Veri Analizi, kaosu düzene sokma sanatıdır.

Gerçek Dünyada Veri Asla Mükemmel Değildir

- NaN = Veri eksik / bilinmiyor.
  - Kullanıcı girişi eksiklikleri
  - Sistem hataları
  - Ölçüm alınamaması
  - Veri kaybı

Damaged Data Table

ID	Name	Age	Score
101	Ali	25	88
102	Ayşe		92
103	Veli	28	76
NaN	Mehmet	30	77
105	Seda	22	95
106	Can	35	NaN

Teşhis: Eksik Veri Kontrolü

Sorunu çözmeden önce boyutunu anlamalıyız.

Komutlar:

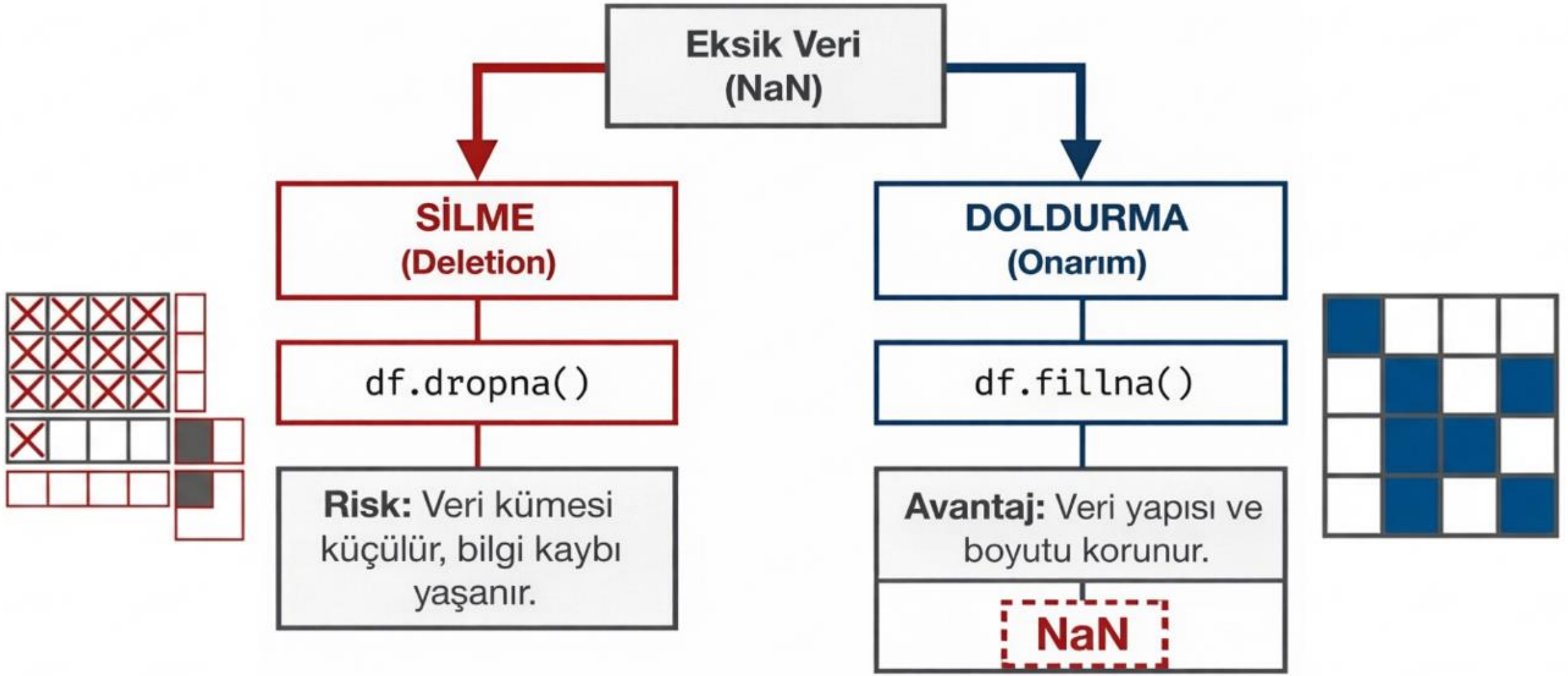
```
df.isna().sum() # Sütun bazlı sayım

df[df['Age'].isna()] # Satır tespiti
```

Health Report

```
Name:   0
Age:    14
City:   3
Score: 0
```

Silmek kolay ama bazen yanlıştır. Önce anlamaya çalışın.

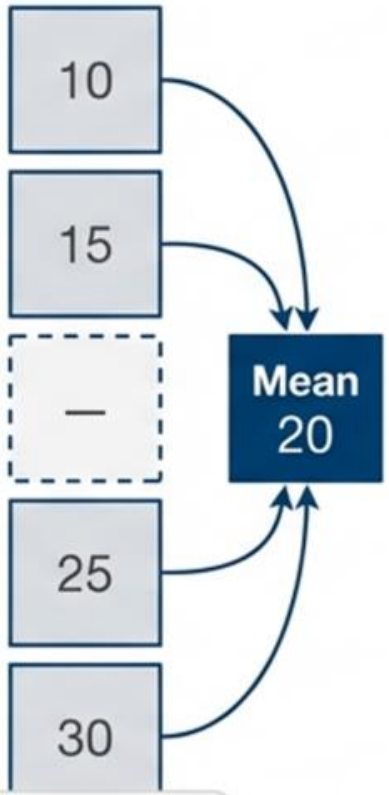
Stratejik Karar: Silmek mi, Doldurmak mı?



Bilimsel Yaklaşım: Veriyi İstatistikle Onarmak

Sayısal Veriler

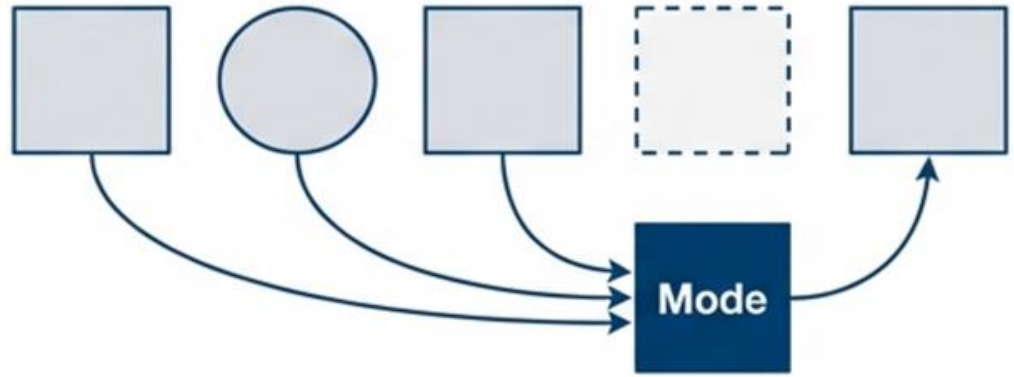
Ortalama (Mean) veya Medyan kullanılır.




```
df['Age'].fillna(df['Age'].mean())
```

Kategorik Veriler

Mod (En sık tekrar eden değer) kullanılır.




```
df['City'].fillna(df['City'].mode()[0])
```

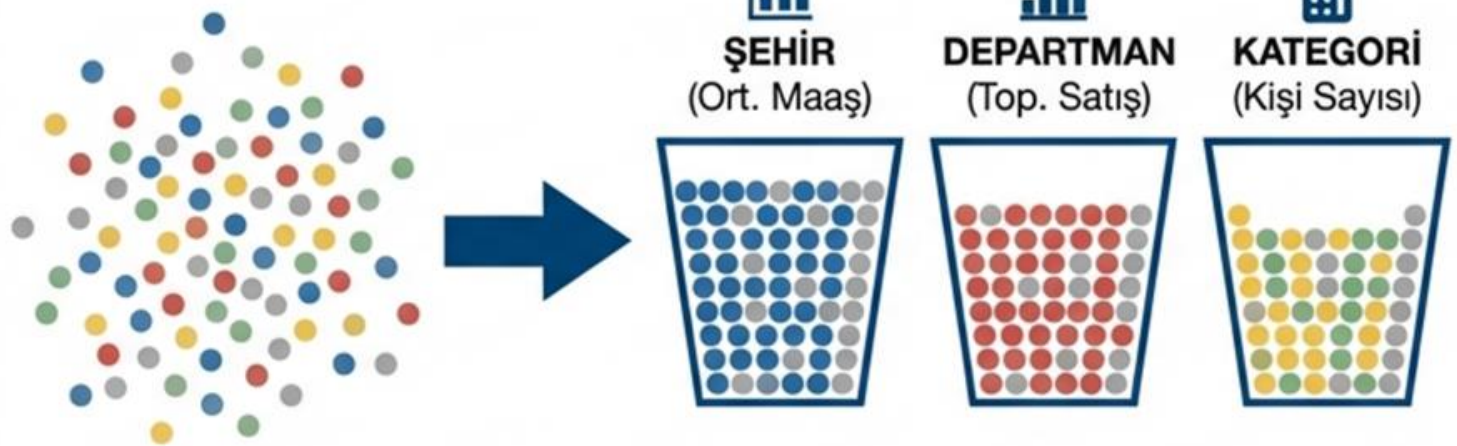
Mikro Düzeyden Makro Özetlere Geçiş

GroupBy: Veriyi gruplara ayırır ve özetler.

Soru 1: Şehre göre ortalama maaş nedir?

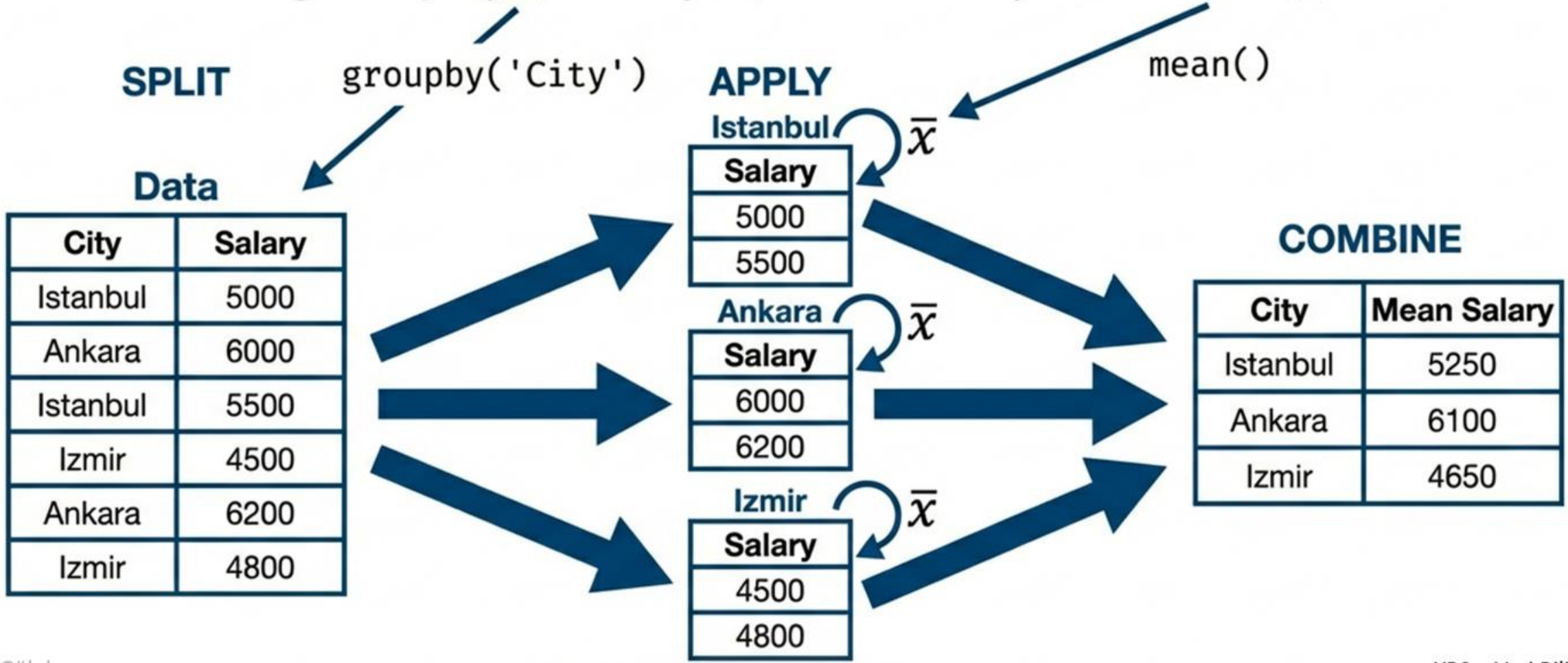
Soru 2: Departmana göre toplam satış kaç?

Soru 3: Kategoriye göre kaç kişi var?



GroupBy Mekanizması

```
df.groupby('City')['Salary'].mean()
```



Çok Boyutlu Analiz: Pivot Table

Rapor formatında, matris şeklinde özet tablo.

- Satır (Index): Kategori 1 (Örn: Şehir)
- Sütun (Columns): Kategori 2 (Örn: Cinsiyet)
- Değer (Values): Özetlenecek veri (Örn: Maaş)

```
pd.pivot_table(df, index='City',
    columns='Gender', values='Salary')
```

		Gender	
		F	M
City	Istanbul	₺5000	₺5500
	Ankara	₺6000	₺6200

Karşılaştırma: GroupBy vs Pivot Table

GroupBy

Category	Value
City A	1200
City B	1500
City A	1100
City C	900
City B	1300
City A	1250

Tek yönlü özet.
Basit listeler için ideal.

Pivot Table

		Product		
		Type 1	Type 2	Type 3
City	City A	3550	3500	2100
	City B	2800	2800	1600
	City C	1100	1300	900

İki boyutlu matris.
Çapraz analiz için rapor formatı.

Entegrasyon: Veri Silolarını Birleştirmek

Gerçek hayatta veriler farklı tablolarda yaşar.

Müşteriler (Customers)

MüşteriID	Ad Soyad
1	Mehmet Kaya
2	Elif Yıldız
3	Özkan Demir
4	Ayşe Soylu

Siparişler (Orders)

SiparişID	MüşteriID	Tutar
1	1	400.00
2	2	350.00
3	3	500.00
4	4	130.00

- Müşteri Tablosu + Sipariş Tablosu
- Öğrenci Tablosu + Not Tablosu
- Ürün Tablosu + Satış Tablosu

```
Çözüm: Merge (Join) İşlemi
```

Ortak Anahtar: Merge İşleminin Temeli

İki tabloyu birbirine bağlayan ortak sütun.

Table A: df_customers

customer_id	name	email
1	Ali Yılmaz	ali@email.com
2	Ayşe Kaya	ayse@email.com
3	Veli Demir	veli@email.com

Table B: df_orders

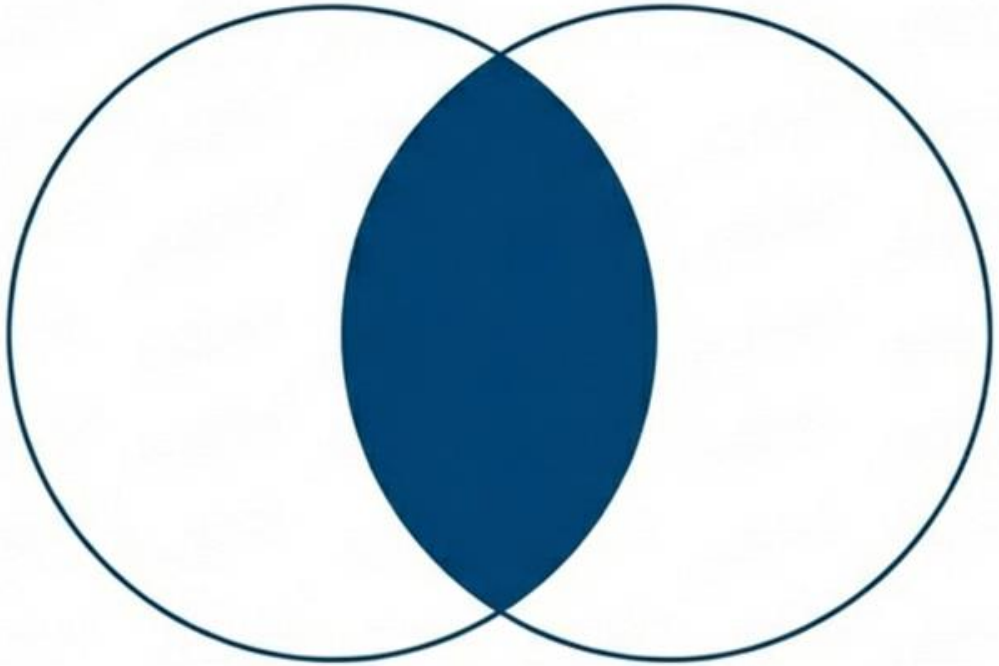
order_id	customer_id	amount
101	1	150.00
102	2	200.00
103	1	50.00

KEY

```
pd.merge(df_customers, df_orders, on='customer_id')
```

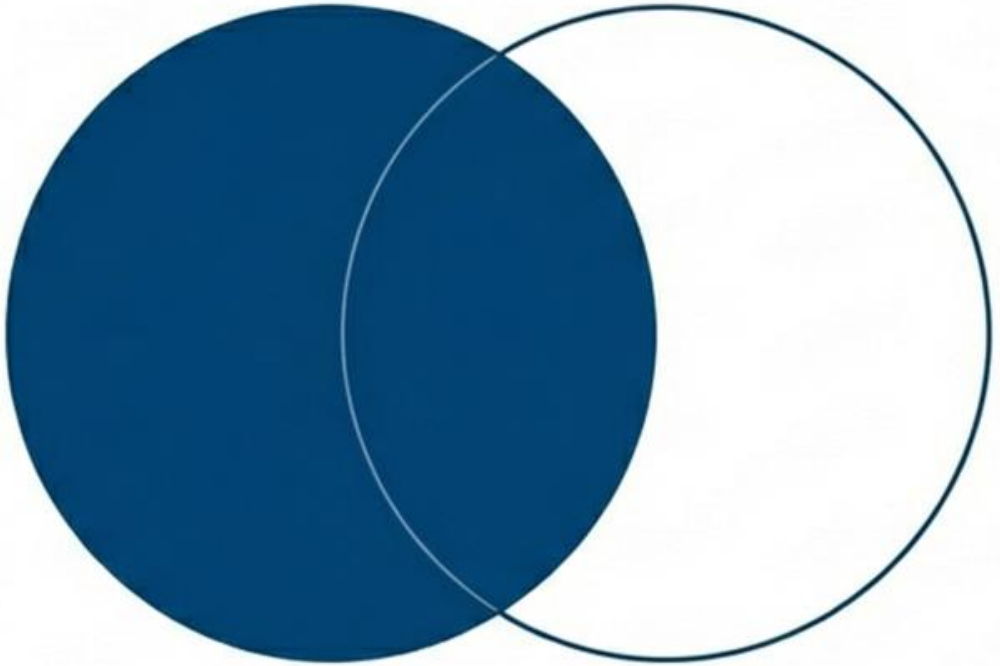
Kapsama Mantığı: Inner ve Left Join

Inner Join



Sadece eşleşen kayıtlar.
(Kesişim Kümesi)

Left Join



Tüm sol tablo korunur.
Eşleşmeyenler NaN olur.

Kaostan Düzene: Sürecin Özeti

"Ham veri gürültüdür; işlenmiş veri bilgidir."