

# Haftalık Ödev – Yaz Kampı: Veri Bilimi – Hafta 3

---

## Hedefler:

- İleri seviye veri yapılarıyla (liste, sözlük, küme) rahatça çalışabilmesi,
  - Fonksiyonlar, lambda ve gömülü fonksiyonları etkin kullanabilmesi,
  - Modüller ve temel istatistiksel hesaplamaları uygulayabilmesi,
  - Kodlarını düzenli ve okunabilir şekilde yazabilmesi,
  - Basit bir veri analizi sürecini baştan sona gerçekleştirebilmesi beklenmektedir.
- 

## Soru 1 – Liste Metotları

Bir sınıfta öğrencilerin notları şu şekilde tutuluyor:

**notlar = [85, 92, 76, 92, 100, 76, 85, 92]**

- Listedeki tekrar eden notları silip **benzersiz** bir liste oluşturun.
  - En yüksek ve en düşük notu bulun.
  - Notları küçükten büyüğe sıralayın.
- 

## Soru 2 – Sayılar

Bir sayının **Armstrong sayısı** olup olmadığını kontrol eden bir Python fonksiyonu yazın.

**Armstrong sayısı:** Her basamağının küplerinin toplamı kendisine eşit olan sayılar.

Örn:  $153 \rightarrow 1^3 + 5^3 + 3^3 = 153$

---

## Soru 3 – Kümeler

Aşağıdaki iki küme verilmiştir:

$A = \{\text{"Python"}, \text{"R"}, \text{"SQL"}, \text{"Java"}\}$

$B = \{\text{"C++"}, \text{"Python"}, \text{"JavaScript"}, \text{"SQL"}\}$

- Ortak dilleri bulun.
  - Sadece A'da olan dilleri listeleyin.
  - İki kümenin birleşimini alfabetik olarak yazdırın.
- 

## Soru 4 – Modüller

- random modülünü kullanarak 1–100 arasında **10 rastgele sayı** üretin.
- Bu sayıların **ortalamasını** ve **standart sapmasını** statistics modülü ile hesaplayın.

---

### Soru 5 – Fonksiyonlar

kelime\_sayacı(metin) adında bir fonksiyon yazın.  
Fonksiyon verilen metindeki:

- toplam kelime sayısını
- en uzun kelimeyi
- en sık geçen kelimeyi döndürsün.

---

### Soru 6 – Gömülü Fonksiyonlar

Aşağıdaki liste için map, filter, sorted gibi gömülü fonksiyonları kullanarak:

sayılar = [5, 12, 7, 18, 24, 3, 16]

- Sadece çift sayıları filtreleyin.
- Bu sayıların karelerini bulun.
- Karelerini **azalan sırada** sıralayın.

---

### Soru 7 – Lambda İfadeleri

Aşağıdaki listeyi, her kelimenin uzunluğuna göre küçükten büyüğe sıralayın.

kelimeler = ["veri", "bilim", "analiz", "yapayzeka", "python"]

Bunu sorted + lambda ile yapın.

---

### Soru 8 – Metodlar

Bir string içinde geçen **tüm rakamları** bulun ve bunların toplamını döndüren bir fonksiyon yazın.

Örn: "abc12def3" → 12 + 3 = 15

---

### Soru 9 – (Ekstra) Numpy 1

10 elemanlı bir numpy dizisi oluşturun.

- Elemanlar 0–50 arasında rastgele sayılar olsun.
  - Dizinin ortalamasını, standart sapmasını ve en büyük değerini bulun.
-

## Soru 10 – (Ekstra) Numpy 2

5x5 boyutunda rastgele 0–1 arasında değerlerden oluşan bir numpy matrisi üretin.

- Her sütunun ortalamasını bulun.
- 0.5'ten büyük olan değerleri 1, küçük eşit olanları 0 yaparak **binary matris** oluşturun.

---

## Proje – “Kitap Satış Analiz Sistemi”

Bir yayınevinin farklı türlerde ve yazarlarda kitapları var. Örnek veri:

```
kitaplar = [  
    {"isim": "Veri Bilimi 101", "yazar": "Ali", "tur": "Bilim", "satis": 1200, "yil": 2021},  
    {"isim": "Python ile Yapay Zeka", "yazar": "Ayşe", "tur": "Bilim", "satis": 950, "yil":  
2020},  
    {"isim": "İstatistik Temelleri", "yazar": "Ali", "tur": "Akademik", "satis": 700, "yil": 2019},  
    {"isim": "Makine Öğrenmesi", "yazar": "Can", "tur": "Bilim", "satis": 1800, "yil": 2022},  
    {"isim": "Veri Görselleştirme", "yazar": "Deniz", "tur": "Sanat", "satis": 400, "yil": 2018},  
    {"isim": "Matematiksel Modelleme", "yazar": "Ali", "tur": "Akademik", "satis": 1500,  
"yil": 2021},  
    {"isim": "Bilgi Toplumu", "yazar": "Ayşe", "tur": "Sosyal", "satis": 600, "yil": 2022}  
]
```

### Yapılması Gerekenler:

#### 1. Fonksiyon Yazma:

- **en\_cok\_satan(kitaplar)** → En çok satan kitabın bilgilerini döndürsün.
- **yazar\_satislari(kitaplar)** → Her yazarın toplam satışını bir sözlük olarak döndürsün.

#### 2. Liste ve Küme İşlemleri:

- Tüm kitap türlerini (**tur**) **küme** halinde çıkarın (tekrar eden türler olmadan).
- Satış adedi **1000'den fazla olan kitapların isimlerini** bir listede toplayın.

#### 3. Lambda / Filter / Map Kullanımı:

- filter ile **2020'den sonra çıkan kitapları** süzün.
- map ile tüm satış adetlerini %10 artırılmış şekilde yeni bir listeye aktarın.
- sorted + lambda ile kitapları satış miktarına göre azalan şekilde sıralayın.

#### 4. İstatistiksel Analiz:

- Ortalama satış adedini bulun.
- En çok satış yapan türü bulun.
- Satışların **standart sapmasını** hesaplamak için statistics modülünü kullanın.

#### 5. Ekstra (Zorlayıcı): Train/Test Simülasyonu

- Kitap listesini rastgele **%70 eğitim (train)**, **%30 test** verisine ayırın (random.sample).
- Eğitim verisinden yazarların ortalama satışını hesaplayın.
- Test verisinde, hangi kitapların satışlarının bu ortalamanın üzerinde olduğunu kontrol edin.

---

#### Beklenen Çıktı Örneği

- En çok satan kitap: "**Makine Öğrenmesi**"
- Yazar satışları: {"Ali": 3400, "Ayşe": 1550, "Can": 1800, "Deniz": 400}
- Türler: {"Bilim", "Akademik", "Sanat", "Sosyal"}
- 1000'den fazla satan kitaplar: ["Veri Bilimi 101", "Makine Öğrenmesi", "Matematiksel Modelleme"]
- Ortalama satış: 1021.4
- Standart sapma: ~480.2

- |  |
|--|
| • Eğitim/ Test ayırımı sonrası analiz: |
|--|

Analizde Sizden Beklenen:
---------------------------

- |  |
|--|
| <ul style="list-style-type: none"><li>• Veriyi uygun oranlarda ayırın (örneğin %70 train, %30 test).</li><li>• Train seti üzerinde basit analizler yapın (ortalama, sıklık, vs.).</li><li>• Test seti üzerinde aynı analizleri tekrarlayın.</li><li>• Sonuçları karşılaştırın ve kısa yorum ekleyin.</li></ul> |
|--|

---

#### Teslim Şekli ve Kurallar:

- Tüm sorular tek bir .ipynb (Jupyter Notebook) dosyasında çözülecek.
- Proje ödevi ayrı bir python projesi olarak çözülecek.
- Tüm ödev tek bir repository de toplanacak ve githuba public olarak yüklenilecek.
- Github bağlantısı eğitmenlere istenilen şekilde iletilecek.