



# Yazılım ve Bilişim Topluluğu 5

## Haftalık ‘Python’ Kampı

### 4. Hafta – Listeler ve Koleksiyonlar

Bu Hafta Neden Önemli?

Gerçek dünyada veriler tek başına değil, gruplar halinde karşımıza çıkar. Örneğin bir alışveriş listesi, öğrencilerin notları veya kullanıcıların giriş bilgileri... İşte bu tür durumları yönetebilmek için **koleksiyon yapıları** kullanılır.

Bu hafta en sık kullanılan veri yapısı olan **listeler** ile başlayacağız. Ardından tuple, dictionary ve set gibi farklı koleksiyonları öğreneceğiz. Ayrıca Python'a özgü güçlü araçlar olan **list comprehension**, **enumerate**, **zip** ve **variable unpacking** konularını işleyerek çok daha kısa, okunabilir ve “Pythonic” kod yazmayı öğreneceğiz.

Kısacası, bu hafta:

- Verileri **düzenli saklamayı**,
- Onlar üzerinde **işlemler yapmayı**,
- Ve daha **etkili yazılım geliştirmeyi** öğreneceğiz.

Bu Haftanın Konuları

- Listeler ve Liste Metotları**
  - append(), remove(), insert(), sort(), reverse()
  - İç içe listeler
- Tuple (Demetler)**
  - Değiştirilemeyen koleksiyon yapısı
  - Kullanım alanları
- Dictionary (Sözlükler)**
  - Anahtar – değer ilişkisi
  - Veri tabanı mantığına giriş
- Set (Kümeler)**
  - Benzersiz elemanlar
  - Küme işlemleri: kesişim, birleşim, fark



## 1. List Comprehension

- Tek satırda liste oluşturma
- Koşullu comprehension

## 2. Enumerate ve Zip Kullanımı

- Listeler üzerinde numaralandırma
- Birden fazla listeyi aynı anda döngüye sokma

## 3. Variable Unpacking

- Çoklu atamalar ( $a, b, c = [1, 2, 3]$ )

# Bu Haftanın Ödevleri

## 1. Alışveriş Listesi

Bir alışveriş listesi oluştur, kullanıcıdan ürün ekle/çıkar, güncel listeyi yazdır.

Örnek Girdi/Çıktı:

```
Liste: ['Ekmek', 'Süt']
Yeni ürün ekleyin: Peynir
Güncel liste: ['Ekmek', 'Süt', 'Peynir']
```

## 2. Öğrenci Notları

Bir sözlükte öğrencilerin isimlerini ve notlarını sakla, ortalamayı hesapla.

Örnek Çıktı:

```
Öğrenciler: {'Ahmet': 80, 'Mehmet': 70, 'Ali': 90}
Ortalama: 80.0
```

## 3. Küme İşlemleri

İki küme oluştur, ortak elemanları ve farklılıklarını bul.

Örnek Çıktı:

```
A = {1, 2, 3, 4}
B = {3, 4, 5, 6}

Kesişim: {3, 4}
Birleşim: {1, 2, 3, 4, 5, 6}
Fark (A-B): {1, 2}
```



## 4. List Comprehension

1'den 20'ye kadar olan çift sayıları listeleyiniz.

Örnek Çıktı:

```
[2, 4, 6, 8, 10, 12, 14, 16, 18, 20]
```

## 5. Enumerate Kullanımı

Bir film listesi oluşturun, enumerate ile sıra numarası vererek yazdırınız.

Örnek Çıktı:

```
1. Matrix  
2. Inception  
3. Interstellar
```

## 6. Variable Unpacking

Bir tuple'daki değerleri değişkenlere ayırınız.

Örnek Çıktı:

```
person = ("Burak", 25, "İstanbul")  
Ad: Burak  
Yaş: 25  
Şehir: İstanbul
```

## 7. Sözlük ile Kelime Sayacı

Kullanıcıdan bir metin al, her kelimenin kaç kez geçtiğini bir **dictionary** kullanarak sayınız.

Örnek Girdi/Çıktı:

```
Bir metin girin: python öğrenmek çok güzel ve python çok eğlenceli  
{'python': 2, 'öğrenmek': 1, 'çok': 2, 'güzel': 1, 've': 1, 'eğlenceli': 1}
```

Ekstra:

- Büyük/küçük harf farkını ortadan kaldırın (ör. "Python" = "python").
- Kullanıcı sadece boşluk değil noktalama işaretleri de girerse, temizleyip sayınız.



## 🤝 Hafta Sonu Grup Etkinliği – Mini Proje

### Proje: Telefon Rehberi Uygulaması

Bu etkinlikte, dictionary kullanarak basit bir telefon rehberi yapacağız.

- Kullanıcı yeni kişi ekleyebilecek,
- Var olan kişiyi arayabilecek,
- İstediğinde kişi silebilecek.

Örnek Girdi/Çıktı:

```
1. Kişi Ekle  
2. Kişi Ara  
3. Kişi Sil  
Seçiminiz: 1  
İsim: Mehmet  
Numara: 5551234567  
Kişi eklendi!  
  
Rehber: {'Mehmet': '5551234567'}
```

Bu proje sayesinde dictionary mantığını, veri ekleme/silme işlemlerini ve pratik menü tasarımını öğreneceğiz.

## 🤝 Hafta Sonu Grup Etkinliği – Mini Proje 2

### Günün Özeti Programı – Koleksiyonlarla Geliştirilmiş Sürüm

Bu proje, **3. haftada** yaptığımız temel “Günün Özeti Programı”nın geliştirilmiş hâlidir.

Bu haftaki amaç, **koleksiyon veri yapıları** (listeler, sözlükler vb.) kullanarak programı daha profesyonel hale getirmektir.

### Proje Amacı

- Her gün girilen bilgileri (uyku süresi, çalışma süresi vb.) **sözlük** yapısında tutmak
- Tüm günleri **bir liste** içinde saklamak
- **Ortalama uyku ve çalışma sürelerini** hesaplamak
- **Enumerate** kullanarak günleri numaralandırmak
- **List comprehension** ile kısa, temiz ve tekrarlı işlemleri yapmak



## Programın Çalışma Mantığı

- + Kullanıcıdan **adı, yaşı, uyandığı saat, uykusu süresi ve çalışma süresi** alınır.
- + Uyku süresi ve çalışma süresine göre **boş zaman** hesaplanır.
- + Bu bilgiler bir **dictionary** (gun\_ozeti) içinde saklanır.
- + Bu sözlük, genel **listeye** (gunler) eklenir.
- + İstenirse başka günler eklenebilir.
- + Tüm veriler işlendiğinde, program:
  - Ortalama uykusu süresini,
  - Ortalama çalışma süresini hesaplar.
- + Her günün özeti numaralandırılarak ekrana yazdırılır.
- + Her gün için 2 defa **motivasyon mesajı** basılır (list comprehension ile).

## Örnek Girdi:

```
Adınız: Aslı
Yaşınız: 20
Uyandığınız saat: 07:00
Uyku süresi (saat): 6
Çalışma süresi (saat): 8
Yeni gün eklemek ister misin? (e/h): e
```

```
Adınız: Mete
Yaşınız: 21
Uyandığınız saat: 08:30
Uyku süresi (saat): 7
Çalışma süresi (saat): 6
Yeni gün eklemek ister misin? (e/h): h
```



## Örnek Çıktı:

--- Gün Özeti ---

Ad: Aslı

Yas: 20

Uyandigi\_saat: 07:00

Uyku\_suresi: 6

Calisma\_suresi: 8

Bos\_zaman: 10

👉 Kodlama bir gelişim yolculuğudur!

👉 Kodlama bir gelişim yolculuğudur!

--- Gün Özeti ---

Ad: Mete

Yas: 21

Uyandigi\_saat: 08:30

Uyku\_suresi: 7

Calisma\_suresi: 6

Bos\_zaman: 11

👉 Kodlama bir gelişim yolculuğudur!

👉 Kodlama bir gelişim yolculuğudur!

📊 Haftalık Özeti:

Ortalama uyku süresi: 6.5

Ortalama çalışma süresi: 7.0

📅 Gün Gün Özeti:

1. gün - Aslı (6 saat uyku, 8 saat çalışma)

2. gün - Mete (7 saat uyku, 6 saat çalışma)

## Kazanımlar

- Artık kullanıcı verilerini **tek değil, gruplar halinde** saklıyoruz.
- Program artık bir “**veri tabanı mantığı**”na sahip.
- **Koleksiyon yapılarını** (liste, sözlük, comprehension, enumerate) gerçek bir senaryoda uyguladık.



## 🎬 Film Önerisi – *Megamind* (2010)

Süper zekâlı bir kötü karakter olan Megamind'ın hikâyesini anlatan bu film, aslında zekânın ve stratejinin gücünü vurgular. Megamind, en büyük düşmanını yendikten sonra amacını kaybeder; fakat yeni bir yol bulmak için zekâsını, yaratıcılığını ve öğrenme isteğini kullanır.



## 📖 Kitap Önerisi (Modern Klasik) – *Fahrenheit 451* – Ray Bradbury

Kitapların yasaklandığı, insanların ekranlarla uyuşturulduğu bir distopyayı anlatan bu eser, bilgiye erişimin ne kadar önemli olduğunu gösterir. Kodlama öğrenirken bilgiye ulaşma özgürlüğünün değerini daha iyi anlayacaksınız.

