



# Python

---

## 1. Başlangıç Seviyesi

### Temel Kavramlar

- Veri tipi dönüşümleri
- String işlemleri ve metotları
- Input/output işlemleri (print, input)

### Kontrol Yapıları

- if-elif-else koşul ifadeleri
- for ve while döngüleri
- break, continue ve pass ifadeleri
- Basit algoritmalar kurma

### Veri Yapıları

- Listeler ve liste metotları
- Demetler (tuples)
- Sözlükler (dictionaries)
- Kümeler (sets)

### Fonksiyonlar

- Fonksiyon tanımlama ve çağırma
- Parametreler ve return ifadeleri
- Değişken kapsamı (scope)

- Lambda fonksiyonları

## Uygulama Projeleri

- **Akıllı Alışveriş Listesi:** Kullanıcının alışveriş ürünlerini kategorilere ayıran, fiyat takibi yapan ve bütçe analizi sunan bir uygulama. Dosya işlemleriyle verileri kaydedecek

## 2. Orta Seviye

### Nesne Yönelimli Programlama (OOP)

- Sınıflar ve nesneler
- Miras (inheritance)
- Kapsülleme (encapsulation)
- Çok biçimlilik (polymorphism)
- Soyut sınıflar ve metotlar

### Modüller ve Paketler

- Standart kütüphane modülleri (os, sys, datetime, math)
- Modül oluşturma ve içe aktarma
- pip ile paket yönetimi
- Virtual environment kullanımı

### Dosya İşlemleri

- Dosya okuma ve yazma
- CSV, JSON işlemleri
- Dizin işlemleri

### Hata Yönetimi

- try, except, finally blokları
- Özel hata oluşturma
- Debugging teknikleri

## Uygulama Projeleri

- **Kişisel Finans Yöneticisi:** OOP kullanarak gelir-gider takibi yapan, grafik raporlar oluşturan ve tasarruf önerileri sunan bir sistem. Verileri JSON formatında saklayan ve veri analizi sunabilecek modüler yapı

## 3. İleri Seviye

### İleri Python Özellikleri

- Dekoratörler
- Jeneratörler ve iteratörler
- Context managers
- Regular expressions
- Eşzamanlılık (threading, multiprocessing)
- Asenkron programlama (asyncio)

### Veritabanı Programlama

- SQLite, MySQL veya PostgreSQL ile çalışma
- ORM kullanımı (SQLAlchemy)
- Veritabanı tasarımı ve optimizasyon

### Test ve Kalite Güvencesi

- Unit testing (unittest, pytest)
- TDD (Test Driven Development)
- Kod kalite araçları (pylint, flake8)
- Dökümantasyon (docstrings, Sphinx)

## Uygulama Projeleri

- **Veri Analizi Otomasyonu:** Farklı kaynaklardan (CSV, web API) veri çeken, temizleyen, analiz eden ve otomatik raporlar oluşturan kapsamlı bir sistem. Asenkron işlemler ve veri görselleştirme içerecek. ML algoritmalarıyla basit tahminler de yapabilir

## 4. Uzmanlaşma Yolları

### Temel Seviye

#### Veri İşleme ve Analiz

- Pandas temel işlemler (veri yükleme, temizleme, filtreleme)
- **Polars** giriş - yüksek performanslı veri işleme
- NumPy ile bilimsel hesaplama
- **DuckDB** ile SQL sorguları ve Pandas entegrasyonu

#### Görselleştirme Temelleri

- Matplotlib ve Seaborn kullanımı
- **PyGWalker** ile sürükle-bırak analizler
- **Lux** ile otomatik veri keşfi

#### İstatistik ve ML Temelleri

- Temel istatistik kavramları
- Scikit-learn giriş (regresyon, sınıflandırma)
- Model değerlendirme metrikleri
- **PyCaret** ile düşük kodlu ML

### Proje

**Veri Keşif Otomasyonu:** Herhangi bir veri setini analiz eden, otomatik istatistikler ve grafikler üreten, temel ilişkileri tespit eden bir araç

### Orta Seviye

#### İleri Veri İşleme

- **Vaex** ile büyük veri setleri işleme
- **Modin** ile çok çekirdekli veri işleme

- Veri özellik mühendisliği
- Zaman serisi veri işleme (**sktime**)

## Kapsamlı ML

- Hiperparametre optimizasyonu (**Optuna**)
- Ensemble yöntemler (XGBoost, LightGBM, CatBoost)
- **AutoGluon** ile otomatik model seçimi
- Model açıklanabilirliği (SHAP, LIME)

## Deployment ve MLOps Temelleri

- Streamlit ve Gradio ile uygulama geliştirme
- **Kedro** ile veri işlem pipeline'ları
- **MLflow** ile model yönetimi
- FastAPI ile model servisleştirme

## Proje

**Tahmin Modellemesi Platformu:** Kullanıcıların veri yükleyebildiği, otomatik model oluşturabildiği, performans karşılaştırması yapabildiği ve modeli API olarak dışa açabildiği bir web uygulaması

## | İleri Seviye

## Derin Öğrenme

- PyTorch temelleri
- **PyTorch Lightning** ile ölçeklenebilir DL
- CNN, RNN, Transformers
- **fastai** uygulamaları

## NLP ve LLM

- Hugging Face ekosistemi

- **Sentence-Transformers** ile metin gömme
- **LangChain** ve **LlamaIndex** ile LLM uygulamaları
- **PEFT/LoRA** ile model ince ayarı
- **vLLM** ve **Ollama** ile verimli LLM dağıtımı

## Görsel Veri İşleme

- OpenCV ve bilgisayarlı görü
- **Ultralytics (YOLO)** ile nesne tespiti
- **Albumentations** ile görüntü veri artırımı

## MLOps Gelişmiş

- **ZenML** ile ML pipeline'ları
- **Evidently AI** ile model izleme
- **DVC** ile veri ve model versiyonlama
- **Metaflow** ile production iş akışları

## Proje

**Çok Modlu AI Asistanı:** Metin, görüntü ve yapılandırılmış veri işleyebilen, RAG (Retrieval Augmented Generation) kullanarak özel veri kaynakları üzerinde çalışabilen interaktif bir asistan

## Uzmanlaşma Alanları

### Ölçeklenebilir Veri İşleme

- Apache Spark ve PySpark
- **RAPIDS** ile GPU hızlandırılmalı veri bilimi
- Dask ile paralel hesaplama

### GenAI

- Diffusion modelleri ve görüntü üretimi
- Çok modlu modeller (görüntü-metin)

- **LMQL** ile yapılandırılmış LLM programlama

## Takviyeli Öğrenme

- RL temelleri
- Stable Baselines3 ve acme kütüphaneleri

## İleri MLOps

- Kubeflow ile ML iş akışları
- Model sunumu optimizasyonu
- **BentoML** ile model paketleme

## Projeler

1. **Kişiselleştirilmiş RAG Sistemi:** Özel veri kaynakları üzerinde çalışan, kullanıcı davranışına adapte olan bilgi erişim sistemi
2. **GenAI İçerik Stüdyosu:** Metin, görüntü ve ses üreten çok modlu bir içerik üretim platformu
3. **Zaman Serisi Tahmin Sistemi:** Gerçek zamanlı veri akışlarını işleyen, çoklu kaynaklardan beslenen tahmin sistemi
4. **MLOps Orkestrasyon Platformu:** Modelleri üretimde izleyen, otomatik geri eğitim tetikleyen ve model yönetimi sağlayan platform

Her aşamanın sonunda modern teknolojileri kullanarak pratik projeler geliştireceğiz ve gerçek dünya problemlerine çözümler üreteceğiz.