## Python ile Kapsamlı Bir Proje: Kişisel Bilgi Yönetimi Uygulaması

**Proje Amacı:**

Bu proje, Python'daki temel kavramları (fonksiyonlar, sınıflar, JSON işlemleri) bir araya getirerek kişisel bilgileri yöneten basit bir uygulama oluşturmayı amaçlamaktadır. Bu uygulama sayesinde kullanıcılar, kişisel bilgilerini JSON formatında saklayabilir, bu bilgilere erişebilir ve güncelleyebilirler.

**Proje Özellikleri:**

* **Kişisel Bilgi Sınıfı:** Kişinin adı, soyadı, doğum tarihi, telefon numarası gibi bilgileri tutan bir sınıf oluşturulacaktır.
* **JSON İşlemleri:** Kullanıcı bilgileri, JSON formatında bir dosyada saklanacak ve bu dosya gerektiğinde okunarak güncellenecektir.
* **Fonksiyonellik:**
  + **Kişi Ekleme:** Yeni bir kişi eklemek için gerekli bilgilerin kullanıcıdan alınması ve JSON dosyasına eklenmesi.
  + **Kişi Listeleme:** Tüm kayıtlı kişilerin bilgilerinin ekranda listelenmesi.
  + **Kişi Arama:** Ad veya soyadı gibi belirli bir kritere göre kişi arama ve bulunan kişilerin bilgilerinin gösterilmesi.
  + **Kişi Güncelleme:** Var olan bir kişinin bilgilerinin güncellenmesi.
  + **Kişi Silme:** Belirli bir kişinin kaydının silinmesi.
* **Dinamik Kod:** Kullanıcı etkileşimine göre farklı işlemler yapacak şekilde kodun yazılması.

**Kod Örneği:**

Python

import json

class Kisi:

def \_\_init\_\_(self, ad, soyad, dogum\_tarihi, telefon):

self.ad = ad

self.soyad = soyad

self.dogum\_tarihi = dogum\_tarihi

self.telefon = telefon

def kisi\_ekle(kisiler):

ad = input("Adı: ")

soyad = input("Soyadı: ")

dogum\_tarihi = input("Doğum Tarihi: ")

telefon = input("Telefon Numarası: ")

yeni\_kisi = Kisi(ad, soyad, dogum\_tarihi, telefon)

kisiler.append(yeni\_kisi)

json.dump(kisiler, open("kisiler.json", "w"))

def kisi\_listele(kisiler):

for kisi in kisiler:

print(f"Adı: {kisi.ad}, Soyadı: {kisi.soyad}, Doğum Tarihi: {kisi.dogum\_tarihi}, Telefon: {kisi.telefon}")

# ... diğer fonksiyonlar (kisi\_ara, kisi\_guncelle, kisi\_sil)

try:

with open("kisiler.json", "r") as dosya:

kisiler = json.load(dosya)

except FileNotFoundError:

kisiler = []

while True:

print("1. Kişi Ekle")

print("2. Kişi Listele")

# ... diğer seçenekler

secim = input("Seçiminiz: ")

if secim == "1":

kisi\_ekle(kisiler)

elif secim == "2":

kisi\_listele(kisiler)

# ... diğer seçenekler

Kodu [dikkatli](/faq" \l "coding" \t "_blank) kullanın.

**Proje Geliştirme Önerileri:**

* **Hata Yönetimi:** Kullanıcı hatalarına karşı (örneğin, boş girişler, geçersiz veri formatları) daha sağlam bir hata yönetimi sistemi ekleyebilirsiniz.
* **Kullanıcı Arayüzü:** Basit bir komut satırı arayüzü yerine, daha kullanıcı dostu bir arayüz (örneğin, Tkinter, PyQt) kullanarak uygulamayı geliştirebilirsiniz.
* **Veritabanı Entegrasyonu:** Küçük çaplı bir proje için JSON yeterli olsa da, daha büyük projelerde veritabanı (SQLite, PostgreSQL) kullanmak daha uygun olabilir.
* **Ek Özellikler:** Arama kriterlerini geliştirme, istatistiksel analizler yapma, farklı dosya formatlarını destekleme gibi ek özellikler ekleyebilirsiniz.

**Bu proje, Python'daki temel kavramları öğrenmek ve uygulamak için harika bir başlangıç noktasıdır. Kendi fikirlerinizi ekleyerek ve projeyi daha da geliştirerek Python becerilerinizi artırabilirsiniz.**

**Not:** Bu sadece bir örnek proje olup, kendi ihtiyaçlarınıza ve ilgi alanlarınıza göre özelleştirebilirsiniz.

**Yardımcı Olabilecek Kaynaklar:**

* **Python Resmi Belgeleri:** [https://docs.python.org/tr/3/](https://www.google.com/url?sa=E&source=gmail&q=https://docs.python.org/tr/3/)
* **Yazbel Python Eğitimleri:** [https://python-istihza.yazbel.com/](https://python-istihza.yazbel.com/" \t "_blank)
* **YouTube Eğitim Videoları:** Python ile ilgili birçok ücretsiz eğitim videosu bulunmaktadır.

**Başarılar dilerim!**

**Ek Sorularınız Olursa Çekinmeden Sorabilirsiniz.**

**Bu projede hangi konuda daha detaylı bilgi almak istersiniz?** (örneğin: sınıfların daha derinlemesine incelenmesi, JSON işlemleri, hata yönetimi, kullanıcı arayüzü)



yukarida belirttigin projeyi bana adim adim yapmam gerekenleri gorev olarak yaz. ben de bu gorevleri takip ederek projeyi tamamalamaya calisayim

Taslakları göster

## Python Projesi: Kişisel Bilgi Yönetimi Uygulaması İçin Adım Adım Görevler

Harika bir başlangıç! Şimdi, projeyi adım adım tamamlaman için sana bir yol haritası sunacağım.

### 1. ****Projenin Temelini Kurma****

* **Yeni bir Python dosyası oluştur:** kisisel\_bilgiler.py gibi anlamlı bir isim ver.
* **Gerekli kütüphaneyi ekle:** json kütüphanesi, veriyi JSON formatında işlemek için gerekli.

Python

import json

Kodu [dikkatli](/faq#coding) kullanın.

### 2. ****Kişi Sınıfını Tanımlama****

* **Sınıf oluştur:** Kisi adında bir sınıf oluşturarak kişinin özelliklerini (ad, soyad, doğum tarihi, telefon) temsil eden değişkenler tanımla.
* **init metodu:** Sınıfın bir örneği oluşturulduğunda bu metod otomatik olarak çağrılarak nesneye başlangıç değerleri atanır.

Python

class Kisi:

def \_\_init\_\_(self, ad, soyad, dogum\_tarihi, telefon):

self.ad = ad

self.soyad = soyad

self.dogum\_tarihi = dogum\_tarihi

self.telefon = telefon

Kodu [dikkatli](/faq#coding) kullanın.

### 3. ****JSON Dosyasını İşleme****

* **Dosya oluşturma:** İlk çalıştırmada kisiler.json adında bir dosya oluştur ve boş bir liste olarak kaydet. Bu listeye daha sonra kişiler eklenecek.
* **Veri okuma:** Program başlangıcında kisiler.json dosyasını oku ve verileri bir listeye yükle.
* **Veri yazma:** Her değişiklikten sonra listeyi tekrar dosyaya yaz.

Python

try:

with open("kisiler.json", "r") as dosya:

kisiler = json.load(dosya)

except FileNotFoundError:

kisiler = []

Kodu [dikkatli](/faq#coding) kullanın.

### 4. ****Fonksiyonları Tanımlama****

* **Kişi ekleme:** kisi\_ekle fonksiyonu, kullanıcıdan bilgi alarak yeni bir Kisi nesnesi oluşturur ve kisiler listesine ekler. Sonrasında güncellenen listeyi dosyaya yazar.
* **Kişi listeleme:** kisi\_listele fonksiyonu, kisiler listesindeki tüm kişilerin bilgilerini ekranda gösterir.
* **Kişi arama:** kisi\_ara fonksiyonu, kullanıcıdan bir arama kriteri alarak listede arama yapar ve bulunan kişileri gösterir.
* **Kişi güncelleme:** kisi\_guncelle fonksiyonu, kullanıcıdan güncellenecek kişinin indeksini ve yeni bilgilerini alarak listeyi günceller.
* **Kişi silme:** kisi\_sil fonksiyonu, kullanıcıdan silinecek kişinin indeksini alarak listedeki ilgili kişiyi siler.

Python

def kisi\_ekle(kisiler):

# ... (Yukarıdaki kodda olduğu gibi)

def kisi\_listele(kisiler):

# ... (Yukarıdaki kodda olduğu gibi)

# ... diğer fonksiyonlar (kisi\_ara, kisi\_guncelle, kisi\_sil)

Kodu [dikkatli](/faq" \l "coding" \t "_blank) kullanın.

### 5. ****Kullanıcı Arayüzü Oluşturma****

* **While döngüsü:** Programı sürekli çalıştıran bir while döngüsü oluştur.
* **Menü:** Kullanıcıya seçenekler sunan basit bir menü oluştur.
* **Seçeneklere göre işlem:** Kullanıcının seçtiği seçeneklere göre ilgili fonksiyonları çağır.

Python

while True:

print("1. Kişi Ekle")

print("2. Kişi Listele")

# ... diğer seçenekler

secim = input("Seçiminiz: ")

if secim == "1":

kisi\_ekle(kisiler)

elif secim == "2":

kisi\_listele(kisiler)

# ... diğer seçenekler

Kodu [dikkatli](/faq#coding) kullanın.

### 6. ****Test Etme ve Geliştirme****

* **Farklı senaryoları dene:** Programı çalıştırıp farklı işlemleri yaparak hataları tespit et ve düzelt.
* **Ek özellikler ekle:** Arama kriterlerini geliştir, hata yönetimi ekle, kullanıcı arayüzünü iyileştir.

**Önemli Notlar:**

* **Hata Yönetimi:** Kullanıcı hatalarına karşı (örneğin, yanlış veri girişi) try-except bloklar kullanarak hata yönetimi ekle.
* **Kullanıcı Arayüzü:** Daha gelişmiş bir kullanıcı arayüzü için Tkinter, PyQt gibi kütüphaneleri kullanabilirsin.
* **Veritabanı Entegrasyonu:** Daha büyük projelerde SQLite gibi bir veritabanı kullanmak daha uygun olabilir.

**Unutma:** Bu sadece bir başlangıç noktası. Projeni kendi ihtiyaçlarına ve ilgi alanlarına göre özelleştirebilirsin.

**Hangi adımda takılırsan veya daha fazla bilgiye ihtiyacın olursa çekinmeden sorabilirsin.**

**Ek olarak, şu konularda daha detaylı bilgi almak ister misin?**

* Sınıfların daha derinlemesine incelenmesi
* JSON işlemleri
* Hata yönetimi
* Kullanıcı arayüzü
* Veritabanı entegrasyonu

Bu görevleri takip ederek adım adım ilerleyebilir ve kendi kişisel bilgi yönetimi uygulamanı oluşturabilirsin. Başarılar!