



ADI: MEHMET

SOYADI: YILDIRIM

NUMARA: 171419006

**VERİTABANI YÖNETİM SİSTEMLERİ
PROJE RAPORU**

Projem JPG uzantılı dosyaları PDF' e çeviren bir converter yapmaktı. Bu projeyi gerçekleştiren Visual Studio Code IDE' sini kullandım. Programı geliştirirken 6 adet classla çalıştım. Bu classlar: **login**, **sign_up**, **main**, **converter**, **log** ve **database** classlarıdır. Bu program bir kullanıcının kayıt, ücretli üyelik, kullanıcı bilgileri, pdf converter ve geçmişte oluşturduğu pdfleri tekrar listeleme ve açma fonksiyonlarını gerçekleştirmektedir. Program tasarımında ara yüz tasarımı için **Python**'un **TKINTER** eklentisi kullanılmıştır.

Veritabanı çalışmaları için MYSQL kullanılmıştır. Bu verilerin depolanması için pdf_converter şeması oluşturulmuştur. Bu şema altında 4 adet tablo bulunmaktadır. Bunlar: kullanıcı_giris, bilgiler, ucretli_uyelik ve pdfler tablolarıdır. Bu tablolara ait detaylı bilgiler classların anlatımından sonra anlatılacaktır.

1. LOGIN:

Login ekranında mail ve şifre alanları için iki adet textbox ve giriş yapmak ve kayıt olmak için iki ayrı buton bulunuyor. Sign Up butonu sign_up ekranını tetiklerken, login butonu veritabanından kullanıcı bilgilerini kontrol ederek main ekranını tetiklemektedir.

Login Class'ında Kullanılan Fonksiyonlar ve Görevleri:

- login(): Kullanıcı adı ve şifre alanlarını kontrol edip database classından login_database fonksiyonuna gönderir. Eğer kullanıcı adı ve şifre eşleşirse sisteme girişi sağlar.
- sign_up(): Sign Up butonuna tıklandığında sign_up classını tetikleyip kayıt ekranını açmayı sağlayan fonksiyondur.

2. SIGN_UP:

Sign_up ekranında kullanıcı adı, e-mail ve password alanları için üç adet textbox ve premium ve normal üyelik için iki adet radiobutton bulunmaktadır. Bunlara ek olarak veritabanına bilgilerin kaydı ve giriş ekranına dönmeyi sağlamak için iki adet buton bulunmaktadır. Bu alanda normal üyelik ve premium üyelik için iki farklı form bulunmaktadır. Bu formları radiobuttonlar tetikler. Bu formun görevi sisteme yeni üye kaydı yapmak ve bilgilerini almaktır.

Normal üye kaydı formu:

Premium üye kaydı formu:

Sign_up Class'ında Kullanılan Fonksiyonlar ve Görevleri:

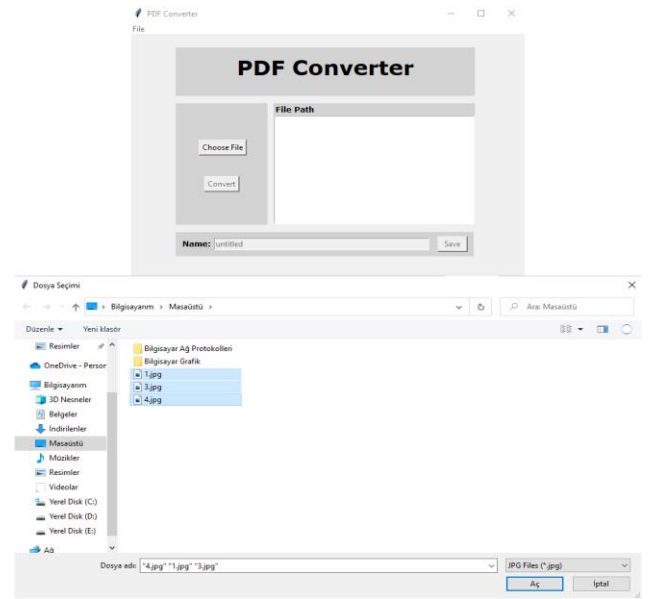
- register(): Seçilen üyelik türünü kontrol edip bilgileri veritabanına kaydeden fonksiyondur. Register butonu tarafından tetiklenir.
- back(): Login ekranına dönüşü sağlayan fonksiyondur. Back butonu tarafından tetiklenir.
- premium(): Premium üye kaydı formunun görünümünü etkinleştirir.
- no_premium(): Normal üye kaydı formunun görünümünü etkinleştirir.

3. MAIN:

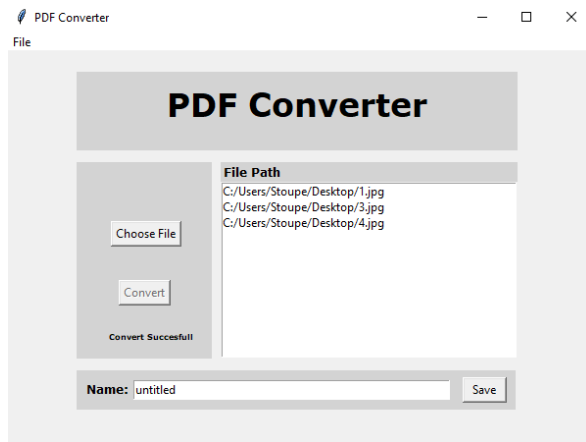
Main ekranında dosya seçmek ve seçilen dosyaları pdf'e convert etmek için iki adet buton ve seçilen dosyaların eklendiği bir listbox bulunur. Dosya seçmeden convert butonu etkinleşmez. Dosya convert edilmeden dosya isimlendirme textbox'u ve save butonu etkinleşmez. Bu class'ın görevi JPG uzantılı dosyaları listelemek, PDF'e convert etmek ve kaydetmektir. Üstte bulunan menü barda File sekmesi bulunmaktadır. O sekmedeki Data Log butonuna tıklandığında log formu tetiklenmektedir.

Main Formuna ait görseller:

Choose File Butonu Dosya Seçme:



Convert butonu ile JPG'lerin PDF'e çevrilmesi:



Dosyanın kaydedileceği isim ve dosya kaydı:



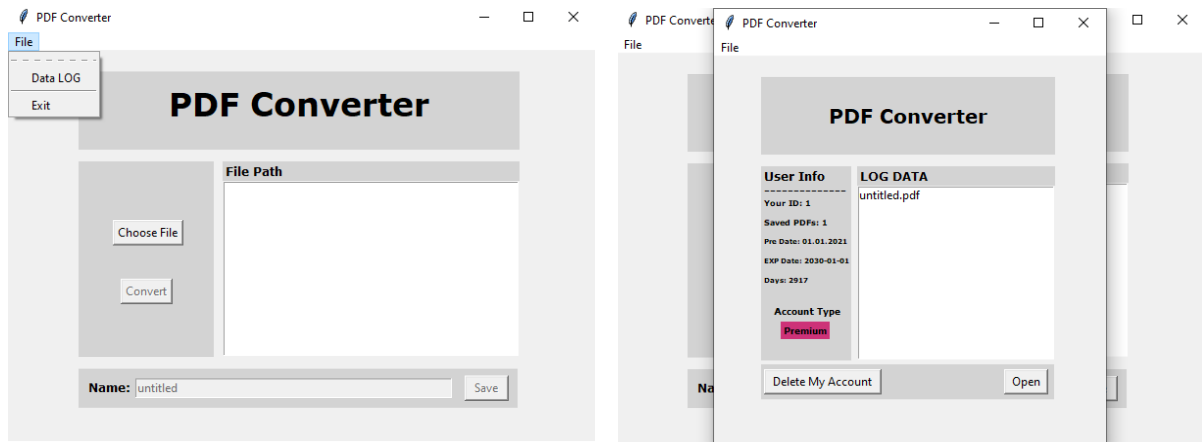
MAIN Class'ında Kullanılan Fonksiyonlar ve Görevleri:

- openFile(): Dosya seçme ve seçilen dosyaların listbox'a eklenmesini sağlayan fonksiyon.
- convertFile(): JPG dosyalarını PDF'e çevirecek convert fonksiyonuna ileten fonksiyon.
- savePath(): PDF'in kaydedileceği dosya yolunu kaydeden fonksiyon.
- data_LOG(): log ekranını tetikleyen fonksiyon

4. LOG:

Main ekranındaki File menüsünde Data LOG butonuna basınca tetiklenir. Bu formda kullanıcının kullanıcı_ID'si, üyelik tarihi, üyelik bitiş tarihi, üyelik türü ve kalan premium günlerine dair bilgiler bulunur. Bu ekranda bir adet listbox ve open butonu bulunmaktadır. Veritabanına kaydedilen pdfler listelenir. Listedenden kayıt seçilir ve open butonuna basıldığında o veri veritabanından çekilir ve pdf'e çevrilerek açılır. Delete My Account butonu ise kullanıcının tüm bilgilerini veritabanından siler(bilgileri depolamak amaçlı kalıcı olarak silinen veriler tablolarına kaydırır).

LOG Formuna ait görseller:

**LOG Class'ında Kullanılan Fonksiyonlar ve Görevleri:**

- list_print(): O anki kullanıcının daha önce veritabanına kaydettiği pdf'leri ve kullanıcı bilgilerini listeler.
- openPdf(): Listedenden seçilen pdf'i veritabanından çeken ve açan fonksiyondur.
- delete_Account(): Mevcut Kullanıcının bilgilerini silen ve silinen bilgileri koruyucu veritabanında saklayan fonksiyondur.

5. CONVERTER:

Main ekranında convert butonuna basıldığında çalışan class'tır. Bu classta gönderilen JPG uzantılı dosyalar. PDF'e dönüştürülür. Bu dönüştürme işlemi yapılırken PIL kütüphanesinden faydalanılmıştır. Bu fonksiyonda PDF'e convert gerçekleştirildikten sonra veritabanına kayıt eden fonksiyonlar tetiklenmektedir. Bu formun bir ara yüzü bulunmamaktadır.

CONVERTER Class'ında Kullanılan Fonksiyonlar ve Görevleri:

- convert_images(): Main formundan gönderilen JPG dosyalarını PDF'e çeviren fonksiyondur.
- save_pdf(): PDF dosyasını bilgisayarın *masaüstüne* kullanıcıdan alınan isimle kaydeden fonksiyondur.

6. DATABASE:

Programda kullanılan veritabanlarıyla ilgili tüm işlemlerin gerçekleştirildiği class'tır. Bu classta kullanılan fonksiyonlar pdf_converter veritabanına bilgileri kaydeder, düzenler ve listeler. Yapılan tüm veritabanı işlemleri MYSQL ile gerçekleştirilmiştir. Bu formun bir ara yüzü yoktur.

DATABASE Class'ında Kullanılan Fonksiyonlar ve Görevleri:

- a. login_database(): Login ekranında girilen kullanıcı bilgilerini alan ve veritabanında bulunup bulunmadığını kontrol eden fonksiyondur.
- b. convertToBinary(): Veritabanında PDF uzantılı dosya kaydetmek mümkün değildir. PDF ilk önce BLOB Data'ya çevrilir, sonra veritabanına kaydedilir. Bu fonksiyon PDF'i BLOB Dataya çevirir.
- c. convertBinaryToFile(): Veritabanında kayıtlı olan BLOB Data'yı PDF'e çevirir.
- d. pdfsaveDatabase(): PDF'e ait kullanıcı bilgilerini ve BLOB Data'yı veritabanına kaydeden fonksiyondur.
- e. ReConvert(): convertBinaryToFile fonksiyonunu kullanarak veritabanında kayıtlı olan pdf'leri açmayı sağlayan fonksiyon.
- f. list_pdf(): Mevcut kullanıcının veritabanına kaydettiği PDF'leri listeleyen fonksiyon.
- g. account_type(): Mevcut kullanıcının üyelik durumunu(normal,premium) listeleyen fonksiyon.
- h. normal_account_register(): Normal üyelik bilgilerini veritabanına kaydeden fonksiyondur.
- i. premium_account_register():Premium üyelik bilgilerini veritabanın kaydeden fonksiyondur.
- j. get_kullanici_id(): Mevcut kullanıcının kullanıcı_ID' sini listeleyen fonksiyondur.
- k. bilgiler_kayit(): Kullanıcı bilgilerini veritabanına kaydeden fonksiyondur.
- l. premium_date(): Premium kullanıcıların premium bilgilerini(Son premium tarihi vb.) listeleyen fonksiyondur.
- m. delete_acc(): Kullanıcı bilgilerini silmek için kullanılan fonksiyondur. Kullanıcının sisteme giriş bilgilerini veritabanından siler.
- n. delete_info(): Kullanıcı bilgilerini silmek için kullanılan fonksiyondur. Kullanıcının sistemde kayıtlı bilgilerini veritabanından siler.
- o. Delete_pdf(): Kullanıcı bilgilerini silmek için kullanılan fonksiyondur. Kullanıcının sistemde kayıtlı pdf bilgilerini veritabanından siler.
- p. trigger(): Kullanıcının sisteme giriş bilgileri silinmek istendiğinde tetiklenir. Kullanıcının tüm bilgilerini veritabanından silecek sql sorgularını tetikler.

7. VERİTABANI İŞLEMLERİ:

Veritabanı çalışmaları için MYSQL kullanılmıştır. Bu verilerin depolanması için pdf_converter şeması oluşturulmuştur. Bu şema altında 6 adet tablo bulunmaktadır. Bunlar: kullanıcı_giris, bilgiler, ucretli_uyelik ve pdfler tablolarıdır.

1. kullanıcı_giris: Bu tabloda kullanıcı_ID, kullanıcı_mail ve kullanıcı_sifre kolonları bulunmaktadır.
 - a. kullanıcı_ID (PK, NN, AU) olup INT türündedir. Kullanıcı ID sini tutar.
 - b. kullanıcı_mail (NN) olup VARCHAR türündedir. Kullanıcıya ait maili tutar.
 - c. kullanıcı_sifre (NN) olup VARCHAR türündedir. Kullanıcıya ait şifreyi tutar.
2. bilgiler: Bu tabloda kullanıcı_ID, kullanıcı_isim_soyisim, kullanıcı_mail ve kullanıcı_uyelik_duzeyi kolonları bulunmaktadır.
 - a. kullanıcı_ID (FK, NN) olup INT türündedir. Kullanıcı ID sini tutar.
 - b. kullanıcı_isim_soyisim (NN) olup VARCHAR türündedir. Kullanıcıya ait isim bilgisini tutar.
 - c. kullanıcı_mail (NN) olup VARCHAR türündedir. Kullanıcıya ait mail bilgisini tutar.
 - d. kullanıcı_uyelik_duzeyi (NN) olup VARCHAR türündedir. Kullanıcının üyelik düzeyi bilgisini tutar.
3. ucretli_uyelik: Bu tabloda iducretli_uyelik, kullanıcı_ID, uyelik_baslangic_tarihi, uyelik_bitis_tarihi, kredi_karti_no, cvv, son_kullanim_tarihi kolonları bulunmaktadır.
 - a. iducretli_uyelik (PK, NN, AU) olup INT türündedir. Ücretli üyelik ID'sini tutar.
 - b. kullanıcı_ID (FK, NN) olup INT türündedir. Kullanıcı ID sini tutar.
 - c. uyelik_baslangic_tarihi (NN) olup VARCHAR türündedir. Premium üyelik başlangıç tarihini tutar.
 - d. uyelik_bitis_tarihi (NN) olup VARCHAR türündedir. Premium üyelik bitiş tarihini tutar.
 - e. kredi_karti_no (NN) olup INT türündedir. Kullanıcının kredi kartı numarasını tutar.
 - f. cvv (NN) olup INT türündedir. Kullanıcının kredi kartına ait CVV numarasını tutar.
 - g. son_kullanim_tarihi (NN) olup INT türündedir. Kullanıcının kredi kartına ait son kullanım tarihini tutar.

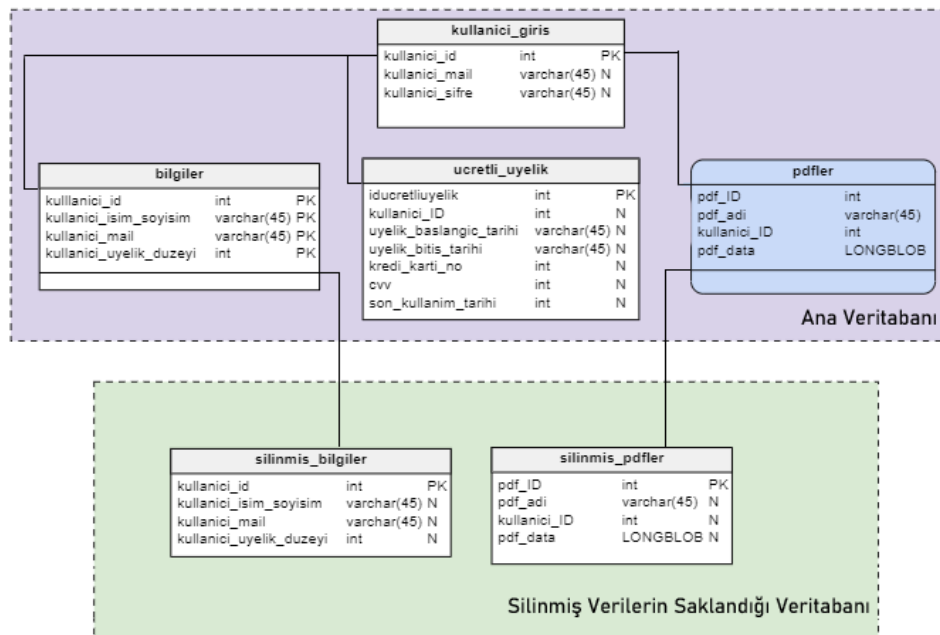
4. pdfler: Bu tabloda pdf_ID, pdf_adi, kullanıcı_ID ve pdf_data kolonları bulunmaktadır.
 - a. pdf_ID (PK, NN, AU) olup INT türündedir. Pdf'e ait pdf_ID'yi tutar.
 - b. pdf_adi (NN) olup VARCHAR türündedir. Pdf'e ait pdf ismini tutar.
 - c. kullanıcı_ID (FK) olup INT türündedir. Kullanıcıya ait kullanıcı ID'sini tutar.
 - d. pdf_data (NN) olup LONGBLOB türündedir. Pdf'yi BLOB türünde depolar.
5. silinmis_bilgiler: Bu tabloda kullanıcı_ID, kullanıcı_isim_soyisim, kullanıcı_mail ve kullanıcı_uyelik_duzeyi kolonları bulunmaktadır.
 - e. kullanıcı_ID (FK, NN) olup INT türündedir. Kullanıcı ID sini tutar.
 - f. kullanıcı_isim_soyisim (NN) olup VARCHAR türündedir. Kullanıcıya ait isim bilgisini tutar.
 - g. kullanıcı_mail (NN) olup VARCHAR türündedir. Kullanıcıya ait mail bilgisini tutar.
 - h. kullanıcı_uyelik_duzeyi (NN) olup VARCHAR türündedir. Kullanıcının üyelik düzeyi bilgisini tutar.
6. silinmis_pdfler: Bu tabloda pdf_ID, pdf_adi, kullanıcı_ID ve pdf_data kolonları bulunmaktadır.
 - e. pdf_ID (PK, NN, AU) olup INT türündedir. Pdf'e ait pdf_ID'yi tutar.
 - f. pdf_adi (NN) olup VARCHAR türündedir. Pdf'e ait pdf ismini tutar.
 - g. kullanıcı_ID (FK) olup INT türündedir. Kullanıcıya ait kullanıcı ID'sini tutar.
 - h. pdf_data (NN) olup LONGBLOB türündedir. Pdf'yi BLOB türünde depolar.

5. ve 6. Tablolar veritabanından üyeliğini silen kullanıcının bilgilerini daha sonra oluşabilecek durumlar için saklamaktadır. Bu tablolara bilgi eklenmesi kullanıcı bilgileri tablosundan veri silinmesiyle "trigger"ıyla gerçekleştirilir

VERİTABANINA AİT VARLIK İLİŞKİ ŞEMASI



VERİTABANINA AİT DİYAGRAM



8. VERİTABANI SORGULARI ÇIKTILARI: Çıktılar mevcut kodların çalıştırılmasıyla sağlanmıştır.

1. **Örnek Join işlemi:** *Kullanici_giris* tablosundan aldığı kullanıcı_ID' lere göre *pdfler* tablosunda kaydedilmiş pdf dosya isimlerini ve blob datalarını listeler.

```
SELECT 1_kullanici_giris.kullanici_ID, pdf_adi, pdf_data
FROM 1_kullanici_giris
INNER JOIN pdfler ON pdfler.pdf_ID
```

	kullanici_ID	pdf_adi	pdf_data
▶	1	untitled.pdf	BLOB
	37	untitled.pdf	BLOB
	36	untitled.pdf	BLOB
	1	untitled.pdf	BLOB
	37	untitled.pdf	BLOB
	36	untitled.pdf	BLOB
	1	untitled.pdf	BLOB
	37	untitled.pdf	BLOB
	36	untitled.pdf	BLOB

2. **Örnek Fonksiyon İşlemi:** Fonksiyona verilen kullanıcı_ID' sine göre kullanıcının kaç pdf kaydettiğini döndüren fonksiyon.

```
CREATE FUNCTION kayitsayi(@gelensayi int)
RETURNS int
AS
BEGIN
DECLARE @sayi int
SELECT @sayi=count(pdf_adi) FROM pdfler WHERE kullanici_ID=@gelensayi
RETURN @sayi
END
```

	count(pdf_adi)
▶	2

3. **Örnek Trigger İşlemi:** Kullanıcı hesabını silmek istediğinde(bu işlemi onayladığında) gerekli verileri silinmiş veritabanına taşınır ve kullanıcı bilgileri silinir.

```
DELIMITER //
CREATE TRIGGER hesap_silindi BEFORE DELETE
ON 1_kullanici_giris
FOR EACH ROW
BEGIN
INSERT INTO z_silinmis_pdfler(pdf_ID,pdf_adi,kullanici_ID,pdf_data) values
(pdf_ID,pdf_adi,kullanici_ID,pdf_data);
INSERT INTO z_silinmis_bilgiler(kullanici_ID,kullanici_isim_soyisim,kullanici_mail,kullanici_uyelik_duzeyi) value
(kullanici_ID,kullanici_isim_soyisim,kullanici_mail,kullanici_uyelik_duzeyi);
END//
DELIMITER ;
```

Trigger çalıştırılması sonucu silinmiş verilerin taşındığı tablolardaki veriler. (Trigger database.py dosyasından sağlanmaktadır.)

	pdf_ID	pdf_adi	kullanici_ID	pdf_data
▶	19	Bleedat.pdf	41	BLOB
*	NULL	NULL	NULL	NULL

	kullanici_ID	kullanici_isim_soyisim	kullanici_mail	kullanici_uyelik_duzeyi
▶	41	deneme	deneme	0
*	NULL	NULL	NULL	NULL