**PACKAGE MODEL CLASS’LARI**

***EVRAK SINIFI***

Burda numara, ad, tur, tarih, sahip, kategori, aciklama, adres değişkenleri ile Evrak constactoru oluşturuldu bütün değişkenler private olarak tanımlandı , getter setter oluşturuldu, listView için to string metotu overidde edildi .

***EVRAKLAR SINIFI***

Bu sınıf, evrak nesnelerini içeren bir "**ObservableList**" nesnesine sahiptir. Ayrıca, bu sınıf "**DosyaIslem**" adında bir arayüzü **implemente** eder. Bu arayüz, dosyalar ile ilgili olarak okuma ve yazma işlemleri yapabilecek iki metod içerir.

Bu sınıf içerisinde, evrak nesnelerini ekleme işlemini gerçekleştiren "**EvrakEkle**" adlı bir metod ve evrak dosyasını açma işlemini gerçekleştiren "**DosyaAc**" adlı bir metod bulunur. "**EvrakEkle**" metodu, evrak nesnesi oluşturmak için gerekli olan tüm özellikleri argüman olarak alır ve bu özellikleri kullanarak bir evrak nesnesi oluşturur. Bu evrak nesnesi, daha sonra "**evraks**" adlı ObservableList nesnesine eklenir. "**DosyaAc**" metodu ise, verilen dosya adresiyle birlikte bu dosyayı açmayı sağlar.

Bu sınıfın içerisinde ayrıca, "DosyaOkuma" ve "DosyaYazma" adlı iki metod bulunur. Bu metodlar, arayüz tarafından tanımlanan metodları implemente eder ve dosya ile ilgili okuma ve yazma işlemlerini gerçekleştirir. "**DosyaOkuma**" metodu, verilen dosya adresinde bulunan dosyayı okur ve dosyadaki verileri "evraks" adlı ObservableList nesnesine ekler. "**DosyaYazma**" metodu ise, "evraks" adlı ObservableList nesnesindeki verileri verilen dosya adresinde bulunan dosyaya yazar.

Bu sınıf ayrıca, "**Initializable**" arayüzünü de implemente eder. Bu sınıfın içerisinde, "initialize" metodu içerisinde "DosyaOkuma" metodu çağrılır.

***KULLANICILAR SINIFI***

Bu kod, bir Java sınıfı olan "Kullanıcılar" için bir tanımlama sağlar. Sınıf, "**DosyaIslem**" interface’ini uygular ve bir "**ObservableList**" oluşturur. "ObservableList" **arr** adında bir değişkene atanır ve bu değişken, "Kullanıcı" sınıfından oluşan nesneler için bir liste oluşturur.

Sınıf, aşağıdaki yöntemleri içerir:

* "**KullanıcıEkle**(): Yeni bir "Kullanıcı" nesnesi oluşturur ve "arr" listesine ekler.
* "**DogruKullanici():** Giriş için kullanıcı adı ve şifre kontrolü yapar. Eğer doğru bir kullanıcı adı ve şifre girilmişse, "dogruKullancıİndex" değişkenine doğru kullanıcının indeksini atar ve "true" değerini döndürür. Aksi halde, "false" değerini döndürür. Bu yöntem ayrıca "DosyaOkuma**()** yöntemini çağırır, böylece kullanıcı bilgileri dosyadan okunabilir.
* "**DosyaOkuma()"**Kullanıcılar.txt" adlı bir dosyadan verileri okur ve "KullanıcıEkle" yöntemini çağırarak verileri "arr" listesine ekler.
* "**DosyaYazma():** "Kullanıcılar.txt" adlı bir dosyaya "arr" listesindeki verileri yazar. Eğer dosya mevcut değilse, oluşturulur.

***DOSYA ISLEM SINIFI (interface)***

Bu sınıfda iki metod bulunur DosyaOkuma(), DosyaYazma() bu metodlar ile veri kaybı önlenir. Bu interface sınıf evraklar ve kullanıcılar sınıfına implements edilir.

**PACKAGE APPLİCATİON CLASSLARI**

***SAMPLE CONTROLLER SINIFI***

Sample.fxml dosyasına bağlıdır . Bu JavaFX uygulaması, kullanıcının bir giriş ekranındaki mail adresi ve **şifresini doğrulayarak**, giriş yaptığında ikinci bir ekrana geçiş yapmasını sağlar. Uygulama ayrıca, **kaydol butonu** ile kullanıcının sisteme kaydolabileceği bir ekranı da içerir. Bu ekran sayesinde, sisteme daha önceden kayıt olunmamış olan kullanıcılar da kaydolabilir.

***CHİLD CONTROLLER SINIFI***

Bu sınıf, **Initializable** interfaceini implements ediyor ve **initialize()** metodunu override ediyor. Bu metod, arayüz nesnelerinin ilk yapılandırılması için kullanılır. Bu sınıf, ayrıca Menu sınıfından türetilmiş ve bu nedenle *anaEkran(), girisEkran(), useradd() ve evrakekle()* metodlarını da içerir. Bu metotlar butonlara bağlıdır ve diğer ekranlara geçmek için kolaylık sağlar. **TableView** nesnesi, veritabanından okunan evrak bilgilerini göstermek için kullanılır. tarih ve tarihgLabel nesnelerine bugünün tarihi atanır. **AcButon()** tableViewden seçilen evrakların açılmasını sağlar. **SilButon()** tableViewden seçilen evrakların silinmesini sağlar. gisimLabel, gnumaraLabel ile sisteme giren kullanıcının bilgilerini ekrana yansıtırız. Ayrıca **DosyaOkuma** ve **DosyaYazma** işlemleri için kullanıcılar ve evraklar sınıfından nesne oluşturur.

***CHİLD1 CONTROLLER SINIFI***

Bu sınıf, **Initializable** interfaceini implements ediyor ve **initialize()** metodunu override ediyor. Bu metod, arayüz nesnelerinin ilk yapılandırılması için kullanılır. Bu sınıf, ayrıca Menu sınıfından türetilmiş ve bu nedenle *anaEkran(), girisEkran(), useradd() ve evrakekle()* metodlarını da içerir. Burda bulunan **radio button** ve **textlabeller** ile kullanıcı bilgileri alınır ve anında kullanıcılar sınıfında olan **kullanıcı ekle()** metodu çağrılarak **ObservableList’e** atanır . ObservableList **listView** e atanır. **silButon()** metodu ile listViewden seçilen kullanıcı listeden silinir. **temizleButon()** metodu ile label ve radio butonlar temizlenir.

***CHİLD2 CONTROLLER SINIFI***

Bu sınıf, **Initializable** interfaceini implements ediyor ve **initialize()** metodunu override ediyor. Bu metod, arayüz nesnelerinin ilk yapılandırılması için kullanılır. Bu sınıf, ayrıca Menu sınıfından türetilmiş ve bu nedenle *anaEkran(), girisEkran(), useradd() ve evrakekle()* metodlarını da içerir. **SecButon()** kullanıcının bilgisayarından bir dosya seçmesine olanak sağlar. Dosya seçildikten sonra dosyanın adresi bir text field'e yazdırılır. **EvrakEkleButon()** metodu ile **texfield** lerden alınan stringler evraklar sınıfından çağrılan **EvrakEkle()** metoduna yazılır. **temizleButon()** metodu ile label temizlenir. Eklenen evrakları hemen görmek için evraklar anında **listView’**e yansıtılır.

***CHİLD3 CONTROLLER SINIFI***

Bu sınıfın amacı sisteme giremeyen kullanıcıların kaydını almaktır. bir kullanıcının isim, numara, şifre ve e-posta bilgilerini almak için dört adet **TextField** bulunmaktadır. Ayrıca, bu sınıf içerisinde bir Kullanıcılar nesnesi de oluşturulmuştur. **EkleButon** metodu, kullanıcı tarafından girilen bilgileri alır ve Kullanıcılar nesnesinin **KullanıcıEkle** metodunu çağırarak yeni bir kullanıcı ekler. Bu metod sonrasında, kullanıcı bilgilerinin dosyaya yazılması için **DosyaYazma** metodu çağırılır. Bu metodun sonrasında, userEkleButton'ın yazısı değiştirilerek kullanıcıya bilgi verilir ve TextField'lar temizlenir. emizleButon metodu ise, TextField'ları temizlemek için kullanılır. **initialize** metodu ise, başlangıçta Kullanıcılar nesnesinin **DosyaOkuma** metodunu çağırarak kullanıcı bilgilerini dosyadan okur.

***MENU ABSTRACT SINIFI***

**anaEkran(), girisEkran(), useradd() ve evrakekle()** metodlarını içeren bu sınıf controller sınıfında gereksiz tekrarı önler. Ekran değişimine olanak sağlar.

***MAİN SINIFI***

Tek yaptığı sample.fxml’i açıp arayüzün ilk penceresini açar.