LABORATUVAR OTOMASYONU

Main.java :   
A computer screen with text

Description automatically generated

A screen shot of a computer

Description automatically generatedMain pagede sadece uygulama açıldığında ilk olarak loginpage nesnesi olusturulur ve loginpage sayfasi calistirilip ekrana verilir.  
  
loginpage.java :

A screen shot of a computer program

Description automatically generatedA screen shot of a computer

Description automatically generated  
loginpage.java classında ilk önce sayfada olacak olan butonlar, textfıeldlar, arraylist gibi şeyler tanımlanır. Daha sonra bu sayfayı oluşturmak için gereken constructor oluşturulur. Constructor içinde sırasıyla şu işlemler yapılır. İlk önce bütün buton, textfield gibi şeylerin bulunacağı pencerenin boyutu, title’I oluşturulur. Sonra bu sayfanın boyutlandırılma seçeneği kapatılır. Sonra uygulama çarpıdan kapatıldığında arkada çalışmayı bırakıp tamamen kapanması için CloseOperation seçilir. Sonra sayfada bulunan oğelerin yerleştirileceği bir panel oluşturulur. Daha sonra bu oğelerin yerleri ve boyutları setBound() ile ayarlanır. Ardından sayfada bulunan butonlara tıklandığında oluşacak işlemler tanımlanır.  
  
 Quit Button kısmında kullanıcı uygulamadan çıkış yapmak istediğinde ilk önce bir messagebox oluşturur ve kullanıcıya son bir kez çıkmak isteyip istemediğini sorar. Kullanıcı eğer evete basarsa uygulmadan çıkış yapılır. Hayıra basarsa program bir şey olmadan loginpagede çalışmaya devam eder.  
  
 Submit button kısmında ise kullanıcının giriş yapmak için sonraki ekrana geçmesini sağlar. Ancak bundan önce ilk önce users dosyasından satır satır okuma yaparak bu satırların her birini “/” işaretlerinden bölerek bir diziye atar ve bu dizileri bir arraylistte saklar. Uygulamada admin ve kullanıcı olarak iki farklı kullanıcı türü vardır. Kullanıcı eğer giriş yaparken kullanıcı adı ve şifre kısımlarına admin yazarak giriş yapmaya çalışırsa kullanıcıyı admin paneline atar. Eğer admin girişi yapmazsa sonraki sayfaya geçişine izin vermeden önce daha önceden arrayliste atadığımız userlardan kullanıcı bilgilerinin geçerli olup olmadığını control eder. Eğer geçerli bilgiler girilmişse bir message box atarak kullanıcıya giriş yaptığı geri bildirimini vermek için “Welcome kişiismi” olacak şekilde bir dönüş verir ve login page kısmını kapatarak kullanıcının ne görmek isteyeceğini seçeceği seçim sayfasına geçiş yapar. Eğer girilen bilgiler geçerli kullanıcılar arasında bulunamadıysa. Kullanıcıya geçersiz giriş yaptığını söylemek için tekrardan messagebox ile geri bildirim verir. Bu yapılan işlemlerde oluşacak olası hataları yakalamak için bunları try catch blokları içinde yapar. Eğer dosya bulunamıyor filenotfound hatasında bulunan messageboxı gösterir. Eğer uygulamaya okuma işlemi yapılamıyorsa IO Exception hatası verir.  
  
 Butonlara komutlar verildikten sonra bulunan textfieldların manuel olarak yerleştirilebilmesi için setLayout(null) kullanılır. Sonrasında uygulama açıldığında sayfanın ortasında gözükmesi için setLocationRelativeTo(null) fonksiyonu kullanılır. Daha sonra bu sayfadaki bulunan tüm oğeler gösterilmek için sayfaya başta eklediğimiz panele eklenir. Sonra bu panel de sayfaya eklenir. En sonda da setVisible() fonksiyonuyla sayfanın gösterilmesi sağlanır.

A screenshot of a computer program

Description automatically generatedA computer screen shot of a program code

Description automatically generatedChoosePage.java :

ChoosePage kullanıcı sayfası açıldığında her şeyin tek sayfada olup karışmaması için oluşturuldu. Bu sayfada önce bulunacak öğeler oluşturulur. Sonra constructor oluşturulur. Constructor içerisinde ilk önce choosepage penceresinin boyutları ayarlanır. Sonra çarpıdan kapatıldığında tamamen kapanması için CloseOperation seçilir. Pencere titleı oluşturulur. Öğelerin bulunacağı panel oluşturulur. Öğelerin konumları ve boyutları setBounds() ile ayarlanır ve sonrasında öğeler panele eklenir. Ondan sonra kullanıcıya gideceği sayfayı seçmesi için oluşturulan butonlara komutlar atanır.   
  
 Show Chemicals butonuna basılırsa ChoosePage kapatılır ve ChemPage’in nesnesi oluşturulur ve ChemPage açılır.   
  
 Show Equipments butonuna basılırsa ChoosePage kapatılır ve EquipmentPagein nesnesi oluşturularak Equipmentpage açılır.  
  
 Sign out butonuna basarsa choosepage kapatılarak loginpage açılır ve hesaptan çıkış yapılır.  
  
Butonlara komutlar atandıktan sonra loginpagedeki gibi ortada constructor oluşması için setLocationRelativeTo(null) kullanılır. Sonra öğelerin bulunduğu panel pencereye eklenir ve setVisible ile pencere görünür kılınır.

A screen shot of a computer

Description automatically generatedA screen shot of a computer screen

Description automatically generatedChemsPage.java :   
  
  
A screen shot of a computer

Description automatically generated  
A screen shot of a computer

Description automatically generated  
A screenshot of a computer program

Description automatically generated  
A computer screen shot of a program code

Description automatically generated  
  
  
Interfaceden kullanılacak fonksiyonlar implement edilir. Öncelikle pencerede kullanılacak öğeler tanımlanır. Constructor oluşturulur. Constructor içerisinde açılacak olan pencerenin ilk önce boyutları ayarlanır. Sonra boyutlanıdırılabilirliği ayarlanır. Çarpıdan kapatıldığında olacak olan CloseOperation seçilir. Sonra öğelerin koyulacağı panel oluşturulur. Sayfa açıldığında sayfadaki listeye yazılacak olan bilgilerin okunması için önce Inheritance classından bir nesne oluşturulur ve bu classtan gerekli reader fonksiyonu çağrılır. Inheritance classından önce overload ile dosya okuma fonksiyonu çağrılır. Sonra override edilerek Inheritance classından arrayliste ekleme yapmak için diğer reader fonksiyonu çağrılır. Bu fonksiyonlarda try-catch yapısıyla IOException kontrolü yapılır ve hata durumunda mesaj verilir. Daha sonrasında fonksiyonlarla arrayliste eklenen bilgiler okunarak JListin bilgi aldığı DefaultListModele atama yapılır ve Jliste dokunan bilgiler eklenir. Sonra sayfada bulunan öğelerin konumları ve boyutları setBounds ile ayarlanır. Sonra sayfada bulunan butonlara komutlar eklenir.  
  
 Back butonu kullanıcının choosepage’e geri dönmesini sağlar tıklandığında ChemsPage’I kapatarak ChoosePage’I açar.  
  
 Remove butonu listede tıklanan öğenin kaldırılmasını sağlar. getSelectedIndex ile selectedIndex değişkenine bir değer atanır. Seçildiğinde bu değişkenin değeri -1 olur. Sonra bu durum kontrol edilir eğer bu değişkenin değeri butona basıldığında -1 ise arraylistten ve listeden bu seçilen kimyasalın bilgileri kaldırılır. Sonra try-catch içerisinde bu bilgilerin tutulduğu dosya güncellenir. IO Exception durumunda hata yakalanır ve geri bildirim verilir. Eğer değer -1 değilse yani bir şey seçilmemişse silme yapılamayacağından kullanıcıya silmek için öğe seçmesi gerektiğine dair geri bildirim verir.  
  
 Add butonu kullanıcının textfieldlara girdiği bilgilere göre listeye ekleme yapmasını sağlar. Hata yakalama işleminin yapılabilmesi için try-catch blokları kullanılır. Önce yazılacak dosya açılır ve bir yazıcı oluşturulur. Sonra try içerisinde girilen bilgilerin geçerli olup olmadığı kontrol edilir. Catch bloklarında ilk önce dosyanın açılıp açılmadığı kontrol edilir. Sonra yapılan girişlerin doğru türde olup olmadığı kontrol edilir. Yani örneğin sayı girilmesi gereken yerlere metin, metin girilmesi gereken yerlere sayı girilip girilmediğini kontrol eder. Sonra tür kontrol edildikten sonra geçerli olup olmadığı kontrol edilir. Bu kontroller yapıldıktan sonra girilen değer doğru türde de olsa geçerli olup olmadığını kontrol eder. Örneğin gün girişinde 1-30 arasında olmadığını kontrol eder. Girilen kimyasalın zaten listede bulunup bulmunmadığını kontrol eder. Sonucunda duruma göre hata mesajı verir. En sonda IO Exception durumu kontrol edilir. En sonda giriş yapıldığında textfieldlar boşaltılır.  
  
 Sonrasında sayfadaki öğeler panele eklenir. Sayfanın konumu ayarlanır panel sayfaya eklenir ve sayfa açılır. Fonksiyonların amacı textfieldlardaki girişlerin geçerli olup olmadığını kontrol eder.  
  
  
  
EquipmentPage.java :   
A screen shot of a computer

Description automatically generated  
A computer screen shot of a computer code

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A black background with text

Description automatically generated  
Interfaceden kullanılacak fonksiyonlar implement edilir. Öncelikle pencerede kullanılacak öğeler tanımlanır. Constructor oluşturulur. Constructor içerisinde açılacak olan pencerenin ilk önce boyutları ayarlanır. Sonra boyutlanıdırılabilirliği ayarlanır. Çarpıdan kapatıldığında olacak olan CloseOperation seçilir. Sonra öğelerin koyulacağı panel oluşturulur. Sayfada bulunan öğelerin konumları ve boyutları setBounds ile ayarlanır. Try-catch içerisinde önce okunacak dosya açılır. Bir reader oluşturulur ve dosya satır satır okunarak satırlar “/” ile bölünerek diziye atanır ve diziler de arrayliste atanır. Try-catch içerisinde dosya işlemlerinde oluşabilecek hatalar yakalanır. Hata durumunda geri bildirim verilir. Sonra arrayliste atanan dizilerdeki bilgiler ekipman listesine yazdırılır. Sonra sayfadaki butonlara komutlar atanır.  
  
 Back butonu kullanıcının choosepage’e geri dönmesini sağlar tıklandığında ChemsPage’I kapatarak ChoosePage’I açar.  
  
 Remove butonu listede tıklanan öğenin kaldırılmasını sağlar. getSelectedIndex ile selectedIndex değişkenine bir değer atanır. Seçildiğinde bu değişkenin değeri -1 olur. Sonra bu durum kontrol edilir eğer bu değişkenin değeri butona basıldığında -1 ise arraylistten ve listeden bu seçilen ekipmanın bilgileri kaldırılır. Sonra try-catch içerisinde bu bilgilerin tutulduğu dosya güncellenir. IO Exception durumunda hata yakalanır ve geri bildirim verilir. Eğer değer -1 değilse yani bir şey seçilmemişse silme yapılamayacağından kullanıcıya silmek için öğe seçmesi gerektiğine dair geri bildirim verir.  
  
 Add butonu kullanıcının textfieldlara girdiği bilgilere göre listeye ekleme yapmasını sağlar. Hata yakalama işleminin yapılabilmesi için try-catch blokları kullanılır. Önce yazılacak dosya açılır ve bir yazıcı oluşturulur. Sonra try içerisinde girilen bilgilerin geçerli olup olmadığı kontrol edilir. Catch bloklarında ilk önce dosyanın açılıp açılmadığı kontrol edilir. Sonra yapılan girişlerin doğru türde olup olmadığı kontrol edilir. Yani örneğin sayı girilmesi gereken yerlere metin, metin girilmesi gereken yerlere sayı girilip girilmediğini kontrol eder. En sonda IO Exception durumu kontrol edilir. En sonda giriş yapıldığında textfieldlar boşaltılır.  
  
   
 Sonrasında sayfadaki öğeler panele eklenir. Sayfanın konumu ayarlanır panel sayfaya eklenir ve sayfa açılır. Fonksiyonların amacı textfieldlardaki girişlerin geçerli olup olmadığını kontrol eder.  
  
adminframe.java :   
A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screen shot of a computer program

Description automatically generated A screenshot of a computer program

Description automatically generated  
Once pencerede bulunacak öğeler oluşturulur. Sonra constructor oluşturulur. Constructor içerisinde önce açılan pencerenin boyutu, title’ı gibi özellikleri ayarlanır. Öğelerin üstüne yerleştirileceği panel oluşturulur ve background rengi ayarlanır. Sonra öğelerin boyutları ve konumları setBounds ile ayarlanır. Try-catch içerisinde önce okunacak dosya açılır. Bir reader oluşturulur ve dosya satır satır okunarak satırlar “/” ile bölünerek diziye atanır ve diziler de arrayliste atanır. Try-catch içerisinde dosya işlemlerinde oluşabilecek hatalar yakalanır. Hata durumunda geri bildirim verilir. Sonra arrayliste atanan dizilerdeki bilgiler kullanıcı listesine yazdırılır. Sonra sayfadaki butonlara komutlar atanır.   
  
 Sign out butonu kullanıcının çıkış yapmasını sağlar. Tıklandığında admin page’i kapatır ve loginpage sayfasını açar.  
  
 Remove butonu listede tıklanan öğenin kaldırılmasını sağlar. getSelectedIndex ile selectedIndex değişkenine bir değer atanır. Seçildiğinde bu değişkenin değeri -1 olur. Sonra bu durum kontrol edilir eğer bu değişkenin değeri butona basıldığında -1 ise arraylistten ve listeden bu seçilen kullanıcı bilgileri kaldırılır. Sonra try-catch içerisinde bu bilgilerin tutulduğu dosya güncellenir. IO Exception durumunda hata yakalanır ve geri bildirim verilir. Eğer değer -1 değilse yani bir şey seçilmemişse silme yapılamayacağından kullanıcıya silmek için öğe seçmesi gerektiğine dair geri bildirim verir.  
  
 Add butonu kullanıcının textfieldlara girdiği bilgilere göre listeye ekleme yapmasını sağlar. Hata yakalama işleminin yapılabilmesi için try-catch blokları kullanılır. Önce yazılacak dosya açılır ve bir yazıcı oluşturulur. Sonra try içerisinde girilen bilgilerin geçerli olup olmadığı kontrol edilir. Catch bloklarında ilk önce dosyanın açılıp açılmadığı kontrol edilir. Sonra yapılan girişlerin doğru türde olup olmadığı kontrol edilir. Yani örneğin sayı girilmesi gereken yerlere metin, metin girilmesi gereken yerlere sayı girilip girilmediğini kontrol eder. En sonda IO Exception durumu kontrol edilir. En sonda giriş yapıldığında textfieldlar boşaltılır.  
  
 Oluşturulan textfieldların konumlarının manuel ayarlanması için setLayout kullanılır. Sonrasında sayfadaki öğeler panele eklenir. Sayfanın konumu ayarlanır panel sayfaya eklenir ve sayfa açılır. Fonksiyonların amacı textfieldlardaki girişlerin geçerli olup olmadığını kontrol eder.  
  
FileOperations.java :  
Dosya okuma fonksiyonlarını tutar. Aynı isimde farklı imzalı iki fonksiyon tutar. Abstractclass.javadan miras alır.(Overload işlemi yapar).   
Inheritance.java :   
FileOperations.javadan miras alır aynı isim ve imzaya sahip bir fonksiyona sahiptir(Override işlemi yapar)  
  
Funcs.java :   
ChemsPage’de kullanılacak fonksiyonları tutar.  
  
abstractclass.java :   
FileOperationstaki fonksiyonları tutar.