

Kocaeli Üniversitesi

*Programlama Lab. 3.proje

1st 200202087
Ferhat ARSLAN

ferhatarslann117@gmail.com

2nd 200202002
Berkay MALKOÇ

malkocbrky@gmail.com

Özet—Nesneye yönelik programlama ile halihazırda verilen birey bilgilerinden yararlanarak soyağacı oluşturma, aynı isime sahip insanları bulma, ata mesleğini devam ettirenleri bulma gibi fonksiyonları gerçekleştirme hedeflenmektedir.

I. ÖZET

Projemizde halihazırda verilen bireylerin id, isim, soyisim, doğum tarihi, eşi, anne adı, baba adı, kan grubu, meslek, medeni hali, kızlık soyismi, cinsiyet bilgilerini sırasıyla bir dizi içerisinde kaydettik. Kaydedilen bilgileri programımızın ilerleyen safhalarında gerek karşılaştırma gerek ekrana yazdırma fonksiyonlarında kullanmak üzere insan sınıfını oluşturduk. Bilgiler ile eş isimde String ve Int tanımladığımız değerleri aldığımız bilgilerle eşledik. İsterleri yapmak üzere menü oluşturup kullanıcının istediği kısma yönlendirdik.

II. GİRİŞ

Projemiz kapsamında insan classında tutulan bilgileri oluşturduğumuz menü üzerinden kan grubu bulma, aynı isme sahip insanları bulma gibi fonksiyonları çalıştırmak üzere kullanıcıdan alınan input sayesinde programa yön verdik.

III. YÖNTEM

İlk önce bize verilen excel dosyasında 1 sayfadan fazla bilgi varsa sayfa sayfa, 1 sayfa bilgi varsa direkt csv uzantılı olarak dosyanın uzantısını programımız okuyabileceği formata dönüştürerek programın başlangıç koşulunu yerine getirdik. Verilen dosyanın uzantısını csv olarak kaydederken, bilgilerin virgül ile ayrılmış olmasını sağladık. Bu bize programın içerisinde bilgileri ayrı ayrı kaydedebilmemizin olanağını sağladı. Tüm bu excel üzerinden csv uzantılı olarak kaydedilmiş belge üzerinden aldığımız bilgileri başlık sırasına göre sıralı olacak şekilde String bir dizi içerisinde kaydettik. Programın bizden istediği isterleri yerine getirebilmek için insan classını oluşturup başlıklarına uygun class özelliklerine kaydedilmesi için for döngüleri içerisinde belgenin içerisindeki bütün bilgileri (id, isim, soyisim, doğum tarihi, eşi, anne adı, baba adı, kan grubu, meslek, medeni hali, kızlık soyismi, cinsiyet) yine bilgilere uygun isimde değerlere kaydettik. Bütün değerleri String olarak almalı olmamız isterlerden isim ve yaş sorgusu

yapanlarda doğum tarihi sorgusu yaş hesaplaması gibi fonksiyonların çalışmasında sıkıntı çıkarması üzere oluşturduğumuz menü üzerinde bu fonksiyonları kullandığımız kısımlara yönlendiğinde program, bu bilgileri alıp substring ile gerekli kısmını ayırt ettikten sonra integervalueof fonksiyonu ile matematik işlemleri yapabileceğimiz kıvama getirdik. Bu sayede ata mesleğini devam ettiren insanları bulurken, aynı isme sahip isimlerin yaşlarını sorgularken koşulları yerine getirmede büyük birkolaylık sağlamış olduk. Bunun yanı sıra programın doğru çalışma oranını da yükseltmiş olduk. Tüm bu işlemleri yaparken kan grubu sorgusunu yapmak, class oluşturmamız üzerine epey bir kolaylaştı. sadece bireyler arasında gezinmek ve istenilen kan grubunu bulmak kalıyordu geriye. Sonrasında menümüze soyunda ata mesleğini devam ettirenlerin bulunması fonksiyonunu ekledik. Bize verilen excel dosyasında özellikle eş isimi kısmında yapılan farklılıklar bu kısımda epey bir koşul karşılatırması kullanmamızı gerektirdi. Kızlık soyismi ile yazılan ama evli olup soyismi değişmeyenler, çocuğu olup olmadığı kontrolünü yaparken zorluklar çıkarmış olsa da bunları gerekli for döngüleri ve if koşul fonksiyonları içerisinde birbiri ile karşılaştırdık. Programımızın bütün bu işlemleri yapmasının yanı sıra, daha estetik bir görünüme sahip olması için kullanıcının bütün bu işlemleri yapmasını sağladığı menü basma fonksiyonumuz terminalde çalışmasının yerine java swing kütüphanesiyle açılan pop-up penceresinde yönelmek istediği yeri seçmesini sağlıyoruz. bir iki ve üç seçimlerinden birini yapması üzerine gereken işlemleri program arka planda yaparken açılan bu pop-up penceresinde program sonuçları sıralıyor ve ikinci bir pencere tekrar menüyü gösterip hızlı bir şekilde seçim değişikliği olanağını sağlıyor. Böylelikle birinci isteri program halihazırda yapmış ve sonuçları yazdırmışken ikinci bir ister yaptırıp ekranı temizleyip ikinci isteri yürürlüğe sokuyor. Program tüm bu menü basma, kullanıcıdan değer alma ve kullanıcıdan alınan değer üzerine gerekli fonksiyonları çalıştırma gibi işlemleri yaptıktan sonra ekrana bu işlemlerin sonucunda elde edilen sonuçları yazdırırken, elde edilen sonuçlarda kadın ve erkeklerin isimlerini, kadınlar kırmızı erkekler mavi olacak şekilde yazdırıyor. Böylelikle programımıza hedeflediğimiz estetik duruşu da sağlamış oluyoruz.

IV. PROGRAMIN İŞLEYİŞİ

1-) Okunacak dosyanın varlığını kontrol et

```
File belge=new File(pathname: "Test.csv");
if(belge.exists()){
    System.out.println(x: "Belge var.");
}
else{
    System.out.println(x: "Belge yok");
}
```

2-) Dosya varsa virgüller ile teker teker oku

```
while(s.hasNextLine()){
    String satir=s.nextLine();
    String[] split = satir.split(regex: ";");
```

3-)Okunan değerleri string bir diziye ata

```
for(int i=0;i<split.length;i++){
    Kisi[kisisay]=split[i];
    System.out.println(split[i]+" ");
    kisisay++;
    kontrol++;
}
```

4-)İnsan classında constructor ve gerekli değerler oluştur

```
class insan {
    String id,dogum,esi,anne,baba,kan,
    insan(){
    }
    insan(String m,String m1,String m2
        id=m;
        isim=m1;
        soyisim=m2;
        dogum=m3;
        esi=m4;
        anne=m5;
        baba=m6;
        kan=m7;
        meslek=m8;
        meden=m9;
        kizlik=m10;
        cins=m11;
    }
}
```

5-)Koşul içerisinde değerleri nesne oluşturarak olması gereken yerlere ata

```
if(kontrol>12){
    kontrol2=kisisay-12;
    for(int m1=0;m1<say;m1++){
        String kos1=Kisi[kontrol2];
        String kos2=insanlar[m1].id;
        if(kos1.equals(kos2)){
            System.out.println(Kisi[kontrol2]);
            kosul=1;
        }
    }
    if(kosul==0){
        insanlar[say]=new insan(Kisi[kontrol2],Kisi[kontrol2+1],
        say++;
    }
    kosul=0;
}
```

6-) Kullanıcı yönlendirmesi için menü bas

```
while(menu==0){
    String kos3;
    String kos8;
    String kos9;
    String kos4;
    String kos7;
    int kos5=0,kos10;
    int day=0,day1=0,day2=0;

    //menü gelecek
    String girdi5=JOptionPane.showInputDialog(parentComponent: null, message:
    eleman = Integer.parseInt(girdi5);

    System.out.println(x: "Yapmak istediginiz islemi seciniz:");
    System.out.println(x: "1-Aynı isime sahip kisilerin yaslarini");
    System.out.println(x: "2-Kan grubu sorgulama");
    System.out.println(x: "3-Soyundaki meslegi devam ettirenleri sorgulama");
    menusec = eleman;
```

7-) Kullanıcının 1 girişi için aynı isme sahip insanları ve yaşlarını yazdır

```
switch(menusec){
    case 1:
        textPane.setText(t: "");
        for(int k=0;k<say;k++){
            for(int n=k+1;n<say;n++){
                if(insanlar[k].isim.equals(insanlar[n].
                kos3=insanlar[k].dogum;
                kos7=insanlar[n].dogum;
                if(insanlar[k].dogum.length()==9){
                    kos4=kos3.substring(beginIndex:
                    kos5=Integer.valueOf(kos4);
                    day=2022-kos5;
                }
                if(insanlar[k].dogum.length()==10){
                    kos4=kos3.substring(beginIndex:
                    kos5=Integer.valueOf(kos4);
                    day=2022-kos5;
                }
                if(insanlar[n].dogum.length()==9){
                    kos4=kos7.substring(beginIndex:
                    kos5=Integer.valueOf(kos4);
                    day1=2022-kos5;
                }
            }
        }
    }
```

8-)Kullanıcının 2 girişi için kan grubu sorgu ekranına yönlendir

V. YALANCI KOD

```
case 2:
textPane.setText(t: "");
//kullanıcıdan input al
String girdi6=JOptionPane.showInputDialog(parentComponent: null,
System.out.println(x: "Istediginiz kan grubunu giriniz:");
for(int m=0;m<say;m++){
    if(girdi6.equals(insanlar[m].kan)){
        if(insanlar[m].cins.equals(anObject: "Erkek")){
            StyleConstants.setForeground(style, Color.BLUE);
            try { doc.insertString(doc.getLength(), insanlar[m]
            catch (BadLocationException e){}
        }
        else{
            StyleConstants.setForeground(style, Color.RED);
            try { doc.insertString(doc.getLength(), insanlar[m]
            catch (BadLocationException e){}
        }
        StyleConstants.setForeground(style, Color.BLACK);
        try { doc.insertString(doc.getLength(), str: "-----
        catch (BadLocationException e){}
        System.out.println(insanlar[m].isim+ " "+insanlar[m].soy
        System.out.println(x: "-----
    }
}
break;
```

- 1) Excel dosyasını virgülle ayrılmış csv uzantılıya çevir.
- 2) Excel dosyasında 1den fazla sayfa varsa teker teker kaydet.
- 3) Program üzerinden csv dosyalarını oku.
- 4) Okunan değerleri String dizisi içerisine kaydet.
- 5) İnsan nesnesini oluştur.
- 6) İnsan nesnesi içerisindeki değerlere String dizisindeki bilgileri kaydet
- 7) Kullanıcı yönlendirmesi için menü bas.
- 8) Kullanıcının yönlenecek istediği kısımlarda yaş problemi için değer dönüşümü yap.
- 9) Yönlendirme sonucu elde edilen bilgileri yazdır.
- 10) Kullanıcı başka bir yönlendirme girmesi için tekrar menü bas.

VI. DENEYSEL SONUÇLAR

Hali hazırda verilen dosyanın varlığı kontrol edildi. Dosyanın var olması durumunda dosyadan veriler virgül koşulu ile teker teker okundu. Okunan veriler string bir dizi içerisinde kaydedildi. Kaydedilen verileri programın ileri safhalarında kullanmak üzere oluşturulan insan classı içerisinde barındırdığı bilgi dahilinde belirlenen değerlere atandı. Atanan bu veriler kullanıcının yapmak istediği etkinliği seçmesi için oluşturulan menü üzerinden, seçtiği etkinlik dahilinde gerekli koşul ifadeleri ve döngüler içerisinde doğru yönlendirme için kullanıldı. Bu yönlendirmelerin sonucunda kullanıcının istediği bilgiler ekrana yazdırıldı.

9-) Kullanıcının 3 girişi için soyundaki mesleği devam ettirenleri yazdır

```
case 3:
textPane.setText(t: "");
int mlk=0;
int mlk1=0;
String kang;
String kang1;
String kang2;
String kang3;
for(int i=0;i<say;i++){
    kos3=insanlar[i].dogum;
    if(insanlar[i].dogum.length()==9){
        kos4=kos3.substring(beginIndex: 5, endIndex: 9);
        kos5=Integer.valueOf(kos4);
        day=2022-kos5;
    }
    if(insanlar[i].dogum.length()==10){
        kos4=kos3.substring(beginIndex: 6, endIndex: 10);
        kos5=Integer.valueOf(kos4);
        day=2022-kos5;
    }
}
```

VII. AKIŞ DIYAGRAMI

Akış Diyagramı:



VIII. SONUÇ

Kullanıcı excel üzerinden csv uzantılı olarak kaydedilmiş belgeden çekilen bilgilerin kaydolduğu insan classından kontrolü yapılan bilgiler üzerinden, programın oluşturduğu menü üzerinden yapmak istediği işlemi seçerek, dilediği kısma yönelenerek bütün işlemlerini yapabilir.

IX. OLUŞTURMA ORTAMI

Programımızın yapımını Java diliyle Visual Studio Code programı üzerinden tamamladık. Raporumuzu ise Latex ile hazırladık.

KAYNAKLAR

- [1] <https://www.java.com/tr/>
- [2] <https://www.yazilimbilisim.net/java/java-sinif-ve-nesne-olusturma/>
- [3] 2020-2021 Nesneye Yönelik Programlama slaytları -Ahmet SAYAR
- [4] <https://tr.bitdegree.org/tutorial/java-ogrenmek/>
- [5] <https://bilgisayarkavramlari.com/2011/06/01/swing-kutuphanesi/>