

## 2022-2023 GÜZ YARIYILI NESNEYE YÖNELİK PROGRAMLAMA DERSİ

### 2. ÖDEVİ

#### Dershane Otomasyonu

Bir dershaneye kaydolun öğrencileri ve bu dershanede verilen derslerin takip edilebilmesini sağlayan bir otomasyon yapılacaktır. Bu dershanede birçok ders verilmektedir. Dershaneye kaydolun öğrenci bir veya birden fazla derse kaydolabilmektedir. Öğrenci bilgileri, ders bilgileri ve öğrencilerin kaydolduğu ders bilgileri tutulacaktır.

Otomasyon **Ogrenci.java**, **Ders.java** ve **Anasayfa.java** sınıflarından oluşmaktadır. Anasayfa.java sınıfında **main** fonksiyonu bulunmaktadır. Ogrenci.java ve Ders.java sınıflarının içermesi gereken özellikler aşağıdaki gibidir. Bu iki sınıf **Kapsülleme** (Encapsulation – sınıf özellikleri private olmalıdır, Ogrenci.java’daki alınanDersler özelliği hariç) özelliklerini sağlamalıdır. Ders.java sınıfında dersId özelliği 1000’den başlanarak 10’ar 10’ar artarak otomatik artırılıp atanlamalıdır. Sınıflara istediğiniz yapıda constructor metot ekleyebilirsiniz.

#### Ogrenci.java

- ogrId (int)
- ogrAdSoyad (String)
- ogrYas (int)
- alınanDersler (ArrayList<Ders>)

#### Ders.java

- dersId (int)
- dersAd (String)

Öğrencilerin bilgileri **ogrenci.txt** dosyasında tutulacaktır. Ders bilgileri ise **ders.txt** dosyasında tutulacaktır. Örnek dosya içerikleri aşağıdaki gibidir:

#### ogrenci.txt

```
+5050% Ahmet Ada% 17
*1000% Matematik
*1040% Fizik
+6085% Selda Şahin% 16
*1020% Edebiyat
+5090% Deniz Arslan% 18
*1000% Matematik
*1060% Biyoloji
+6174% Tarık Kılıç% 17
*1010% Tarih
*1030% Coğrafya
*1050% Kimya
+7075% Ayşe Kaya% 18
*1070% Müzik
*1080% Tiyatro
*1100% Geometri
```

#### ders.txt

```
1000% Matematik
1010% Tarih
1020% Edebiyat
1030% Coğrafya
1040% Fizik
1050% Kimya
1060% Biyoloji
1070% Müzik
1080% Tiyatro
1090% Dil Bilgisi
1100% Geometri
```

Yukarıdaki dosya içeriklerinde görüldüğü gibi ogrenci.txt dosyasında satırın ilk karakteri ‘+’ ile başlıyorsa öğrencinin bilgileri olduğu ve hemen altında ‘\*’ karakteri ile başlayan satırlar o öğrencinin aldığı dersleri ifade etmektedir. İki dosyada da her bilgi ‘%’ karakteri ile

birbirinden ayrılmıştır. Dosyadan okuma yapılırken ve dosyalara bilgi yazılırken bu formatlara dikkat edilmelidir.

AnaSayfa.java sınıfında işlemler menü kontrolü gerçekleştirilecektir. Menü’ den bitir/çıkış fonksiyonu seçilene kadar program çalışır durumda olacaktır. Program ilk çalıştırıldığında öğrenci bilgileri ogrenci.txt dosyasından okunup öğrenciler listesine (**Oğrenciler ArrayList<Oğrenci>**) eklenecektir. Aynı şekilde ders.txt dosyasının içindeki ders bilgileri dersler listesine (**Dersler ArrayList<Ders>**) eklenecektir. Dosya okuma işlemi sona erdikten sonra kullanıcıya aşağıdaki gibi bir menü sunulacaktır:

**1 – Ders Ekle**

**2 – Ders Listele**

**3 – Ders Ara**

**4 – Ders Sil**

**5 – Öğrenci Ekle**

**6 – Öğrenci Ara**

**7 – Öğrenci Sil**

**8 – Öğrenci Listele**

**9 – Öğrencileri Ayrıntılı Listele**

**10 – Öğrencilerin Ödeyeceği Tutarı Hesapla**

**11 - Çıkış**

Anasayfa.java sınıfında yer alacak menünün sağlayacağı metotlar aşağıda listelenmiştir. Menünün her bir seçeneği bir metot ile yapılacaktır.

1. **DersEkle();** //Yeni dersin adı kullanıcıdan alınacak ve **dersler array list’sine** eklenecektir.
2. **DersListele();** //Dersler listesindeki tüm derslerin bilgileri ekrana aşağıdaki biçimde yazılacaktır.

<b>Ders Id</b>	<b>Ders Adı</b>
1000	Matematik
1010	Tarih
1020	Edebiyat
1030	Cografya
1040	Fizik
1050	Kimya
1060	Biyoloji
1070	Müzik
1080	Tiyatro
1090	Dil Bilgisi
1100	Geometri

3. **DersAra();** //Kullanıcıdan alınan ders adına göre arama yapılacak ve bulunan ders listelenecektir. Dersin bulunamaması durumunda kullanıcıya bulunamadı mesajı verilecektir.
4. **DersSil();** //Kullanıcıdan alınan ders adına göre ilgili ders listeden silinecektir. Silme işlemi yaparken dersi alan öğrenci/öğrenciler varsa silme işlemine izin verilmeyecek ve bu mesaj kullanıcıya verilecektir.
5. **OgrenciEkle();** //Eklenicecek olan öğrenciye ait bilgiler (kaydolacağı dersler de dâhil) kullanıcıdan alınacak ve yeni öğrenci, **Oğrenciler array list'ine** eklenecektir. Ekleme yapılırken aynı Id ile başka kayıt olup olmadığı kontrol edilecektir.
6. **OgrenciAra();** //Kullanıcıdan alınan öğrenci ad-soyadına göre arama yapılacaktır. Eğer öğrenci listede bulunursa bilgileri ekrana yazılacak, bulunamazsa bulunamadı mesajı verilecektir.
7. **OgrenciSil();** //Kullanıcıdan alınan Id bilgisine göre öğrenci Öğrenciler array list'inden silinecektir.
8. **OgrenciListele();** //Listedeki tüm öğrenci bilgileri aşağıdaki formatta ekrana yazdırılacak.

**Tüm Öğrenciler**

5050 Ahmet Ada 17  
6085 Selda Şahin 16  
5090 Deniz Arslan 18  
6174 Tarık Kılıç 17  
7075 Ayşe Kaya 18

9. **OgrenciAyrıntılıListele();** //Listedeki tüm öğrenciler ve aldığı ders bilgileri aşağıdaki formatta ekrana yazdırılacak.

**Tüm Öğrenciler ve Aldıkları Dersler**

5050 Ahmet Ada 17  
1000 Matematik  
1040 Fizik  
6085 Selda Şahin 16  
1020 Edebiyat  
5090 Deniz Arslan 18  
1000 Matematik  
1060 Biyoloji  
6174 Tarık Kılıç 17  
1010 Tarih  
1030 Coğrafya  
1050 Kimya  
7075 Ayşe Kaya 18

1070 Müzik  
1080 Tiyatro  
1100 Geometri

10. **OgrencilerinOdeyeceğiTutarHesaplama();** //Kullanıcıdan alınan Id bilgisine göre öğrencinin dershaneye ödeyeceği aylık toplam ücret hesaplanmalıdır. Her ders ücreti haftalık 400 TL'dir. Kurs kampanyaları şu şekildedir.

**Kampanya 1:** Bu kampanya 2 ders alan öğrenciler içindir. Bu öğrencilere ikinci ders %5 indirimlidir.

**Kampanya 2:** Bu kampanya 3 ders alan öğrenciler içindir. Bu öğrencilere 3. ders %15 indirimlidir.

**Kampanya 3:** Bu kampanya 3 ders üstü alan öğrenciler içindir. Bu öğrencilere ise her ders %10 indirimlidir.

Bu bilgilere dayanarak öğrencilerin aylık dershaneye ödeyeceği toplam ücreti hesaplayınız ve öğrencinin hangi kampanyadan faydalandığı bilgisini veriniz. Öğrenci tek ders almışsa tek ders alana kampanya olmadığı bilgisini de ekleyiniz.

11. **Bitir();** //Ogrenciler listesindeki öğrenci bilgileri **ogrenci.txt** dosyasına **en baştaki formatta** yazılacaktır. Dersler listesindeki ders bilgileri **ders.txt** dosyasına **en baştaki formatta** yazılacaktır. Dosyaya yazma işlemi yapılacağı zaman, dosyadaki önceki kayıtların silinmesi ve yeni bilgilerin yazılması gerekmektedir.

### Ödev Gönderim Formatı:

1) Gönderilecek Dosyalar: **Ogrenci.java, Ders.java, Anasayfa.java** ve **ÖğrenciNumarası.java(Örn:20202020.java içerisinde tüm kodlarınızı kopyaladığımız dosya)** dosyalarını ÖğrenciNumaranız olarak isimlendirilen bir klasöre attıktan sonra bu klasörü (rar/zip/vs...) sıkıştırıp gönderilecektir. **ÖğrenciNumarası.java** sınıfı içerisinde **Ogrenci.java, Ders.java, Anasayfa.java** sınıflarının kodları sınıflar alt alta olacak şekilde kopyalanacaktır. Her bir sınıfın en üst kısmına yorum satırı içinde isim-soyisim öğrenci öğrenci no eklenmelidir.

2) Son Gönderim Tarihi: **05.01.2023 23:55**

**NOT: ZAMANINDA GÖNDERİLMİYEN, EKSİK GÖNDERİLEN, FORMAT DIŞINDA GÖNDERİLEN ÖDEVLER DEĞERLENDİRMEYE ALINMAYACAKTIR. ÖDEVLER BENZERLİK PROGRAMINA TABİİ TUTULUP, KOPYA OLDUĞU TESPİT EDİLEN ÖDEVLER KESİNLİKLE DEĞERLENDİRMEYE ALINMAYACAKTIR.**