

2023-2024 GÜZ YARIYILI NESNEYE YÖNELİK PROGRAMLAMA DERSİ

ÖDEV-2

Spor Merkezi Otomasyonu

Bir spor merkezinin kursiyerleri ve bu spor merkezinde yer alan kursların takip edilebilmesini sağlayan bir otomasyon yapılacaktır. Bu spor merkezinde birçok branşta kurs bulunmaktadır. Spor merkezine kaydolun kursiyer bir veya birden fazla kursa kaydolabilmektedir. Kursiyer bilgileri, kurs bilgileri ve kursiyerlerin kayıt oldukları kurs bilgileri tutulacaktır.

Otomasyon **Kursiyer.java**, **Kurs.java**, **Anasayfa.java** sınıflarından ve **Hesaplama.java** arayüzünden oluşmaktadır. **Anasayfa.java** sınıfında **main** fonksiyonu bulunmaktadır. **Kursiyer.java** ve **Kurs.java** sınıflarının içermesi gereken özellikler ve **Hesaplama.java** arayüzünün içermesi gereken metod aşağıdaki gibidir. **Kursiyer.java** ve **Kurs.java** sınıfları **Kapsülleme** (Encapsulation – sınıf özellikleri private olmalıdır, **Kursiyer.java**'daki **alınanKurslar** özelliği hariç) özelliklerini sağlamalıdır.

Kursiyer.java

- kursiyerId (int)
- kursiyerAdSoyad (String)
- kursiyerYas (int)
- alınanKurslar (ArrayList<Kurs>)

Kurs.java

- kursId (int)
- kursAd (String)

Hesaplama.java

- BorcHesapla()

Kursiyerlerin bilgileri **kursiyer.txt** dosyasında tutulacaktır. Kurs bilgileri ise **kurs.txt** dosyasında tutulacaktır. Örnek dosya içerikleri aşağıdaki gibidir:

kursiyer.txt

```
*5050+Ahmet Ada+23
%1000+Futbol
%1040+Masa Tenisi
*6085+Selda Naz Şahin+35
%1020+Voleybol
*5090+Deniz Arslan+30
%1000+Futbol
%1060+Karate
*6174+Tarık Kılıç+42
%1010+Basketbol
%1030+Tenis
%1050+Judo
*7075+Ayşe Nur Kaya Yaman+20
%1070+Boks
%1080+Fitnes
%1100+Yüzme
```

kurs.txt

```
1000+Futbol
1010+Basketbol
1020+Voleybol
1030+Tenis
1040+Masa Tenisi
1050+Judo
1060+Karate
1070+Boks
1080+Fitnes
1090+Zumba
1100+Yüzme
```

Yukarıdaki dosya içeriklerinde görüldüğü gibi kursiyer.txt dosyasında satırın ilk karakteri ‘*’ ile başlıyorsa kursiyerin bilgilerini ve hemen altında ‘%’ karakteri ile başlayan satırlar o kursiyerin aldığı kursları ifade etmektedir. İki dosyada da her bilgi ‘+’ karakteri ile birbirinden ayrılmıştır. Dosyadan okuma yapılırken ve dosyalara bilgi yazılırken bu formatlara dikkat edilmelidir.

Hesaplama.java arayüzüne BorcHesapla() isminde bir abstract metot yazılmalıdır. Bu arayüz Kursiyer.java sınıfı tarafından implemente edilmeli ve ilgili metot doldurulmalıdır. Metot içeriği aşağıda verilmiştir.

AnaSayfa.java sınıfında işlemler menü kontrolü gerçekleştirilecektir. Menü’den çıkış seçeneği seçilene kadar program çalışır durumda olacaktır. Program ilk çalıştırıldığında kursiyer bilgileri kursiyer.txt dosyasından okunup kursiyerler listesine (**Kursiyerler ArrayList<Kursiyer>**) eklenecektir. Aynı şekilde kurs.txt dosyasının içindeki kurs bilgileri kurslar listesine (**Kurslar ArrayList<Kurs>**) eklenecektir. Dosyadan bilgi okuma işlemi sona erdikten sonra kullanıcıya aşağıdaki gibi bir menü sunulacaktır:

1 – Kurs Ekle

2 – Kurs Listele

3 – Kurs Ara

4 – Kurs Sil

5 – Kursiyer Ekle

6 – Kursiyer Ara

7 – Kursiyer Sil

8 – Kursiyerleri Listele

9 – Kursiyerleri Ayrıntılı Listele

10 – Kursiyerin Ödeyeceği Tutarı Hesapla

11 – Çıkış

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

1. **KursEkle();** //Yeni kurs bilgileri (kursID ve kursAd) kullanıcıdan alınacak ve **kurslar arraylist’ine** eklenecektir. Ekleme yapılırken aynı ID ile başka kayıt olup olmadığı kontrol edilecektir.

2. **KursListele();** //Listedeki tüm kursların bilgileri ekrana aşağıdaki biçimde yazılacaktır.

Kurs Id	Kurs Adı
1000	Futbol
1010	Basketbol
1020	Voleybol

1030	Tenis
1040	Masa Tenisi
1050	Judo
1060	Karate
1070	Boks
1080	Fitnes
1090	Zumba
1100	Yüzme

3. **KursAra();** //Kullanıcıdan alınan kurs adına göre arama yapılacak ve bulunan kurs listelenecektir. Kursun bulunamaması durumunda kullanıcıya bulunamadı mesajı verilecektir.
4. **KursSil();** //Kullanıcıdan alınan kurs adına göre ilgili kurs listeden silinecektir. Silme işlemi yaparken kursu alan kursiyer/kursiyerler varsa silme işlemine izin verilmeyecek ve bu mesaj kullanıcıya verilecektir.
5. **KursiyerEkle();** //Eklenicecek olan kursiyere ait bilgiler (id, adsoyad, yaş ve kaydolacağı kurslar da dâhil) kullanıcıdan alınacak ve yeni kursiyer, **Kursiyerler arraylist'ine** eklenecektir. Ekleme yapılırken aynı Id ile başka kayıt olup olmadığı kontrol edilecektir.
6. **KursiyerAra();** //Kursiyer ad-soyadına göre aranacak. Eğer kursiyer listede bulunursa bilgileri ekrana yazılacak, bulunamazsa bulunamadı mesajı verilecektir.
7. **KursiyerSil();** //Kullanıcıdan alınan Id bilgisine göre kursiyer Kursiyerler arraylist'inden silinecektir.
8. **KursiyerListele();** //Listedeki tüm kursiyer bilgileri aşağıdaki formatta ekrana yazdırılacak.

Tüm Kursiyerler

5050 Ahmet Ada 23
6085 Selda Naz Şahin 35
5090 Deniz Arslan 30
6174 Tarık Kılıç 42
7075 Ayşe Nur Kaya Yaman 20

9. **KursiyerAyrıntiliListele();** //Listedeki tüm kursiyerler ve aldıkları kurs bilgileri aşağıdaki formatta ekrana yazdırılacak.

Tüm Kursiyerler ve Aldıkları Kurslar

5050 Ahmet Ada 23
1000 Futbol
1040 Masa Tenisi
6085 Selda Naz Şahin 35
1020 Voleybol

5090 Deniz Arslan 30
1000 Futbol
1060 Karate
6174 Tarık Kılıç 42
1010 Basketbol
1030 Tenis
1050 Judo
7075 Ayşe Nur Kaya Yaman 20
1070 Boks
1080 Fitnes
1100 Yüzme

10. **KursiyerinÖdeyeceğiTutarHesapla(kursiyerId);** //Id'si verilen kursiyerin spor merkezine ödeyeceği aylık toplam ücret hesaplanmalıdır. Bu metod Id'si verilen kursiyerin BorcHesapla() metodunu kullanarak hesaplama yapar. BorcHesapla() metodu şu şekilde çalışır; Her kurs ücreti haftalık 500 TL'dir. Kurs kampanyaları şu şekildedir.

Kampanya 1: Bu kampanya 2 kurs alan kursiyerler içindir. Bu kursiyerlere ikinci kurs %20 indirimlidir.

Kampanya 2: Bu kampanya 3 kurs alan kursiyerler içindir. Bu kursiyerlere 3. kurs %25 indirimlidir.

Kampanya 3: Bu kampanya 3 kurs üstü alan kursiyerler içindir. Bu kursiyerlere ise her kurs %10 indirimlidir.

Bu bilgilere dayanarak kursiyerin aylık spor merkezine ödeyeceği toplam ücreti hesaplayınız ve kursiyerin hangi kampanyadan faydalandığı bilgisini veriniz. Kursiyer tek kurs almışsa tek kurs alana kampanya olmadığı bilgisini de ekleyiniz.

11. **Cikis();** //Kursiyerler listesindeki kursiyer bilgileri **kursiyer.txt** dosyasına **en baştaki formatta** yazılacaktır. Kurslar listesindeki kurs bilgileri **kurs.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: **Kursiyer.java, Kurs.java, Anasayfa.java, Hesaplama.java** ve **Tum_kisaogrno.java (Örn: Tum_2012.java)** dosyalarını **OgrenciNumaranız** olarak isimlendirilen bir klasöre attıktan sonra bu klasörü (rar/zip/vs...) sıkıştırıp gönderilecektir. Tum_kisaogrno.java bir sınıf değildir. Bu dosyaya Kursiyer, Kurs, Anasayfa, Hesaplama java dosyalarının içeriklerini arka arkaya kopyalamanız gerekmektedir. Benzerlik kontrolü için gerekli bir dosyadır. Her birinizin mutlaka göndermesi gerekmektedir, göndermeyen ödevler değerlendirilmeyecektir. Her class'ın en üst kısmında yorum satırı olarak ad soyad ve öğrenci numarası bulunmalıdır.

2) Son Gnderim Tarihi: **12.01.2024 23:55**

NOT: ZAMANINDA GNDERİLMİYEN, EKSİK GNDERİLEN, FORMAT DIřINDA GNDERİLEN DEVLER DEęERLENDİRMEYE ALINMAYACAKTIR. DEVLER BENZERLİK PROGRAMINA TABİİ TUTULUP, KOPYA OLDUęU TESPİT EDİLEN DEVLER KESİNLİKLE DEęERLENDİRMEYE ALINMAYACAKTIR.

NOT: GEÇMİř ZAMANLARDA GNDERİLMİř DEVLER DE BENZERLİęE KATILACAKTIR.