Nesneye Dayalı Programlama
2021-2022 GÜZ
Bilgisayar Mühendisliği Bölümü
Dönem Projesi

Proje-1:

Konu: Nesneye Dayalı Programlamanın Temel Yapıları, Tasarım Desenleri ve JUnit Kullanımı

Soru:
Aşağıda temel gereksinimleri verilen uygulamayı Java dili ile gerçekleştiriniz. Proje raporu da aşağıda belirtildiği şekilde hazırlanarak ayrıca yüklenecektir.

Temel Gereksinimler:

* Bir firmanın çalışanlarına ilişkin hiyerarşik yapının, uygun bir tasarım deseni kullanılarak temsil edilmesi istenmektedir. Firmada, "Direktör" ve "Memur" çalışanların olduğu varsayılmaktadır. Bir direktörün emrinde sıfır veya N tane çalışan (memur veya direktör) olabilmektedir. Çalışanlar için String tipinde adSoyad ve tamsayı (int) tipinde maaş verileri bulunmaktadır.

Oluşturulması istenen yapıda desteklenmesi istenen iki işlem vardır. Bunlardan birincisi, bir direktörün hiyerarşik olarak altında çalışanların listelenmesidir (direktörün kendi adı da liste başına yazılabilir.) (Eğer memur için emrinde çalışanlar listelenmek istenirse, memurun adı liste başına yazılabilir ancak doğal olarak liste boş kalacaktır.) Diğer işlem ise, her hangi bir çalışanın firmaya maliyetinin hesaplanmasıdır: Direktörlerin maliyeti kendi maaşı artı emrinde çalışanların maaşları toplamıdır. Bir memurun firmaya maliyeti ise sadece kendi maaşı kadardır. Söz edilen iki işlem tasarımınızda ilgili sınıflar içinde desteklenmelidir. Bunların dışında gerekli olan tüm metotlar sizin tarafınızdan yazılmalıdır.

* Hiyerarşik yapının dolaşılması için Iterator deseni kullanılacaktır. Iterator desenini kendinizin baştan yazması istenmektedir, Java ortamındaki hazır Iterator desenini kullanmayınız. Hiyerarşik yapıda ihtiyacınız olan koleksiyonu dizi tabanlı olarak (ArrayList değil). Iterator deseni çerçevesinde tanımlamanız gerekmektedir.

* Aşağıda verilen hiyerarşik yapı, yine bu yapının arkasından aşağıda verilen girdi.txt dosyasında ifade edilmiştir.

D(Mustafa Turksever,5000) *D:DİREKTÖR, M:MEMUR anlamındadır.*
D(Halil Sengonca,4000)
D(Ugur Guclu,2000)
M(Emre Kosar,700) M(Ahmet Egeli,700)
D(Sedat Tunc,2500)
M(Bora Kuzey,1000)
D(Oguz Demir,3000)

M(Önder Bati,500) M(Erdem Altin,500) M(Mehmet Bilir,600) D(Bahar Karaoglan,3500)

girdi.txt

D,Mustafa Turksever,5000,Root
D,Halil Sengonca,4000,Mustafa
D,Bahar Karaoglan,3500,Mustafa
D,Ugur Guclu,2000,Halil
D,Sedat Tunc,2500,Halil
D,Oguz Demir,3000,Halil
M,Emre Kosar,700,Ugur
M,Ahmet Egeli,700,Ugur
M,Bora Kuzey,1000,Sedat
M,Onder Bati,500,Oguz
M,Erdem Altin,500,Oguz
M,Mehmet Bilir,600,Oguz

Bu dosyada, tek karakterlik ilk saha ilgili girdinin direktör mü yoksa memur mu olduğudur. İkinci sahada çalışan ismi, üçüncü sahada maaşı, dördüncü sahada ise hangi çalışana bağlı olduğu bilgisi vardır. Bir girdinin hangi çalışana bağlı olduğu sahası "root" ise en üst direktör demektir.

Bir Deneme sınıfı yazarak, bu sınıfın main metodu içinde yukarıda verilen dosyayı okuyarak verilmiş olan hiyerarşik yapıyı ilgili nesnelerle (grafiksel değil) oluşturunuz. Daha sonra, Mustafa Turksever isimli direktörün maliyeti ve emrinde çalışanlar listesi, Oğuz Demir isimli direktörün maliyeti, emrinde çalışanlar listesi ve Ahmet Egeli isimli memurun maliyetlerini yazdırınız.

* Deneme sınıfını yazarken ilgili girdiler dosyadan okunmuştur. Girdilerin ileride farklı veritabanları veya ortamlardan da okunabileceğini düşünerek, gerekli esnekliği sağlamak üzere bir üst madde kapsamında hiyerarşik yapıyı oluşturmak üzere yazdığınız kodları veri erişimi için uygun bir tasarım deseni çerçevesinde yazınız. Bu amaçla size göre en uygun GoF desenini belirleyiniz. ***DAO, Repository gibi desenler şu anda istenmemektedir. Bu madde için gerek proje raporunda gerekse kaynak kodlarda detaylı açıklamalar yapmanız gerekmektedir.***

* Geliştirdiğiniz uygulamaya ilişkin bir işlevselliği (veya bir senaryoyu) gerçekleştirirken, JUnit kullanarak birim testlere dayalı geliştiriniz. Her ne kadar "Extreme Programlama" nın gerçek hayat uygulamalarında tüm uygulamanın birim testlere dayalı olarak geliştirilmesi gerekse de, nesneye dayalı programlamanın diğer taraflarına daha çok odaklanabilmeniz için bu proje kapsamında geliştirdiğiniz uygulamanın tümünü birim testler ile gerçekleştirmenize gerek yoktur.