



Fenerbahçe Üniversitesi

COMP 205 – Nesneye Yönelimli Programlama

LAB 8: Tasarım Kalıpları 2

Önemli Not: Aşamaları tamamladıkça, dersin hocası veya asistanı yanınıza çağırarak, tamamladığınız aşamayı gösterdikten sonra diğer aşamaya geçiniz.

LAB'ın aşamaları ve puanlar:

1- Ön aşamalar (0 Puan)

Order, Stock, Broker ve CommandPatternDemo isiminde 4 sınıf tanımlayınız.

Order sınıfında herhangi bir işlem yapmayan execute fonksiyonu tanımlanmalıdır.

Stock sınıfında buy ve sell isimli fonksiyonlar bulunmaktadır. Bu fonksiyonlardan ekrana satılan ve satın alınan ürünün ismi ve miktarı yazılmaktadır.

buyStock ve sellStock sınıfları Order sınıfından kalıtım yapmaktadır.

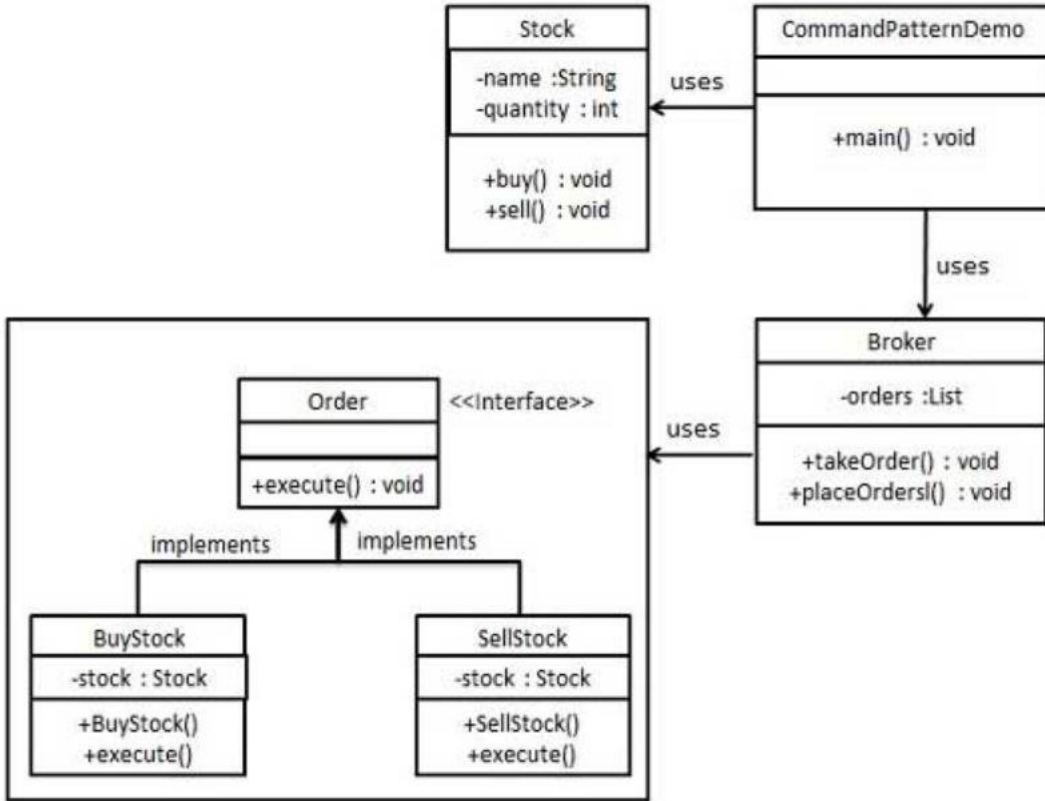
Broker sınıfında liste tanımlaması, takeOrder ve placeOrders fonksiyonları tanımlanacaktır. takeOrder fonksiyonları gelen elemanları listeye eklemektedir. placeOrder sınıfında bu listedeki elemanlar execute edilerek çalıştırılmaktadır.

Program çalışmaya CommandPatternDemo sınıfından başlayacaktır. Stock, buyStock, sellStock, Broker sınıflarından obje oluşturunuz. Komutları almak ve yürütmek için Broker sınıfı kullanılacaktır.

Aşağıdaki çıktıyı veren programın kod parçacıklarını yazınız.

```
Stock: Iphone 12 Quantity: 10 bought  
Stock: Iphone 12 Quantity: 10 sold
```

(Yazılacak programın diyagramı aşağıdaki gibidir)



2-

motorKontrol ve kanatKontrol isminde iki sınıf tanımlayınız.

motorKontrol sınıfında, private tanımlanmış motorHizi değişkeni, motorHiziDegistir ve motorHiziAl fonksiyonu bulunmaktadır. motorHiziDegistir fonksiyonu aldığı değişkeni, motorHizina atamakta, motorHiziAl fonksiyonu ise geriye motorHizini döndürmektedir.

kanatKontrol sınıfında, private tanımlanmış kanatAcisi değişkeni, kanatAcisiAyarla ve kanatAcisiAl fonksiyonu bulunmaktadır. kanatAcisiAyarla fonksiyonu aldığı değişkeni, kanatAcisi değişkenine atamakta, kanatAcisiAl fonksiyonu ise geriye kanatAcisini döndürmektedir.

Komutlar isminde bir sınıf tanımlayıp, calistir ve degerOku isminde iki fonksiyon tanımlayınız.

calistir fonksiyonu, argüman olarak 3 değişken almaktadır. İlk değişken sınıfTuru, ikinci değişken sınıfObjesi, üçüncü değişken ise deger'dir. Aldığı sınıfTuru argümanının değerine göre, sınıfObjesi.kanatAcisiAyarla veya sınıfObjesi.motorHiziAyarla fonksiyonlarını çağırarak ve argüman olarak üçüncü deger değişkenini verecektir.

degerOku fonksiyonu iki argüman almaktadır. Birincisi sınıfTuru, ikincisi ise sınıfObjesidir. Aldığı sınıfTurune göre, sınıfObjesi.kanatAcisiAl veya sınıfObjesi.motorHiziAl fonksiyonlarını çağırarak dönen degeri ekrana basacaktır.

Komutlar, motorKontrol ve kanatKontrol sınıflarından birer obje oluşturunuz. Komutlar sınıfından oluşturulan objenin calistir ve degerOku fonksiyonlarını; motorKontrol ve kanatKontrol sınıflarından yaratılan objeleri besleyerek test ediniz.

3-

Student, StudentView, StudentController ve MVCPatterDemo sınıflarını tanımlayınız.

Student sınıfında setName, getName, SetRollNo, getRollNo fonksiyonları yer alacaktır, gelen name ve rollNo değişkenleri set fonksiyonlarında güncellenip, çağrıldığında get fonksiyonlarıyla geri döndürülecektir.

StudentView sınıfında ekrana student name ve rollno bilgilerini gösterecek printStudentDetails fonksiyonu yer alacaktır.

StudentController sınıfında Student sınıfından tanımlanmış model, StudentView sınıfından tanımlanmış view değişkenleri tanımlanacaktır. setStudentName, getStudentName, setStudentRollNo, getStudentRollNo ve printStudentDetails fonksiyonun çağrılacağı updateView fonksiyonu tanımlanacaktır.

Programın main fonksiyonu MVCPatternDemo sınıfında tanımlanacaktır. retrieveStudentFromDatabase fonksiyonunda student name ve rollNo bilgileri verilecektir. setStudentName ile name bilgisini güncelleyip önceki ve sonraki bilgileri ekranda gösteren programın kod parçacıklarını yazınız.

Örnek Çıktı:

```
Student:  
Name: Acun  
Roll No: 10  
Student:  
Name: Ugur  
Roll No: 10
```

(Yazılacak programın diyagramı aşağıdaki gibidir)

