

NESNEYE YÖNELİK PROGRAMLAMA DERSİ DÖNEM PROJESİ

PROJENİN AMACI

Bu projede Nesne Yönelimli Programlama dersi kapsamında öğrenilen bilgileri kullanarak bir banka uygulaması geliştirilmesi amaçlandı.

İŞLEM BASAMAKLARI

1. Main metodunun olduğu MainBank sınıfında ad,soyad,email,telefon numarası ve random 11 haneli TC kimlik numarası oluşturulup bu değişkenler yaratılan kişi nesnesine parametre olarak verilerek Kisi sınıfının değişkenlerine atanır. Eğer yeni bir kişi oluşturmak istenirse sadece burada Kisi nesnesi yaratılması yeterli olacaktır.
2. Üretilen Kisi nesnesinin constructoründe alınan parametreler o kişinin değişkenlerine atanır ve kullanıcının sisteme banka personeli olarak mı yoksa müşteri olarak mı devam etmek istediği sorulur.
 - Kullanıcı eğer banka personeli ise personelID'sini girmesi istenir(random 5 haneli olarak girildi). Personelin hangi müşterisinden işlem yapmak istediğini öğrenmek için müşterisine ait olan "müşteri numarasını" girmesi istenir. Personele, bu müşteri için var olan hesabının silinmesini mi yoksa yeni bir hesap eklenmesini mi istediği sorulur. Eğer personel müşterinin hesabını silmeyi seçerse Musteri sınıfının hesapSil methodunda toplam bakiyesinin kontrolü yapılır. Müşterinin bakiyesi boşsa hesabı silinir ve aynı zamanda personelin müşteriler ArrayList'inden de bu hesap kaldırılır. Müşterinin bakiyesi boş değilse bakiyesini başka bir hesaba aktarması için uyarı verilir ve toplam bakiyesi ekrana bastırılır. Hesap eklemeyi seçerse aşağıda açıklanan müşterinin hesap eklemesi metoduna gidip aynı işlemler bu müşteri için de yapılır.
 - Kullanıcı eğer müşteriye "müşteri numarasını" girmesi istenir ve hesap eklemek için 1'i, hesap silmek içinse 1'den farklı sayıyı girmesi istenir (örneğin 2).
 - Hesap eklemek istenirse Musteri sınıfının constructorunde hesapEkle methodu çağırılır ve eklenmek istenen hesabın maaş hesabı mı yoksa normal hesap mı olacağı belirlenir. Cevaba göre eklenecek hesabın hesap bilgisi BankaHesap sınıfının altındaki private hesap bilgisine setHesapBilgisi methoduyla atanır. Hesap bilgileri, Musteri sınıfında random olarak üretilen Iban ve Musteri sınıfında BankaHesap nesnesi üretilirken parametre olarak verilen toplam bakiyesi (6000 TL) ile ekrana bastırılır. Açılış tarihi ise

BankaHesap sınıfının hesapGoruntuleme methodunda hesabın açıldığı tarih eğer 1 ocaktan sonraysa şartını kontrol ederek ekrana bastırılır. Vadeli, vadesiz, yatırım hesabı yine aynı şekilde belirlenip BankaHesap sınıfındaki private hesapTuru değişkenine setHesapTuru metoduyla atanır. Aylık geliri Musteri sınıfının “gelir” değişkeninde 10 bin TL olarak belirlenen hesap,

- Maaş hesabı-Vadeli hesap ise faiz oranı %20 , Normal hesabı-Vadeli hesap ise faiz oranı %10 olarak işlenip polimorfizm ile hesabın vadeliHesap olması sağlanır. Böylelikle VadeliHesap constructorunde kişinin faizden kazandığı aylık miktar vadeli bakiyeye atanır ve hesaptaki toplam miktar ekrana bastırılır. Dolayısıyla faiz oranı belirlenen vadeli hesap başarıyla oluşturulur ve Musteri sınıfındaki hesaplar ArrayList’e eklenir.
- Maaş hesabı-Vadesiz hesap ise VadesizHesap constructorunde hesaptan para transfer edildiğinde kesinti ücretinin 0 TL olacağını , Normal hesabı -Vadesiz hesap ise kesinti ücretinin 8 TL olacağı belirlenir. Toplam bakiyeden de paraTranser metodunun geriye döndürdüğü transfer etmek istenen işlem miktarı(200 TL belirlendi) ile kesinti ücreti çıkartılarak transfer sonrası toplam bakiyenin yeni değeri ekrana bastırılır.
- Eklenmek istenen hesap yatırım hesabı ise kullanıcı, kuru 20 olarak belirlenen altın üzerinden veya kuru 10 olarak belirlenen döviz üzerinden ilerleyebilir. Bunlardan biri seçildikten sonra YatırımHesap constructorunde 100 birim üzerinden para ekleme veya para bozma işlemlerini, gerekli metotları çağırıp parametre olarak aldığı toplam bakiye üzerinden işlemi yaparak yeni toplam bakiyeyi belirler ve ekrana bastırır.

Böylelikle belirlenen hesap Musteri sınıfındaki hesaplar ArrayList’ine de eklenir.

- Hesap silmek istenirse Musteri sınıfının hesapSil methodunda toplam bakiyesinin kontrolü yapılır. Müşterinin bakiyesi boşsa hesabı silinir ve aynı zamanda hesaplar arraylist’inden bu hesap kaldırılır. Müşterinin bakiyesi boş değilse bakiyesini başka hesaba aktarması gerektiği uyarısı yapılır ve toplam bakiyesi ekrana bastırılır.
- Kredi kartı eklemek istenirse kullanıcıya kartın limitinin 5000 TL altı mı (3000 TL belirlendi) yoksa 10 bin TL üstü mü (12 bin belirlendi)

sorulur. Kisi sınıfında 600 TL belirlenen guncelBorc değişkenini, 5 haneli random kart numarasını ve seçilen limit değerini parametre alan yeni KrediKartı nesnesi üretilir ve constructorde alınan limit değeri 5 bin TL altı ise ekrana müşterinin güncel borcu yazdırılır. Limit değeri 10 bin TL üzeri ise 6 haneli random kredilD'si kredi teklifi olarak müşteriye sunulur. Aylık geliri 10 bin TL belirlenen müşterinin kredi miktarına yıllık geliri(120 bin)'nin %50'si(60 bin) atanır ve bu miktar müşteriye kampanya olarak sunulur. Eklenecek bu kredi kartı krediKartıEkle methodunda müşterinin Musteri sınıfındaki kredikartları arraylist'ine de eklenir ve random olarak üretilen kart numarası kullanıcıya bildirilir. krediKartıEkle methodunda kullanılabilirLimit methodu da çağırılarak müşterinin borcu varsa istemiş olduğu krediden bu borcu alınır ve geriye kalan kullanılabilir limit olarak müşteriye bildirir.

- Kredi kartı silmek istenirse Kisi sınıfında var olan kredi kartının krediKartıSil methodu çağırılarak aldığı müşteri parametresi sayesinde(ben ekledim) güncel borcu eğer 0'a eşitse müşterinin kredikartları arraylist'den bu krediyi siler. Güncel borcu 0'dan farklıysa öncelikle borç ödemesi yapın uyarısını ekrana yazdırır. (güncelBorc kişi sınıfında 600 tl int değişkeni olarak tanımlandı çünkü parametre olarak göndermem gerekiyordu.)
- Kredi kartı borç ödemesi yapılmak istenirse (300 TL üzerinden) Kisi sınıfında krediye ait krediKartıBorcOdeme metodu çağırılır ve parametre olarak alınan bu kredi kartının güncel borcundan 300 TL eksiltir. Aynı zamanda borç ödendikten sonra hesapta kalan miktar değeri getter ve setter metotlarıyla işlenip ekrana bastırılır.

YAPTIĞIM DEĞİŞİKLİKLER:

- ✓ **public** Kisi(){} Ekledim çünkü müşteri constructorunde parametrelili kişi constructorunu kullanmamam gerekiyordu. Döngüye giriyor gibi çıktı alıyordum.
- ✓ Kisi sınıfında limit, kartnumara,guncelborc ve alan tanımlayıp bunları gerekli constructorlerde gerçek değerlerine atadım.
- ✓ Musteri sınıfında kullanıcıdan aldığım değerleri belirlemek için alan,hesano,hesapBilgino değerleri oluşturdum.
- ✓ KrediKarti sınıfının krediKartiEkle metoduna ve kredikartı sil metoduna parametre olarak müşteriye ve kredikartı'nı gerekli müşteriye ve gerekli kredikartını eklemek için ben oluşturdum.
- ✓ krediKartiBorcOdeme (geriye değer döndürmemesi için void yaptım) ve krediOdeme metotlarını vadesiz hesap sınıfından alıp krediKarti sınıfına taşıdım. (Sizden izin almıştım)

KULLANDIĞIM EDITÖR: ECLIPSE

EKRAN ÇIKTILARI:

```
Problems @ Javadoc Declaration Search Console X
<terminated> MainBank [Java Application] C:\Users\ACER\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_17.0.5.v20221102-0933\jre\bin\javaw.exe (7 May 2023 2:
Adınızı giriniz:Abdulsamet
Soyadınızı giriniz:Uçar
Emailinizi giriniz:abdulsamettucar@gmail.com
Telefon numaranızı giriniz:0542 261 0207
TC kimlik numaranız:92638977118
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Banka Personeli      2-Müşteri
2
Müşteri numaranızı giriniz:
123
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Hesap ekle      2-Hesap sil
1
Lütfen hesap türü seçiniz
1-Maaş Hesabı      2-Normal Hesap
1
-----Hesap Bilgileri-----
Müşteri Numarası:123
Ad:Abdulsamet
Soyad:Uçar
Iban:TR1061870282697885898
Hesap Bilgisi:Maaş Hesabı
Açılış tarihi:Sun May 07 22:36:38 TRT 2023
Toplam Bakiye:6000.0
-----
Açmak istediğiniz hesap türünü giriniz
1-Vadeli hesap      2-yatırım hesabı      3-Vadesiz hesap:
1
Vadeli hesapta faiz oranıyla oluşan güncel toplam bakiye:18000
faiz oranı 0.2 olan Maaş-vadeli hesap başarıyla oluşturuldu
-----
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Kredi kartı ekle      2-Kredi kartı sil      3-Kredi kartı borç öde(300TL)
2
Lütfen öncelikle borç ödemesi yapınız
```

Kredi kartı silmek istenildiğinde 600 t1 borç olduğundan uyarı yapıyor.

```
Problems @ Javadoc Declaration Search Console X
<terminated> MainBank [Java Application] C:\Users\ACER\p2\pool\plugins\org.eclipse.justj.o
Adınızı giriniz:Abdulsamet
Soyadınızı giriniz:Uçar
Emailinizi giriniz:abdulsamettucar@gmail.com
Telefon numaranızı giriniz:0542 261 0207
TC kimlik numaranız:92941592192
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Banka Personeli      2-Müşteri
1
Personel numaranızı giriniz:36194
Müşterinizin numaranısını giriniz:
12315
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Hesap ekle      2-Hesap sil
2
Lütfen öncelikle bakiyenizi başka bir hesaba aktarınız!!!
toplam bakiye:6000.0TL
Müşteri adediniz:1
```

```
Problems @ Javadoc Declaration Search Console ×
<terminated> MainBank [Java Application] C:\Users\ACER\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_
1-Banka Personeli      2-Müşteri
2
Müşteri numaranızı giriniz:
21365
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Hesap ekle      2-Hesap sil
1
Lütfen hesap türü seçiniz
1-Maaş Hesabı      2-Normal Hesap
2
-----Hesap Bilgileri-----
Müşteri Numarası:21365
Ad:Abdulsamet
Soyad:Uçar
Iban:TR1017601039067464037
Hesap Bilgisi:Normal Hesap
Açılış tarihi:Sun May 07 23:17:13 TRT 2023
Toplam Bakiye:6000.0
-----
Açmak istediğiniz hesap türünü giriniz
1-Vadeli hesap      2-yatırım hesabı      3-Vadesiz hesap:
3
normal-vadesiz hesap başarıyla oluşturuldu
para transfer işleminde 8 tl kesinti olacaktır!!
200 TL Transfer sonrası toplam bakiye:5792.0
-----
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Kredi kartı ekle      2-Kredi kartı sil      3-Kredi kartı borç öde(300TL)
1
Limiti kaç olsun
1-3000TL(5000TL altı)      2-12.000TL(10.000TL üstü):
1
Güncel borcunuz:600.0
-----
91606no'lu Kredi kartınız başarıyla oluşturuldu
Kullanılabilir limitiniz:2400.0
```

kredi kartı Limiti 5 bin TL altı olduğundan güncel borcu yazdırıyor.

```
Problems @ Javadoc Declaration Search Console ×
<terminated> MainBank [Java Application] C:\Users\ACER\p2\pool\plugins\org.eclipse.justj.openjdk.hotspot.jre.full.win32.x86_64_1
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
a 1-Hesap ekle 2-Hesap sil
1
Lütfen hesap türü seçiniz
1-Maaş Hesabı 2-Normal Hesap
1
-----Hesap Bilgileri-----
Müşteri Numarası:165
Ad:Abdulsamet
Soyad:Uçar
Iban:TR1044745710346840831
Hesap Bilgisi:Maaş Hesabı
Açılış tarihi:Sun May 07 23:18:55 TRT 2023
Toplam Bakiye:6000.0
-----
Açmak istediğiniz hesap türünü giriniz
1-Vadeli hesap 2-yatırım hesabı 3-Vadesiz hesap:
2
Lütfen hesap türü seçiniz
1-Altın(kur:20) 2-Döviz(kur:10):
2
1-Para Ekle(100 Birim) 2-Para Boz(100 Birim)
1
Yeni Toplam Bakiye:7000.0TL
-----
Lütfen hangi alandan işlem yapmak istediğinizi seçiniz
1-Kredi kartı ekle 2-Kredi kartı sil 3-Kredi kartı borç öde(300TL)
1
Limiti kaç olsun
1-3000TL(5000TL altı) 2-12.000TL(10.000TL üstü):
2
limitiniz 10 bin tl üzeri olduğu için size özel 162437 ID'li kredi teklifimiz:
yıllık kazancınızın %50 si kadar:60000.0TL
-----
93110no'lu Kredi kartınız başarıyla oluşturuldu
Kullanılabilir limitiniz:11400.0
```

Kredi kartı limiti 10 bin TL üzeri için kampanya sunuluyor.