**Banka Otomasyon Uygulaması**

**Ad : Zehra**

**Soy ad : Aktürk**

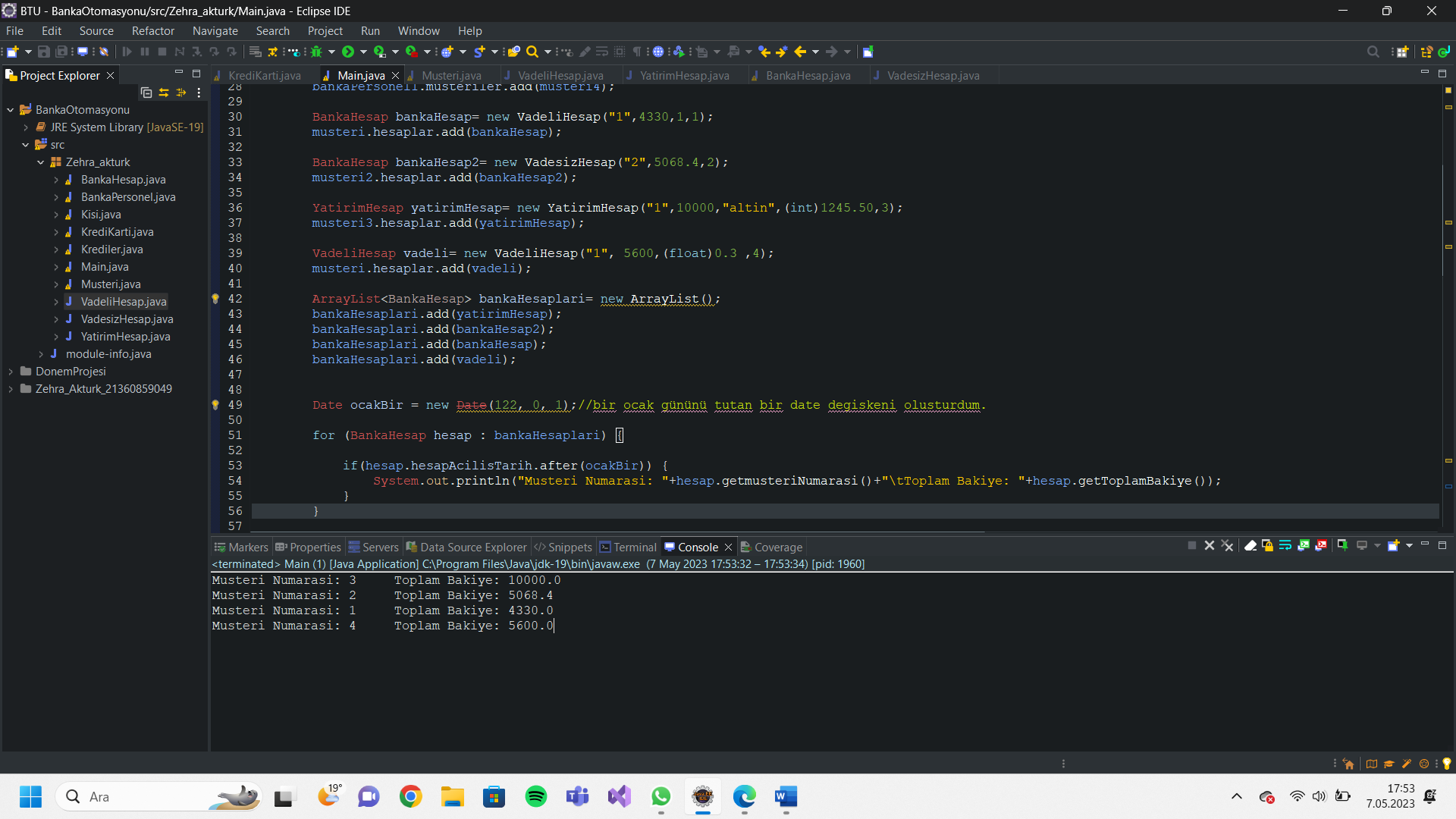
**Numara : 21360859049**

**Ders Kodu : BLM0121**

**ÖZET**

**Bu proje bir banka sistemini taklit eden bir otomasyon sistemidir. Bu projedeki amacımız Nesneye Yönelik Programlamadaki inheritance yapısını ve polimorfizm yapısını kullanabildiğimiz bir ortam yaratmak. Projenin büyük çoğunluğu UML diyagramını baz alınarak yapılmıştır. Class lar UML diyagramına göre oluşturulmuştur, getter ve setter methotları yazılmış, her sınıfa toString() methotu eklenmiştir. Ancak bazı methotlarda parametreler değiştirilmiştir. İstenilen maddeler yorumlanmış istenilen özellikler yerine getirilmiştir. Genel olarak bu projede bir banka personeli müşteri kaydedebilir ve müşterinin bilgilerini görüntüleyebilir. Bir müşteri yeni hesap açabilir ,kredi kartı oluşturabilir, kredi kartı silebilir ve kredi alabilir. Ayrıca konsoldan herhangi bir müşterinin hesap bilgileri görüntülenebilir.**

**PROJENİN ÇIKTISI**



**Bu çıktı 1 Ocak tan sonra açılan hesapların müşteri numaralarını ve toplam bakiyelerini yazdırmıştır. Bu çıktı için 4 müşteri nesnesi yaratılmıştır.**

metin, ekran görüntüsü, yazılım, ekran, görüntüleme içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Görüldüğü gibi 2 nesnenin değerleri random verilmiş,2 nesnenin değerleri girilmiştir.**

metin, ekran görüntüsü, yazılım, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu **Random atama yapmamızı sağlayan fonksiyon main sınıfının altında bu şekildedir. Bu fonksiyon kaç basamaklı random sayi istediğinizi parametre olarak alır. Ve o kadar basamaklı bir sayi geriye döndürür.**

metin, ekran görüntüsü, yazılım, bilgisayar içeren bir resim

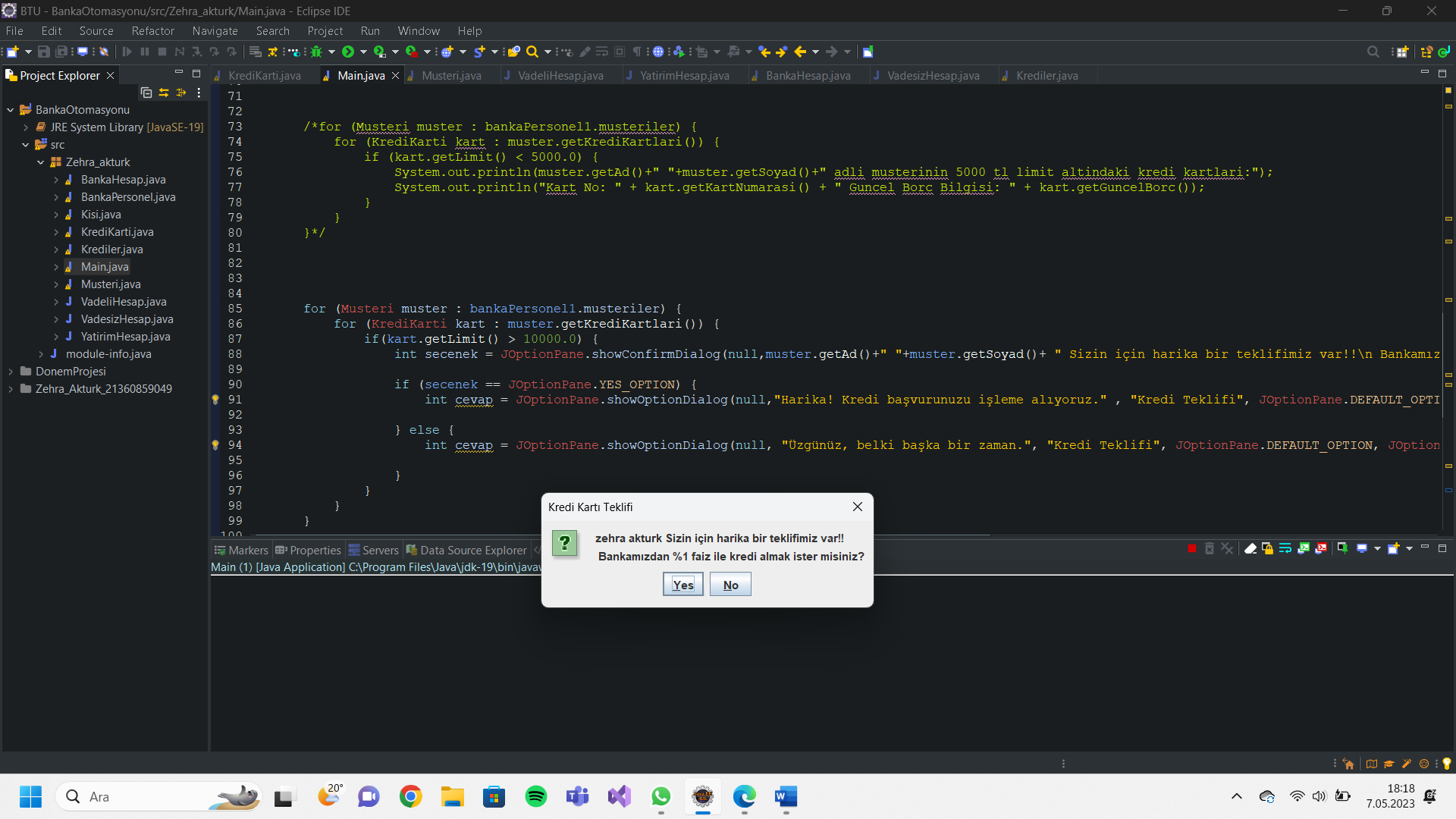
Açıklama otomatik olarak oluşturuldu**Bir tane banka personel oluşturulmuştur. Bilgileri girilmiştir ve resimde görülmektedir.**

**Ayrıca yukarıdaki resimde 4 banka hesabı oluşturulmuş ve ilgili müşterilerin hesaplar array listine eklenmiştir.**

metin, ekran görüntüsü, yazılım, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Yukarıdaki çıktıda kredi kart limiti 5000 tl nin altında olan müşterilerin güncel borç bilgisi yazdırılmıştır.**

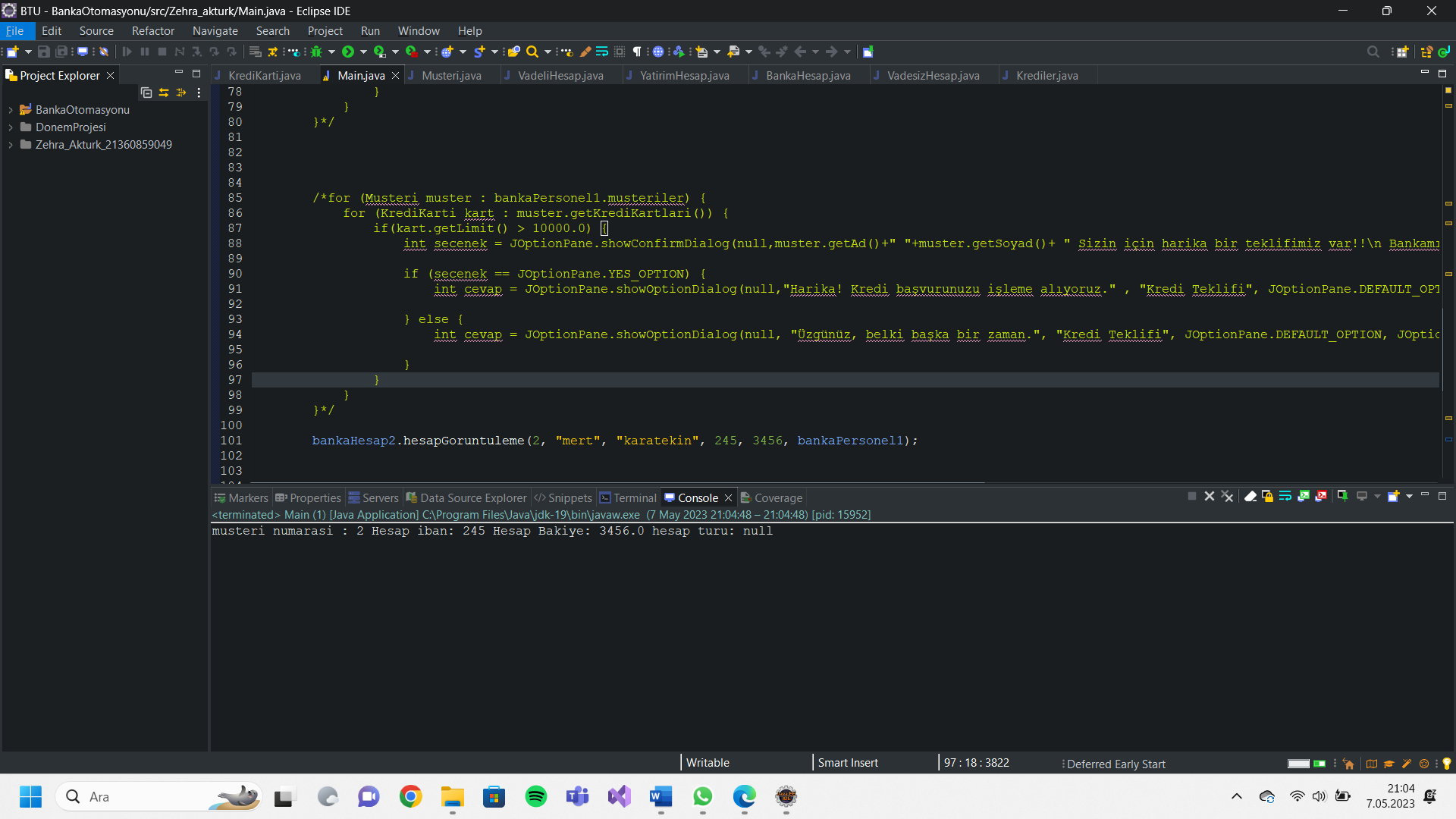


**Bu kısımda yes butonuna basılınca:**

metin, ekran görüntüsü, yazılım, bilgisayar içeren bir resim

Açıklama otomatik olarak oluşturuldu

**Yukarıdaki çıktıda kart limiti 10000 tl nin üzerinde olan müşterilere kredi teklif edilmiştir.**

 **Bu çıktıda ise hesap görüntüleme methodu çalıştırılmıştır.**

**PROJENİN OLUŞUM SÜRECİ**

**1)Kişi Sınıfı**

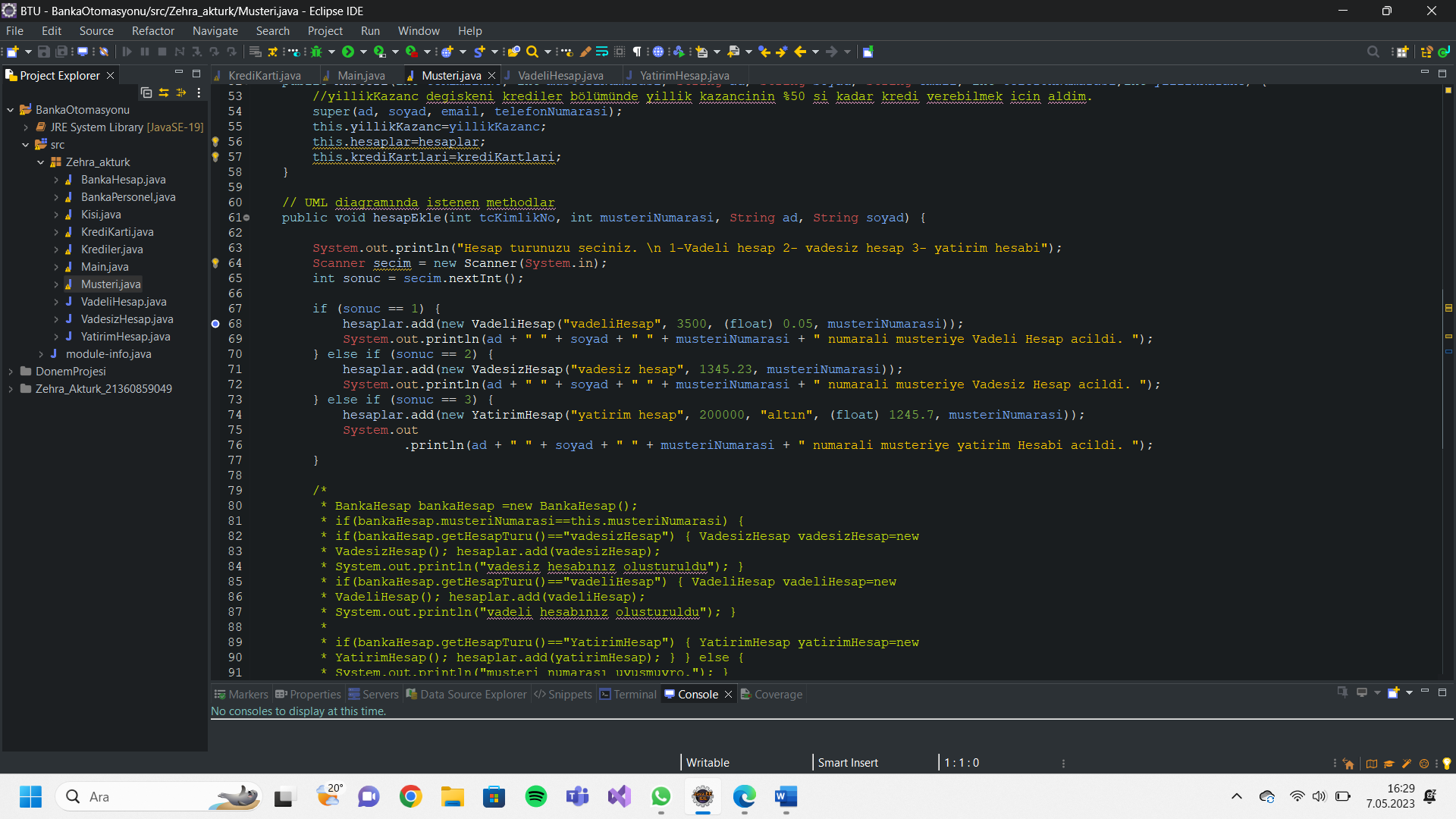
**Öncelikle UML diyagramına göre Kişi sınıfı oluşturulmuştur. Getter ve setter methotları yazılmıştır. Parametreli constructor UML diyagramındaki belirtilen özellikler çerçevesinde oluşturulmuştur. Parametreli constructor oluşturduğumuz için java default olarak parametresiz constructor oluşturmamıştır. Bundan dolayı parametresiz constructor oluşturulmuştur.**

**2) Müşteri Sınıfı**

**Müşteri sınıfı Kisi sınıfından miras alan sınıflardan bir tanesidir. Kişi sınıfından miras aldığı için parametreli constructor ında süper() methotu eklenmiştir.**

**i)hesapEkle Methotu**

**Bu sınıftaki hesapEkle methotu kullanıcının seçimine göre Vadeli ,Vadesiz ve Yatirim hesabı açabilmektedir.**



**ii) hesapSil methotu**

**Bu methot önce bütün hesapları tek tek dolaşır içerisinden müşteri numarası methota gönderilen müşteri numarası ile eşit olanı bulur ve bulunca buHesapMı isimli değişkeni true olarak atar. Sonrasında banka hesabında bakiye olup olmadığını kontrol eder. Hesapta bakiye yoksa o hesabı siler.**

**3)BankaPersonel Sınıfı**

**BankaPersonel sınıfı Kişi sınıfından miras almıştır. Bu sınıfta oluşturulan müşteriler, müşteriler isimli ArrayList e eklenmekte ve burada tutulmaktadır. Projemizin büyük çoğunluğu müşteri üzerinden devam etmektedir. Bundan dolayı bu sınıfta iş yapan bir methot yoktur.**

**4) Krediler Sınıfı**

**Bu sınıf Müşterinin kredi bilgilerinin tutmaktadır. Parametreli constructor ümüzde müşteri numarası kontrolü yapılmıştır. BankaPersonel sınıfında bulunan Müşteri nesnelerini tutan ArrayListten yararlanılmıştır. İlgili müşterinin yıllık kazancının %50 si kadar kredi miktari belirlenmiştir.**

**i)Kampanya Methotu**

**Bu methotda kullanıcının kart limiti güncel borcuna eşitse o müşteriye özel %0.1 faizle ihtiyaç kredisi fırsatı sunuluyor. Ve kart limiti 5000 tl ve üzerinde olan müşterilere özel her internet alışverişinde 100 bonus para fırsatı sunuluyor.**

**Eğer kişinin kart limiti 5000 tl altında ise bonus para kazabilmek için limit arttırma fırsatı sunuluyor.**

**5)KrediKarti Sınıfı**

**Bu sınıfta Kredi karti özelliklerini ve methodlarını tutmaktadır. Parametreli constructor kullanilabilirLimit değişkenine kullanilabilirLimit methotunu atanmıştır.**

**i) krediKartiEkle Methotu**

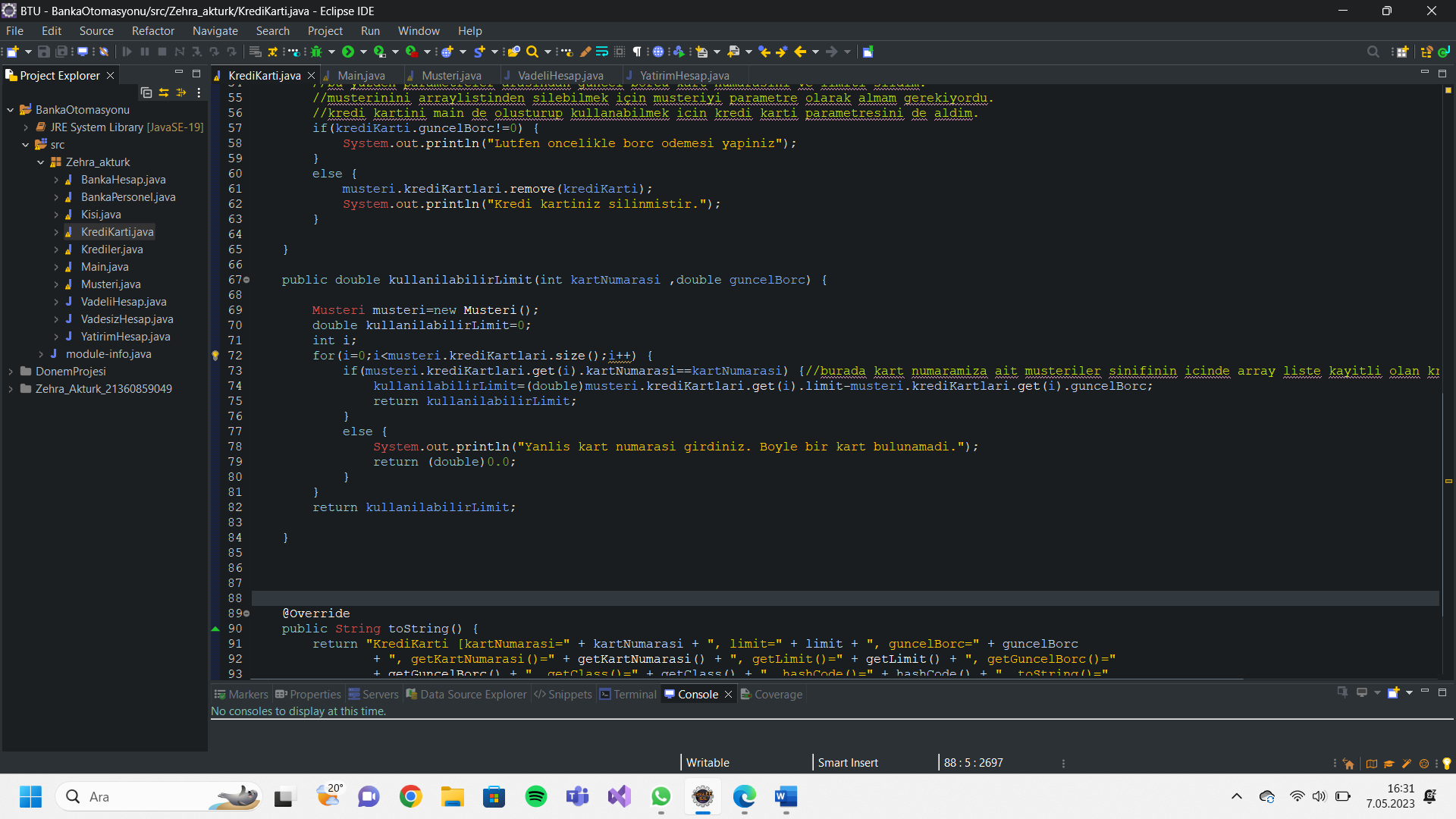
**Bu methot UML diyagramından farklı olarak Müşteri nesnesini ve KrediKarti nesnesini parametre olarak almaktadır. Böylece belirli müşteri nesnesinin krediKartlari ArrayList ine belirli kredi kartı başarı ile eklendi.**

**ii) krediKartiSil Methotu**

**Bu methot da UML diyagramından farklı olarak belirli kredi kartini belirli müşteri nesnesinden silebilmek için krediKarti ve Müşteri nesnesi parametre olarak alınmıştır. Güncel borç, limit ve kartNumarasi parametreleri silinmiştir. Bu parametrelere krediKarti nesnesi üzerinden ulaşılabilmektedir. Belirli kredi karti belirli müşterinin krediKartlari ArrayList inden başarılı bir şekilde silinmiştir.**

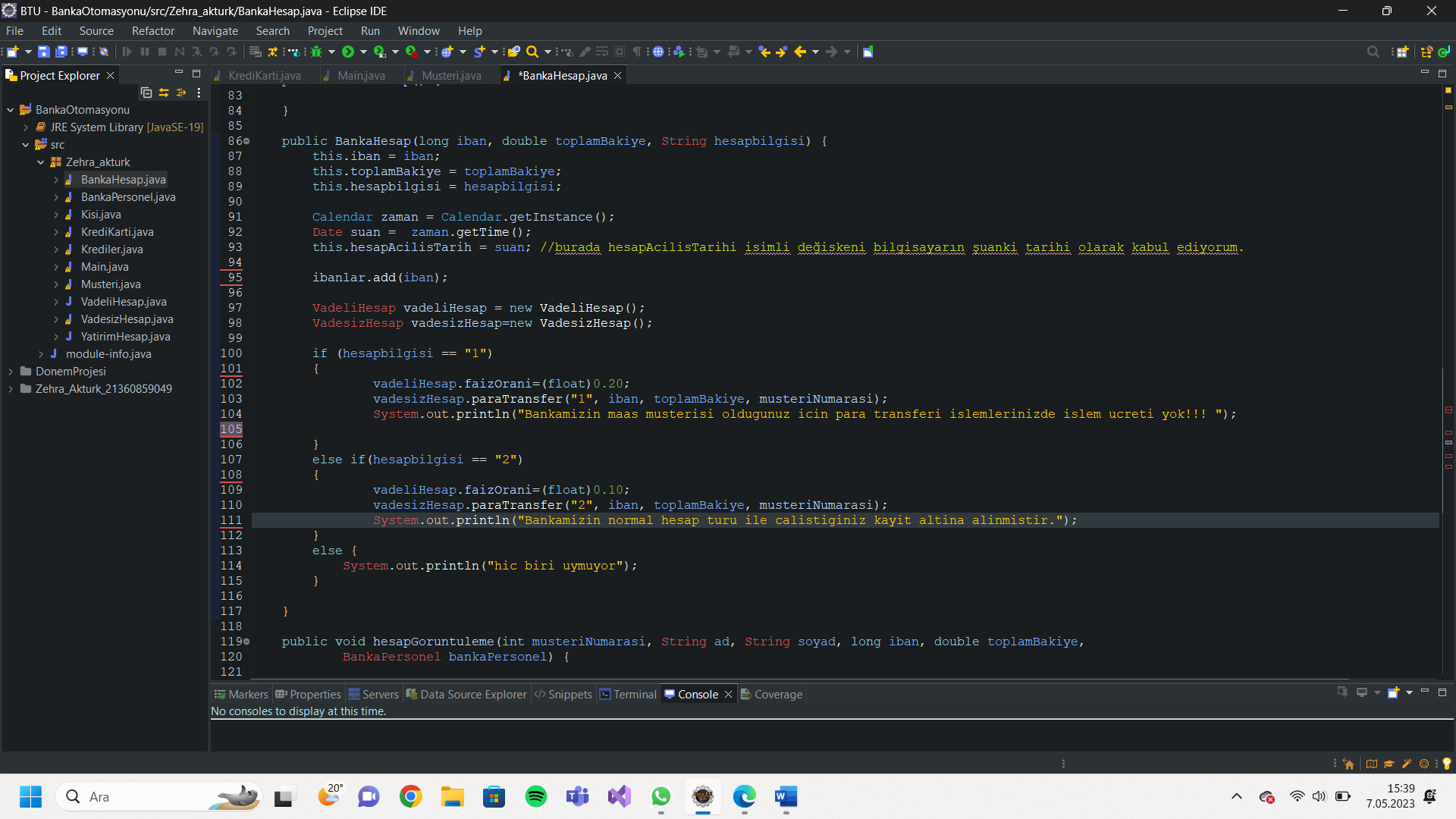
**iii)kullanilabilirLimit Methotu**

**Önce kartNumarasi adlı parametre müşteri nesnesinde bulunan krediKartlari ArrayListinden kontrol edilmiştir. Bu methot müşterinin kullanılabilir limitini döndürmektedir.**



**6)BankaHesap Sınıfı**

**Bu sınıf müşterilerin her hesap açış işleminde kullanılmaktadır. Bütün banka hesaplarının ata sınıfıdır. Parametreli constructor ın çağırıldığı tarih hesap açılış tarih olarak kabul edilir. Constructor da hesap bilgisine göre o müşteri için maaş müşterisi ise farklı normal müşteri ise farklı faiz oranı ve kesinti ücreti hesaplanmıştı.**



**i)hesapGoruntuleme Methotu**

**Bu methotta bankaPersonel nesnesi parametre olarak alınmıştır. Belli bir bankaPersonel müşterilerin numarasında göre hesap görüntüleme yapabilmektedir.**

**7) VadeliHesap Sınıfı**

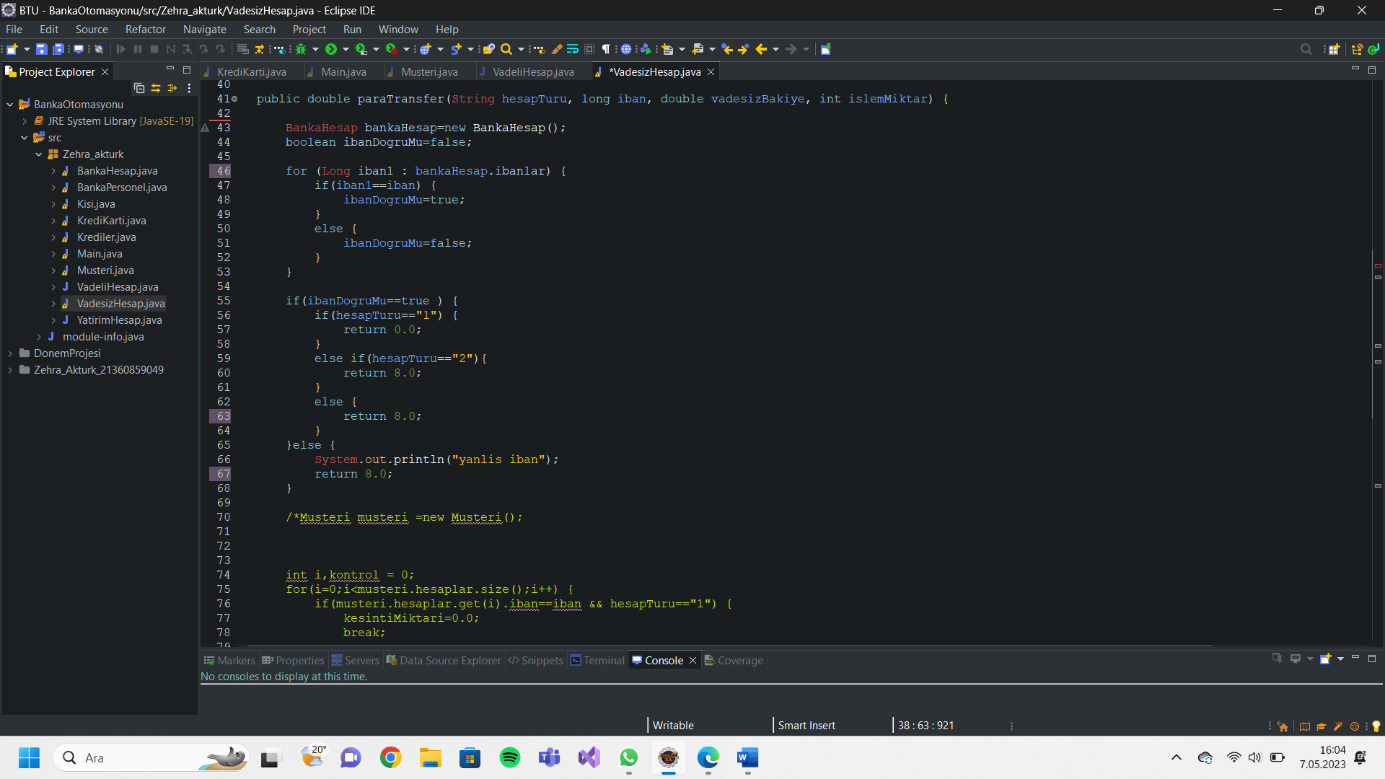
**VadeliHesap sınıfı müşterinin vadeli hesap bilgilerini tutan bir sınıftır. BankaHesap sınıfından miras almaktadır.**

**8)VadesizHesap Sınıfı**

**Bu sınıf müşterinin vadesiz hesap bilgilerini tutmaktadır. BankaHesap sınıfından miras almaktadır.**

**i)paraTransfer Methotu**

**Bu methotda önce verilen ibanın kontrolü yapılmaktadır. Sonrasında hesabın maaş hesabı ya da normal hesap olmasını sorgular. Maaş hesabı ise para transferi sırasında kesinti miktari 0, normal hesap ise kesinti miktarı 8 tl olarak atanmıştır.**



ii) krediKartiBorcOdeme Methotu

Bu methot da müşterinin kredi karti borç odemesi sağlanmıştır. Belli bir kredi karti ile işlem yapabilmek için kredi karti nesnesi parametre olarak alınmıştır.

iii)krediOdeme Methotu

Bu methot da müşterinin kredi odemesi yapabilmesi sağlanmıştır. Kredi nesnesi parametre olarak alınmıştır. Kredi miktarı ödenen miktar kadar azaltılmış, taksit miktarı bir taksit azaltılmıştır.

metin, bilgisayar, yazılım, ekran görüntüsü içeren bir resim

Açıklama otomatik olarak oluşturuldu

**9)YatirimHesap Sınıfı**

**Bu sınıf yatırım hesabı bilgilerini tutmaktadır. BankaHesap sınıfından miras almaktadır.**

**i)paraEkle Methotu**

**Bu methot belli bir kur şekilde belli miktar para eklememiz sağlanmıştır. Yatirilan miktar yatirimBakiye değişkenine eklenmiş, ve yatirimBakiye değişkeni döndürülmüştür.**

**ii)paraBoz Methotu**

**Bu methot girilen kur değerinde istenilen miktarda para bozma işlemi yapmaktadır. Bozulan miktar yatirimBakiye değişkeninden çıkartılmış ve yatirimBakiye değişkeni döndürülmüştür.**

**Not: Proje Eclipse de yazılmıştır.**