



Yazılım Mühendisliği

YZM-1106 Algoritma ve Programlama II Dersi (2018-2019)

Dönem Projesi / Banka Yazılım Otomasyonu

ÖĞRETİM ÜYESİ: DOÇ.DR.ERSİN ASLAN

ÖĞRENCİLER: 172805015-OZAN BATUHAN ÖZDAL

172805005-DENİZ UZUNDAĞ

172805008-ENDER GÜLGER

İÇİNDEKİLER

- PROJE ANALİZİ.....3
- TASARIM.....4
- PROGRAMCI KATALOĞU.....7
- KAYNAK KOD.....8
- KULLANICI KATALOĞU.....31

PROJE ANALİZİ

Banka Otomasyonu adı altında geliştireceğimiz

Proje kullanıcıya birçok işlem seçeneği sunmaktadır.

1)MÜŞTERİ EKLEME: İlk olarak banka sistemine müşteri eklemesi yapılmalıdır. Eklenen her müşteriye farklı rastgele müşteri numarası verilmektedir ve müşteri tipi seçilmektir (bireysel müşteri veya ticari müşteri).

2)HESAP AÇMA: Eklenen müşteriye birden fazla hesap açılabilir. Açılan hesaplara ayrı ayrı hesap numaraları verilmelidir.

3)PARA ÇEKME: Müşteri, istediği hesabından bakiyesi yetmiyorsa ek hesabından para çekebilmelidir. Eğer müşterimiz bireysel müşteri ise günlük limiti 750 TL, ticari müşteri ise 1500 TL çekebilmektedir. Hesapta yapılan her işlem hesap bilgilerinin bulunduğu dosyada da güncellenecektir.

4)PARA YATIRMA: Müşteri istediği hesabında istediği miktarda para yatırabilecektir. İşlem sonucunda mevcut bakiye güncellenecektir.

5)HESABA HAVALE: Müşterinin havale yapacağı hesabın mevcut olması gerekmektedir. Havale sonunda göndereninde alıcınındı hesap bakiyesi güncellenecektir. Ayrıca işlemi yapan müşteri bireysel müşteri ise %2 oranında havale ücreti kesilecektir, ticari müşteri ise havale ücreti kesilmeyecektir.

6)BANKA GELİR-GİDER RAPORU: Müşteri ana menü üzerinden rapor al kısmını seçerse bankanın gelir-giderleri hesaplanıp rapor.txt dosyasına yazılacaktır.

7)HESAP ÖZETİ: müşterinin seçtiği hesaba ilişkin belirtilen tarihler arasında yapılan işlemleri tarihleri ile birlikte dekont.txt dosyasına yazılacaktır.

8)HESAP KAPAMA: müşteri istediği hesabını kapatabilecektir. Kapatılacak hesabın bakiyesi 0 olmalıdır.

TASARIM

İlk olarak projede kullanılan structlar tanımlandı.

```
struct Hesaplar {  
    inthesapNo;  
    float hesapBakiye;  
    float hesapLimit;  
};  
  
struct Tarih {  
    int gun;  
    int ay;  
    int yil;  
};
```

```
struct timeinfo {  
    int gun  
};  
  
struct Musteriler {  
    char MusteriIsim[20];  
    char MusteriTelNo[20];  
    char musterAdres[20];  
    char musterKimlikNo[20];  
    char musterYas[3];  
    int musterTipi;  
    int MusteriNo;  
}Musteriler,MusteriHesap;
```

Daha sonra projemizi fonksiyonlara bölerek kodlamaya başladık.

Void Menu(): Bu fonksiyonun içinde ana menüyü yazdırıp switch-case yapısıyla kullanıcıya yapmak istediği işlemin seçimi yaptırıldı.

Void yeni_Musteri(): Bu fonksiyonda sisteme müşteri ekleme işlemi yapıldı. Random olarak müşteri ve hesap numarası atanıp kullanıcıdan pointer yardımıyla müşteri ismi, kimlik numarası, yaşı, adresi, telefon numarası, müşteri tipi bilgileri alındı. Müşteriler, hesaplar, yatırılan para, bireysel hesaplar, ticari hesaplar dosyaları oluşturuldu. Girilen müşteri Müşteriler dosyasına ve müşteri tipi seçimine göre (bireysel ya da ticari) ilgili dosyaya yazdırıldı ve dosyalar kapatıldı. Ve kullanıcıya müşteri başarıyla oluşturuldu mesajı verildikten sonra yapmak istenilen işlem soruldu.

yeni_Hesap(): Bu fonksiyon ile kullanıcıya ilk olarak hesap açmak istediği müşterinin numarasını soruyoruz ve girilen hesap numarası üzerinden müşterinin sistemde bulunup bulunmadığını kontrol ediyoruz. Müşteri bulunuyorsa yeni hesap için başlangıç bakiyesi bilgisini alıp hesap ve yatırılan para dosyalarına pointer yardımıyla yazma işlemi yapıp müşteriye çıkış ya da ana menüye dönmek isteyip istemediğini soruyoruz.

para_Cekme(): Kullanıcıdan müşteri numarası istiyoruz tekrar o müşterinin olup olmadığını kontrol ediyoruz, müşteri mevcut ise para çekmek istediği hesap numarasını girmesini ardından çekmek istediği miktarı girmesini bekliyoruz. Çekmek istediği bakiye seçtiği hesap bakiyesinden fazla ve müşteri tipine göre belirlenen limitten fazla ise müşteri ek hesabından para çekme işlemi gerçekleştirebilecektir. Eğer ek hesap limiti de yetersizse para çekemeyecektir. Limit onaylanırsa işlem gerçekleştirilecektir. İşlem sonucunda hesap bakiyeleri ve dosyalar güncellenecektir.

para_Yatirma(): Müşteriden para yatırmak istediği hesap numarasını istiyoruz hesabın olup olmadığını kontrol ediyoruz . Hesap mevcutsa yatıracağı miktarı girmesini bekliyoruz .yatırdığı miktarı mevcut hesap bakiyesinin üstüne ekleyip hesap bakiyesinde ve hesap dosyasında güncellemeleri yapıyoruz.

havale(): müşteri kendi hesap numarasını sonra yatıracağı miktarı giriyor, müşteri bireysel müşteri ise %2 havale ücreti kesiliyor. Girilen miktar hesabında bulunan bakiyeden fazla ise işlem yapılamıyor. Bakiye yeterli ise havale yapılacak hesap numarasının girilmesi bekleniyor ve girilen hesabın mevcut olup olmadığı kontrol edilip havale işlemi yapıp hesap bakiyeleri ve dosyalarda güncelleme yapılır

rapor(): While döngüleri ile bankadan çekilen toplam para ,yatırılan toplam para, hesaplanıp hem ekrana hem de rapor.txt isimli dosyaya yazılacaktır.

hesap_Ozeti(): kullanıcıdan hesap özetini görmek istediği hesap numarasını ve tarih aralığını girmesini bekliyoruz . Belirtilen tarihlerde yapılan para çekme, yatırma, havale işlemleri rapor.txt dosyasına ve ekrana yazılıyor.

hesap_Kapat(): Kullanıcıdan kapatmak istediği hesap numarasını girmesini bekliyoruz. Eğer girdiği hesap bulunuyorsa ve bakiyesi sıfırdan büyükse müşteri bulunan bakiyeyi çekme işlemi yapacak ve hesap öyle kapatılacaktır.

Oluşturulan dosyalar:

Musteriler.txt

Hesaplar.txt

bireyselHesaplar.txt

ticariHesaplar.txt

CekilenPara.txt

YatirilanPara.txt

yapilanHavale.txt

dekont.txt

Rapor.txt

PROGRAMCI KATALOĞU

- **Analiz:** Analiz için harcanan süre 1 hafta.
- **Tasarım:** Tasarım için harcanan süre 1-4 gün
- **Gerçekleştirim:** Gerçekleştirim için geçen süre 1-2 hafa •
- **Test:** Projenin test edilmesi için harcanan süre 2-3 saat
- **Raporlama:** Raporlama için geçen süre 2 saat

GÖREV DAĞILIMI

- **OZAN BATUHAN ÖZDAL:** Yeni müşteri ekleme, hesap açma, para yatırma, para çekme işlemleri.
- **DENİZ UZUNDAĞ:** Hesaba havale, banka gelir-gider raporu, test ve raporlama işlemleri.
- **ENDER GÜLGER:** Hesap özeti, Hesap kapama, test işlemleri.

KAYNAK KOD

```
#include <stdio.h>
#include <stdlib.h>
#include <time.h>

struct Hesaplar {

    int hesapNo;
    float hesapBakiye;
    float hesapLimit;

};

struct Tarih {

    int gun;
    int ay;
    int yil;

};

struct timeinfo {

    int gun

};

struct Musteriler {

    char MusteriIsim[20];
    char MusteriTelNo[20];
    char musterAdres[20];
```



```

char musterikimlikNo[20];
char musteriyas[3];
int musteritipi;
int MusteriNo;

    }Musteriler,MusteriHesap;

// ANA MENU
void Menu()
{
    int secim;
    printf("\n\n\t\t\tBanka Otomasyon Yazilimi");
    printf("\n\n\n\t\t\t\t\t\xB2\xB2\xB2\xB2\xB2\xB2\xB2 Ana Menu
\xB2\xB2\xB2\xB2\xB2\xB2\xB2");
    printf("\n\n\t\t\t1.Yeni Musteri Olustur\n\t\t\t2.Hesap
Acma\n\t\t\t3.Hesaptan Para Cekme\n\t\t\t4.Hesaba Para
Yatirma\n\t\t\t5.Baska Hesaba Havale\n\t\t\t6.Banka Gelir-Gider
Raporu\n\t\t\t7.Hesap Ozeti\n\t\t\t8.Hesap
Kapama\n\t\t\t9.Cikis\n\n\n\n\n\n\t\t\t Seciminizi Yapin:");
    scanf("%d",&secim);

    switch(secim)
    {
        case 1:yeni_Musteri();
        break;
        case 2:yeni_Hesap();
        break;
        case 3:para_Cekme();
        break;
        case 4:para_Yatirma();
        break;
        case 5:havale();
        break;
        case 6:rapor();
        break;
        case 7:hesap_Ozeti();
        break;
    }
}

```

```

    case 8: hesap_Kapat();
    break;
    case 0: close();
    break;
    default:
    {
        printf("1 ile 8 arasi bir sayi giriniz.Cikmak icin 0:");
        system("cls");
        Menu();
    }
}
}

void yeni_Musteri()
{
    struct Musteriler *pMusteri,sMusteriler;
    struct Hesaplar *pHesap,sHesap;
    struct Tarih *pTarih,sTarih;
    memset(&sMusteriler, 0, sizeof(sMusteriler));
    memset(&sHesap, 0, sizeof(sHesap));
    memset(&sTarih, 0, sizeof(sTarih));

    pMusteri = &sMusteriler;
    pHesap = &sHesap;
    pTarih = &sTarih;

    int cikis;
    system("cls");
    int secim,RmusteriNo,RhesapNo;
    time_t rawtime;
    struct tm *timeinfo;
    time ( &rawtime );
    timeinfo = localtime ( &rawtime );
    // printf("Ay:%d",timeinfo->tm_mon + 1);
    // printf("Yil:%d",timeinfo->tm_year + 1900);
    FILE *Musteri,*hesapFile,*yatirilanFile;
    Musteri= fopen("Musteriler.txt","a+");
    hesapFile = fopen("Hesaplar.txt","a+");
    yatirilanFile = fopen("YatirilanPara.txt","a+");
    srand(time(NULL));

```

```

RmusteriNo = rand() % 100;
RhesapNo = rand() % 1000;
printf("MusteriIlk:%d\n",RmusteriNo);
printf("HesapIlk:%d\n",RhesapNo);
while(fscanf(Musteri,"%d %d %s %s %s %s %s %d %f %d %d
%d",&pMusteri->MusteriNo,&pHesap->hesapNo,pMusteri-
>MusteriIsim,pMusteri->musteriKimlikNo,pMusteri->musteriYas,pMusteri-
>musteriAdres,pMusteri->MusteriTelNo,&pMusteri->musteriTipi,&pHesap-
>hesapBakiye,&pTarih->gun,&pTarih->ay,&pTarih->yil)!=EOF)
{

    while(RmusteriNo == pMusteri->MusteriNo)
    {
        srand(time(0));
        RmusteriNo = rand() % 100;
        printf("Musteri:%d\n",RmusteriNo);
        rewind(Musteri);
    }
}
printf("2.while");
while(fscanf(hesapFile,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{
    while(RhesapNo == pHesap->hesapNo)
    {
        srand(time(0));
        RhesapNo = rand() % 1000;
        printf("Hesap:%d\n",RhesapNo);
        rewind(hesapFile);
    }
}

pMusteri->MusteriNo = RmusteriNo;
pHesap->hesapNo = RhesapNo;
printf("%Olusan Musteri Numaraniz:%d\n",pMusteri->MusteriNo);
printf("%Olusan Hesap Numaraniz:%d\n",pHesap->hesapNo);
printf("\nMusterinin ismini giriniz:");
scanf("%s",pMusteri->MusteriIsim);
printf("\nMusterinin kimlik Numarasini giriniz:");

```

```

scanf("%s",pMusteri->musteriKimlikNo);
printf("\nMusterinin Yasini giriniz:");
scanf("%s",pMusteri->musteriYas);
printf("\nMusterinin Adresini giriniz:");
scanf("%s",pMusteri->musteriAdres);
printf("\nMusterinin telefon Numarasini giriniz:");
scanf("%s",pMusteri->MusteriTelNo);
printf("\nMusteri Tipini Giriniz Bireysel:1/Ticari:2:");
scanf("%d",&pMusteri->musteriTipi);
printf("\nHesap Baslangic Bakiyesini Giriniz:");
scanf("%f",&pHesap->hesapBakiye);

fprintf(Musteri,"%d %d %s %s %s %s %s %d %f %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pMusteri->MusteriIsim,pMusteri-
>musteriKimlikNo,pMusteri->musteriYas,pMusteri->musteriAdres,pMusteri-
>MusteriTelNo,pMusteri->musteriTipi,pHesap->hesapBakiye,timeinfo -
>tm_mday,timeinfo->tm_mon + 1,timeinfo->tm_year + 1900);
fclose(Musteri);
fprintf(hesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
fclose(hesapFile);
fprintf(yatirilanFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
fclose(yatirilanFile);
FILE *ticariFile,*bireyselFile;
ticariFile= fopen("ticariHesaplar.txt","a+");
bireyselFile= fopen("bireyselHesaplar.txt","a+");
if(pMusteri->musteriTipi==1)
{
    fprintf(bireyselFile,"%d %d %s %s %s %s %s %d %f %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pMusteri-
>MusteriIsim,pMusteri->musteriKimlikNo,pMusteri->musteriYas,pMusteri-
>musteriAdres,pMusteri->MusteriTelNo,pMusteri->musteriTipi,pHesap-
>hesapBakiye,timeinfo ->tm_mday,timeinfo->tm_mon + 1,timeinfo->tm_year +
1900);

}
if(pMusteri->musteriTipi==2)

```

```

    {
        fprintf(ticariFile,"%d %d %s %s %s %s %s %d %f %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pMusteri-
>MusteriIsim,pMusteri->musteriKimlikNo,pMusteri->musteriYas,pMusteri-
>musteriAdres,pMusteri->MusteriTelNo,pMusteri->musteriTipi,pHesap-
>hesapBakiye,timeinfo ->tm_mday,timeinfo->tm_mon + 1,timeinfo->tm_year +
1900);
    }
    fclose(ticariFile);
    fclose(bireyselFile);
    system("cls");
    printf("\nMusteri Basariyla Olusturuldu!");

    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak icin:0'a
Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();
    else if(cikis==0)
        close();

}

yeni_Hesap()
{
    struct Musteriler *pMusteri,sMusteriler;
    struct Hesaplar *pHesap,sHesap;
    struct Tarih *pTarih,sTarih;
    memset(&sMusteriler, 0, sizeof(sMusteriler));
    memset(&sHesap, 0, sizeof(sHesap));
    memset(&sTarih, 0, sizeof(sTarih));

    pMusteri = &sMusteriler;
    pHesap = &sHesap;
    pTarih = &sTarih;

    system("cls");
    int musterino;
    int rhesapno;
    int kontrol=0;

```

```

int cikis;
int musteriT;
time_t rawtime;
struct tm * timeinfo;
time ( &rawtime );
timeinfo = localtime ( &rawtime );
printf("\n \n \tYeni Hesap Acmak Istediginiz Musterinin Numarasini
Girin:");
scanf("%d",&musteriNo);
FILE *hesapFile,*yatirilanFile;
hesapFile = fopen("Hesaplar.txt","a+");
yatirilanFile = fopen("YatirilanPara.txt","a+");
srand(time(NULL));
RhesapNo = rand() % 1000;
printf("HesapIlk:%d\n",RhesapNo);
while(fscanf(hesapFile,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{
    if(musteriNo==pMusteri->MusteriNo)
    {
        kontrol++;
        musteriT = pMusteri->MusteriNo;
    }
    while(RhesapNo == pHesap->hesapNo)
    {
        srand(time(0));
        RhesapNo = rand() % 1000;
        printf("Hesap:%d\n",RhesapNo);
        rewind(hesapFile);
    }
}

if(kontrol == 0)
{
    printf("\tGirdiginiz Musteri Numarasi Sistemde
Bulunmamaktadir...\n");
    printf("\n\n\n\t\tAna menuye donmek icin:1");
    scanf("%d",&cikis);
    system("cls");
}

```

```

    if (cikis==1)
        Menu();

    }

    pHesap->hesapNo = RhesapNo;
    pMusteri->MusteriNo= musterino;
    printf("Baslangic Bakiyesini Giriniz:");
    scanf("%f",&pHesap->hesapBakiye);
    fprintf(hesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    fclose(hesapFile);
    fprintf(yatirilanFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    fclose(yatirilanFile);
    system("cls");
    printf("Hesabiniz Basariyla olusturuldu");
    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak icin:0'a
Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();
    else if(cikis==0)
        close();

    }

para_Cekme()
{
    struct Musteriler *pMusteri,sMusteriler;
    struct Hesaplar *pHesap,sHesap;
    struct Tarih *pTarih,sTarih;
    memset(&sMusteriler, 0, sizeof(sMusteriler));
    memset(&sHesap, 0, sizeof(sHesap));
    memset(&sTarih, 0, sizeof(sTarih));

```

```

pMusteri = &sMusteriler;
pHesap = &sHesap;
pTarih = &sTarih;

system("cls");
int hesapNo2, hesapNo1, MusteriNo, cikis, kontrol=0;
float cekMiktar, yeniBakiye, ilkBakiye, kalanPara, hesapLimiti, kontrolLimit;
float toplamBakiye=0;
time_t rawtime;
struct tm * timeinfo;
time ( &rawtime );
timeinfo = localtime ( &rawtime );

printf("\tMusteri Numaranizi Giriniz:");
scanf("%d",&MusteriNo);
FILE
*hesapFile,*yeniHesapFile,*hesapFile2,*hesapFile3,*yeniHesapFile2,*cekilenH
esap;
hesapFile = fopen("Hesaplar.txt","a+");
hesapFile2 = fopen("Hesaplar.txt","r+");
yeniHesapFile = fopen("hesap.txt","a+");
cekilenHesap = fopen("CekilenPara.txt","a+");

while(fscanf(hesapFile2,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{
    if(MusteriNo == pMusteri->MusteriNo)
    {
        printf("Hesaplariniz: \t%d %d %f %f \n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap->hesapLimit);
        toplamBakiye += pHesap->hesapBakiye;
        kontrol++;
    }
}
if(kontrol ==0)
{
    printf("Girdiđiniz Musteri numarasi sistemde bulunmamaktadir...");
}

```



```

        printf("\n\n\n\t\tAna menuye donmek icin:1 - Tekrardan para
cekme icin 0'a Basiniz:");
        scanf("%d",&cikis);
        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            para_Cekme();

    }
    printf("\tPara Cekmek Istediginiz Hesap Numarasini Giriniz:");
    scanf("%d",&hesapNo2);
    printf("Hesabinizdaki Toplam Para:%f\n",toplamBakiye);
    printf("Cekmek istediginiz miktari giriniz:");
    scanf("%f",&cekMiktar);

    if(cekMiktar>toplamBakiye)
    {
        printf("Hesaplardaki Miktardan Fazla Deger Girdiniz.../");
        printf("\n\n\n\t\tAna menuye donmek icin:1 Tekrardan Para
cekme icin:0'a Basiniz:");
        scanf("%d",&cikis);
        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            para_Cekme();

    }

    while(fscanf(hesapFile,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
    {
        if(hesapNo2==pHesap->hesapNo)
        {

            if(cekMiktar>pHesap->hesapBakiye)
            {
                printf("Para cekmek istediginiz ek hesabinizi giriniz:");
                scanf("%d",&hesapNo1);

```

```

    }

    if(timeinfo ->tm_mday == pTarih->gun && timeinfo->tm_mon + 1 ==
pTarih->ay && timeinfo->tm_year + 1900 == pTarih->yil)
    {

    }
    else
    {
        pHesap->hesapLimit = 0;
    }

    pHesap->hesapLimit += cekMiktar;

    if(pMusteri->musteriTipi == 1)
    {

        if(cekMiktar>750)
        {

            printf("Günlük para çekme limitimiz bireysel müşteriler için
750' TL...\n");
            printf("\n\n\n\t\t\tAna menüye dönmek için:1 Tekrardan Para
çekmek için:0'a Basınız:");
            scanf("%d",&cikis);
            system("cls");
            if (cikis==1)
                Menu();
            else if(cikis==0)
                para_Cekme();
        }

        if(pHesap->hesapLimit>750)
        {
            float cekUcret;
            pHesap->hesapLimit -= cekMiktar;
            printf("Gunluk para çekme limitini astınız
Limitiniz:750' TL\n");
            printf("Sizin Bugün Çektiginiz Miktar:%f\n",pHesap-
>hesapLimit);

```

```

printf("Gunluk para cekme limitimiz bireysel musteriler icin
750' TL...");
printf("\n\n\n\t\t\tAna menuye donmek icin:1 Tekrardan Para
cekmek icin:0'a Basiniz:");
scanf("%d",&cikis);
system("cls");
if (cikis==1)
Menu();
else if(cikis==0)
para_Cekme();

}

}
if(pMusteri->musteriTipi == 2)
{
if(cekMiktar>1500)
{
printf("Gunluk para cekme limitimiz Ticari musteriler icin
1500' TL...");

printf("\n\n\n\t\t\tAna menuye donmek icin:1 Tekrardan Para
cekmek icin:0'a Basiniz:");
scanf("%d",&cikis);
system("cls");
if (cikis==1)
Menu();
else if(cikis==0)
para_Cekme();
}

if(pHesap->hesapLimit>1500)
{
printf("Gunluk para cekme limitimiz Ticari musteriler icin
1500' TL...\n");
printf("Sizin Bugun Cektiginiz Miktar:%f",pHesap-
>hesapLimit-cekMiktar);

printf("\n\n\n\t\t\tAna menuye donmek icin:1 Tekrardan Para
cekmek icin:0'a Basiniz:");
scanf("%d",&cikis);

```

```

        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            para_Cekme();
    }

}

if(cekMiktar<=pHesap->hesapBakiye)
{

    printf("%f: Para Cektiniz\n",cekMiktar);
    pHesap->hesapBakiye -= cekMiktar;
    printf("%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    fprintf(yeniHesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    printf("Para Cekme Islemi Basarili\n");
    printf("Hesabinizdaki Yeni Para:%f",pHesap->hesapBakiye);

    fprintf(cekilenHesap,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,cekMiktar,pHesap->hesapLimit,pMusteri-
>musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon + 1,timeinfo->tm_year +
1900);

    printf("\n\n\n\t\tAna menuye donmek icin:1 Tekrardan Para
cekmek icin:0'a Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();
    else if(cikis==0)
        para_Cekme();

}

if(cekMiktar>pHesap->hesapBakiye && cekMiktar<toplamBakiye)

```

```

        {
            ilkBakiye = pHesap->hesapBakiye;
            printf("%f: Para Cektiniz\n",pHesap->hesapBakiye);
            pHesap->hesapBakiye = 0;
            printf("%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
            fprintf(yeniHesapFile,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap-
>hesapBakiye,ilkBakiye,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo-
>tm_mon + 1,timeinfo->tm_year + 1900);
            fprintf(cekilenHesap,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,ilkBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
        }
    }
    else
    {
        fprintf(yeniHesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
    }
}

fclose(hesapFile);
fclose(yeniHesapFile);
fclose(hesapFile2);

remove("Hesaplar.txt");
rename("hesap.txt","Hesaplar.txt");

yeniHesapFile2 = fopen("hesap.txt","a+");
hesapFile3 = fopen("Hesaplar.txt","r+");
while(fscanf(hesapFile3,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{

```

```

        if(hesapNo1==pHesap->hesapNo)
        {
            if(timeinfo ->tm_mday == pTarih->gun && timeinfo->tm_mon
+ 1 == pTarih->ay && timeinfo->tm_year + 1900 == pTarih->yil)
            {

            }
            else
            {
                pHesap->hesapLimit = 0;
            }

            kalanPara = cekMiktar - ilkBakiye;
            pHesap->hesapBakiye-= kalanPara;
            pHesap->hesapLimit += kalanPara;

            if(pMusteri->musteriTipi == 1)
        {
            printf("tip2 kontrole girdi\n");

            if(pHesap->hesapLimit>750)
            {
                printf("Gunluk para çekme limitini astiniz
Limitiniz:750'TL\n");
                printf("Sizin Bugün Cektiginiz Miktar:%f",pHesap-
>hesapLimit );

                printf("\n\n\n\t\tAna menuye donmek icin:1 Tekrardan
Para cekmek icin:0'a Basiniz:");
                scanf("%d",&cikis);
                system("cls");
                if (cikis==1)
                    Menu();
                else if(cikis==0)
                    para_Cekme();

            }

        }

        if(pMusteri->musteriTipi == 2)
        {

```

```

        if(pHesap->hesapLimit>1500)
        {
            printf("Günlük para cekme limitimiz Ticari müsteriler için
1500' TL...");
            printf("Sizin Bugün Cektiginiz Miktar:%f",pHesap-
>hesapLimit );
            printf("\n\n\n\t\tAna menüye donmek için:1 Tekrardan Para
cekmek için:0'a Basiniz:");
            scanf("%d",&cikis);
            system("cls");
            if (cikis==1)
                Menu();
            else if(cikis==0)
                para_Cekme();

        }

    }

    printf("%f: Para Cektiniz\n",kalanPara);
    printf("%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    fprintf(yeniHesapFile2,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);
    printf("Para Cekme Islemi Basarili\n");
    printf("Hesabinizdaki Yeni Para:%f",pHesap->hesapBakiye);

    fprintf(yeniHesapFile,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap-
>hesapBakiye,kalanPara,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo-
>tm_mon + 1,timeinfo->tm_year + 1900);
    fprintf(cekilenHesap,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,kalanPara,pHesap-
>hesapLimit,pMusteri->musteriTipi,timeinfo->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);

    }
else

```

```

        {
            fprintf(yeniHesapFile2,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
        }

    }

    fclose(hesapFile3);
    fclose(yeniHesapFile2);
    fclose(cekilenHesap);

    remove("Hesaplar.txt");
    rename("hesap.txt","Hesaplar.txt");
    printf("\n\n\n\t\tAna menuye donmek icin:1 Tekrardan Para cekmek icin:0'a
Basiniz:");

        scanf("%d",&cikis);
        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            para_Cekme();

    }

    para_Yatirma()
    {
        struct Musteriler *pMusteri,sMusteriler;
        struct Hesaplar *pHesap,sHesap;
        struct Tarih *pTarih,sTarih;
        memset(&sMusteriler, 0, sizeof(sMusteriler));
        memset(&sHesap, 0, sizeof(sHesap));
        memset(&sTarih, 0, sizeof(sTarih));

        pMusteri = &sMusteriler;
        pHesap = &sHesap;
        pTarih = &sTarih;

        system("cls");
        int hesapNo,cikis,kontrol=0;
        float paraYatir;
        time_t rawtime;

```



```

struct tm * timeinfo;
time ( &rawtime );
timeinfo = localtime ( &rawtime );
FILE *hesapFile,*yeniHesapFile,*yatirilanHesap;
hesapFile = fopen("Hesaplar.txt","a+");
yeniHesapFile = fopen("hesap.txt","a+");
yatirilanHesap = fopen("YatirilanPara.txt","a+");
printf("Para yatirmek istediginiz hesap numarasini giriniz:");
scanf("%d",&hesapNo);

while(fscanf(hesapFile,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{

    if(hesapNo==pHesap->hesapNo)
    {
        printf("Yatiracaginiz miktari giriniz:");
        scanf("%f",&paraYatir);

        printf("Hesabinizdaki Eski Para:%f\n",pHesap->hesapBakiye);

        pHesap->hesapBakiye += paraYatir;

        fprintf(yeniHesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
        printf("Para Yatirma Islemi Basarili\n");
        printf("Hesabinizdaki Yeni Para:%f",pHesap->hesapBakiye);
        kontrol++;
        fprintf(yatirilanHesap,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,paraYatir,pHesap->hesapLimit,pMusteri-
>musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);

    }

    else{

```

```

        fprintf(yeniHesapFile,"%d %d %f %f %d %d %d %d\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
    }
}
if(kontrol ==0)
{
    printf("Girdiginiz hesap numarasinda hesap yoktur...");
    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak
icin:0'a Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();

}

fclose(hesapFile);
fclose(yeniHesapFile);

remove("Hesaplar.txt");
rename("hesap.txt","Hesaplar.txt");

    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak
icin:0'a Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();
    else if(cikis==0)
        close();

}
havale()
{

    struct Musteriler *pMusteri,sMusteriler;
    struct Hesaplar *pHesap,sHesap;
    struct Tarih *pTarih,sTarih;
    memset(&sMusteriler, 0, sizeof(sMusteriler));
    memset(&sHesap, 0, sizeof(sHesap));

```

```

memset(&sTarih, 0, sizeof(sTarih));

pMusteri = &sMusteriler;
pHesap = &sHesap;
pTarih = &sTarih;

system("cls");
int hesapNo,havaleHesapNo,cikis,kontrol=0,musteriT;
float paraYatir,havalePara;
time_t rawtime;
struct tm * timeinfo;
time ( &rawtime );
timeinfo = localtime ( &rawtime );
FILE
*hesapFile,*yeniHesapFile,*hesapFile2,*yeniHesapFile2,*havaleHesap;
hesapFile = fopen("Hesaplar.txt","a+");
yeniHesapFile = fopen("hesap.txt","a+");
printf("Kendi Hesap Numaranizi Giriniz:");
scanf("%d",&hesapNo);

while(fscanf(hesapFile,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
{
    if(hesapNo == pHesap->hesapNo)
    {
        kontrol++;
        musteriT= pMusteri->musteriTipi;
        printf("Hesabiniz: \t%d %d %f\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye);
        printf("Yatiracaginiz miktar giriniz:");
        scanf("%f",&paraYatir);

        havalePara = paraYatir;

        if(musteriT == 1)
        {

```

```

        paraYatir += (paraYatir / 100) * 2 ;
    }

    if(paraYatir>pHesap->hesapBakiye)
    {
        printf("Hesabinizda Bulunan miktardan fazla bir deger
girdiniz...");

        printf("\n\n\n\t\tAna menuye donmek icin:1");
        scanf("%d",&cikis);
        system("cls");
        if (cikis==1)
            Menu();

    }
    else
    {
        pHesap->hesapBakiye -= paraYatir;

        fprintf(yeniHesapFile,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);

    }
    else
    {
        fprintf(yeniHesapFile,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
    }

}

if(kontrol ==0)
{
    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak
icin:0'a Basiniz:");
    scanf("%d",&cikis);

```

```

        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            close();

    }

    fclose(hesapFile);
    fclose(yeniHesapFile);

    remove("Hesaplar.txt");
    rename("hesap.txt","Hesaplar.txt");

    printf("Havale Yapmak istediginiz hesabin numarasini giriniz:");
    scanf("%d",&havaleHesapNo);
    hesapFile2 = fopen("Hesaplar.txt","a+");
    yeniHesapFile2 = fopen("hesap.txt","a+");
    havaleHesap= fopen("yapilanHavale.txt","a+");

    while(fscanf(hesapFile2,"%d %d %f %f %d %d %d %d",&pMusteri-
>MusteriNo,&pHesap->hesapNo,&pHesap->hesapBakiye,&pHesap-
>hesapLimit,&pMusteri->musteriTipi,&pTarih->gun,&pTarih->ay,&pTarih-
>yil)!=EOF)
    {
        if(havaleHesapNo == pHesap->hesapNo)
        {
            kontrol++;
            printf("Eski Hesabiniz: \t%d %d %f\n",pMusteri-
>MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye);
            pHesap->hesapBakiye += havalePara;
            printf("Yeni Hesabiniz: \t%d %d
%f\n",MusteriHesap.MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye);

            fprintf(yeniHesapFile2,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
            fprintf(havaleHesap,"%d %d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,hesapNo,havaleHesapNo,havalePara,pHesap-

```

```

>hesapLimit,pMusteri->musteriTipi,timeinfo ->tm_mday,timeinfo->tm_mon +
1,timeinfo->tm_year + 1900);

        }
        else
        {
            fprintf(yeniHesapFile2,"%d %d %f %f %d %d %d
%d\n",pMusteri->MusteriNo,pHesap->hesapNo,pHesap->hesapBakiye,pHesap-
>hesapLimit,pMusteri->musteriTipi,pTarih->gun,pTarih->ay,pTarih->yil);
        }

    }
    if(kontrol ==1)
    {
        printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak
icin:0'a Basiniz:");
        scanf("%d",&cikis);
        system("cls");
        if (cikis==1)
            Menu();
        else if(cikis==0)
            close();

    }

    fclose(hesapFile2);
    fclose(yeniHesapFile2);
    fclose(havaleHesap);

    remove("Hesaplar.txt");
    rename("hesap.txt","Hesaplar.txt");

    printf("\n\n\n\t\tAna menuye donmek icin:1 Sistemden cikmak icin:0'a
Basiniz:");
    scanf("%d",&cikis);
    system("cls");
    if (cikis==1)
        Menu();
    else if(cikis==0)
        close();

}

```

KULLANICI KATALOĐU

Kaynak kodumuzu alıřtırdıĐımızda ilk olarak karřımıza ana menü gelecektir.

- Programımız ilk kez alıřtırdıĐımız iin ilk olarak sisteme müşteri eklemeliyiz. Bunun iin řeiminizi yapın kısmına 1 yazmalıyız.

```
Banka Otomasyon Yazilimi

██████████ Ana Menu ██████████

1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

Seciminizi Yapin:1
```

- Müşteri ekleme işlemini seçtikten sonra programımız ekleyeceğimiz müşteri için random olarak hesap ve müşteri numarası atandı. Daha sonrada müşterinin ad, kimlik numarası, yaşı, adresi, telefon numarası müşteri tipi ve başlangıç bakiyesi bilgileri girildi ve ekrana aşağıdaki mesaj verildi.

```
Musteriilk:72
Hesapilk:481
Olusan Musteri Numaraniz:72
Olusan Hesap Numaraniz:481

Musterinin ismini giriniz:deniz
Musterinin kimlik Numarasini giriniz:47770345412
Musterinin Yasini giriniz:20
Musterinin Adresini giriniz:izmir
Musterinin telefon Numarasini giriniz:05310296576
Musteri Tipini Giriniz Bireysel:1/Ticari:2:1
Hesap Baslangic Bakiyesini Giriniz:5000
```

```
Musteri Basariyla Olusturuldu!
```

```
Ana menuye donmek icin:1 Sistenden cikmak icin:0'a Basiniz:
```


- Müşteri eklendikten sonra aşağıdaki dosyalar oluştu.

Musteriler - Not Defteri												
Dosya	Düzen	Biçim	Görünüm	Yardım								
72	481	deniz	47770345412	20	izmir	05310296576	1	5000.000000	10	5	2019	^
bireyselHesaplar - Not Defteri												
Dosya	Düzen	Biçim	Görünüm	Yardım								
72	481	deniz	47770345412	20	izmir	05310296576	1	5000.000000	10	5	2019	^
Hesaplar - Not Defteri												
Dosya	Düzen	Biçim	Görünüm	Yardım								
72	481	5000.000000	0.000000	1	10	5	2019					
YatirilanPara - Not Defteri												
Dosya	Düzen	Biçim	Görünüm	Yardım								
72	481	5000.000000	0.000000	1	10	5	2019					

- Sisteme hesap açma işlemi için tekrar giriyoruz.

```

Banka Otomasyon Yazilimi

      Ana Menu

1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

Seciminizi Yapin:2

```

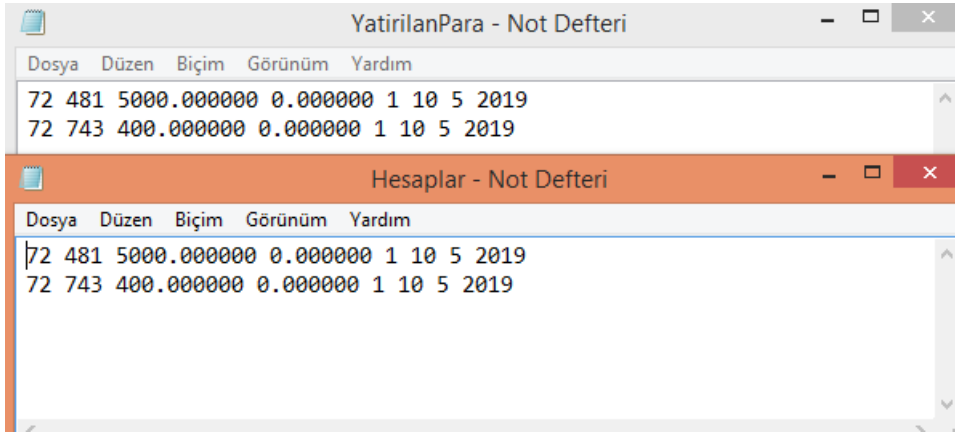
- Hesap açmak istediğimiz müşterinin müşteri numarasını girdikten sonra yeni hesabın başlangıç bakiyesini giriyoruz. Ve aşağıdaki mesajı alıyoruz.

```
Yeni Hesap Acmak Istediginiz Musterinin Numarasini Girin:72
HesapIlk:743
Baslangic Bakiyesini Giriniz:400

Hesabiniz Basariyla olusturuldu

Ana menuye donmek icin:1 Sistemden cikmak icin:0'a Basiniz:
```

- Müşteriye hesap açma işlemi yapıldıktan sonra ilgili dosyalar güncelleniyor



- Sisteme hesaptan para çekme işlemi için tekrar giriyoruz.

```
Banka Otomasyon Yazilimi

      Ana Menu

1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

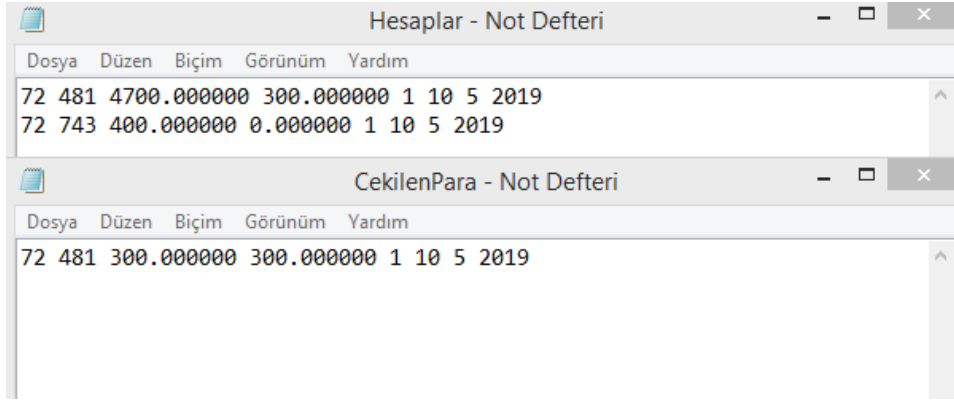
Seciminizi Yapin:3
```

- Para çekmek istediğimiz müşterinin müşteri numarasını girip, Hangi hesabından ne kadar para çekmek istediğini soruyoruz.

```
Musteri Numaranizi Giriniz:72
Hesaplariniz: 72 481 5000.000000 0.000000
Hesaplariniz: 72 743 400.000000 0.000000
Hesaplarinizdaki Toplam Para:5400.000000
Cekmek istediginiz miktari giriniz:300
Para Cekmek Istediginiz Hesap Numarasini Giriniz:481
300.000000: Para Cektiniz
72 481 4700.000000 300.000000 1 10 5 2019
Para Cekme Islemleri Basarili
Hesabinizdaki Yeni Para:4700.000000

Tekrardan Para Cekmek Icin:2 - Ana menuye donmek icin:1 - Cikis icin 0'a Basiniz:
```

- Hesaptan para çekme işlemi yapıldıktan sonra ilgili dosyalar güncelleniyor.



The image shows two overlapping Not Defteri (Notepad) windows. The top window is titled 'Hesaplar - Not Defteri' and contains two lines of text: '72 481 4700.000000 300.000000 1 10 5 2019' and '72 743 400.000000 0.000000 1 10 5 2019'. The bottom window is titled 'CekilenPara - Not Defteri' and contains one line of text: '72 481 300.000000 300.000000 1 10 5 2019'. Both windows have a menu bar with 'Dosya', 'Düzen', 'Biçim', 'Görünüm', and 'Yardım'.

Hesaplar - Not Defteri							
Dosya	Düzen	Biçim	Görünüm	Yardım			
72	481	4700.000000	300.000000	1	10	5	2019
72	743	400.000000	0.000000	1	10	5	2019

CekilenPara - Not Defteri							
Dosya	Düzen	Biçim	Görünüm	Yardım			
72	481	300.000000	300.000000	1	10	5	2019

- Sisteme hesaba para yatırma işlemi için tekrar giriş yapıyoruz.

```
Banka Otomasyon Yazilimi

      ██████████ Ana Menu ██████████

1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

Seciminizi Yapin:4
```

- Para yatırmak istenilen hesap numarası ve yatırılacak miktar bilgileri girildikten sonra hesap bakiyesini güncelliyoruz.

```
Para yatırmak istediğiniz hesap numarasını giriniz:481
Yatıracağınız miktarı giriniz:700
Hesabınızdaki Eski Para:4700.000000
Para Yatırma İşlemi Başarılı
Hesabınızdaki Yeni Para:5400.000000

      Ana menuye donmek icin:1 Sistemden cikmak icin:0'a Basiniz:
```

- Para yatırma işleminden sonra ilgili dosyalar güncelleniyor.

Hesaplar - Not Defteri										
Dosya	Düzen	Birim	Görünüm	Yardım						
72	481	5400.000000	300.000000	1	10	5	2019			
72	743	400.000000	0.000000	1	10	5	2019			

YatirilanPara - Not Defteri										
Dosya	Düzen	Birim	Görünüm	Yardım						
72	481	5000.000000	0.000000	1	10	5	2019			
72	743	400.000000	0.000000	1	10	5	2019			
72	481	700.000000	300.000000	1	10	5	2019			

- Sisteme havale işlemi için tekrar giriş yapıyoruz.

```
Banka Otomasyon Yazilimi

      Ana Menu

1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

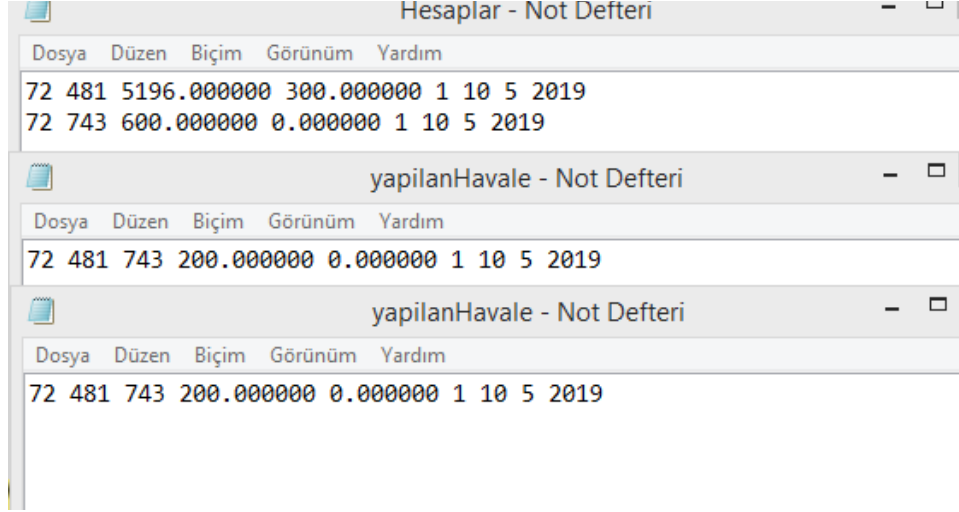
Seciminizi Yapin:5
```

- Müşterinin kendi hesap numarasını girmesini ardından havale edeceği miktarı ve havale yapmak istediği hesabın numarasını girmesini bekliyoruz. İşlem sonunda hesap bakiyeleri güncelleniyor.

```
Kendi Hesap Numaranizi Giriniz:481
Hesabiniz: 72 481 5400.000000
Yatiracaginiz miktari giriniz:200
Havale Yapmak istediginiz hesabin numarasini giriniz:743
Eski Hesabiniz: 72 743 400.000000
Yeni Hesabiniz: 0 743 600.000000

Ana menuye donmek icin:1 Sistemden cikmak icin:0'a Basiniz:
```

- Havale işleminden sonra ilgili dosyalar güncelleniyor.



The screenshot shows three overlapping 'Not Defteri' (Ledger) windows. The top window, titled 'Hesaplar - Not Defteri', contains two rows of data: '72 481 5196.000000 300.000000 1 10 5 2019' and '72 743 600.000000 0.000000 1 10 5 2019'. The middle and bottom windows, both titled 'yapilanHavale - Not Defteri', each contain one row of data: '72 481 743 200.000000 0.000000 1 10 5 2019'. Each window has a menu bar with 'Dosya', 'Düzen', 'Biçim', 'Görünüm', and 'Yardım'.

Dosya	Düzen	Biçim	Görünüm	Yardım
72	481	5196.000000	300.000000	1 10 5 2019
72	743	600.000000	0.000000	1 10 5 2019

Dosya	Düzen	Biçim	Görünüm	Yardım
72	481	743	200.000000	0.000000 1 10 5 2019

Dosya	Düzen	Biçim	Görünüm	Yardım
72	481	743	200.000000	0.000000 1 10 5 2019

- Sisteme bankanın gelir-gider raporunu görmek için tekrar giriş yapıyoruz.

```
Banka Otomasyon Yazilimi

      Ana Menu

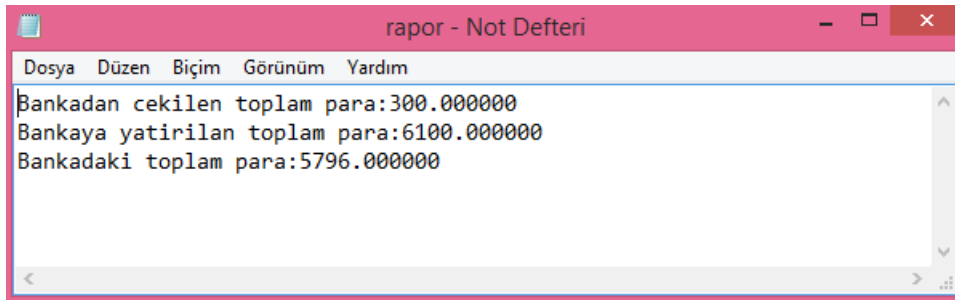
1.Yeni Musteri Olustur
2.Hesap Acma
3.Hesaptan Para Cekme
4.Hesaba Para Yatirma
5.Baska Hesaba Havale
6.Banka Gelir-Gider Raporu
7.Hesap Ozeti
8.Hesap Kapama
9.Cikis

Seciminizi Yapin:6
```

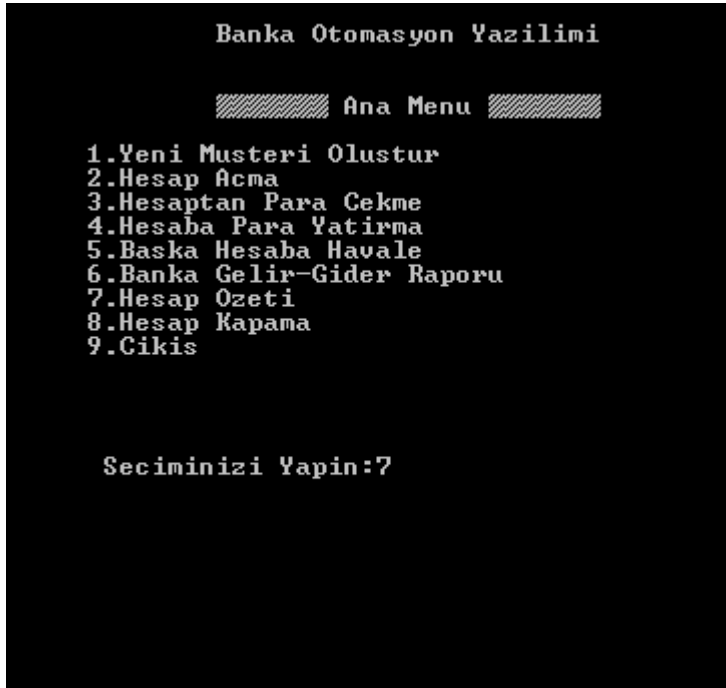
```
Bankadan cekilen toplam para:300.000000  
Bankaya yatirilan toplam para:6100.000000  
Bankadaki toplam para:5796.000000
```

Ana menuye donmek icin:1 Sistemden cikmak icin:0'a Basiniz:

- Gelir-gider raporu ile ilgili dosya güncellendi.



- Sisteme hesap özetini görmek için tekrar giriş yapıyoruz.



- Müşterinin hesap özetini görmek istediği hesabın hesap numarasını ve hangi tarihler arasında yapılan işlemleri görüntülemek istediğini girmesini bekliyoruz ve o hesapla belirtilen tarihler arasında yapılan işlemleri listeliyoruz.

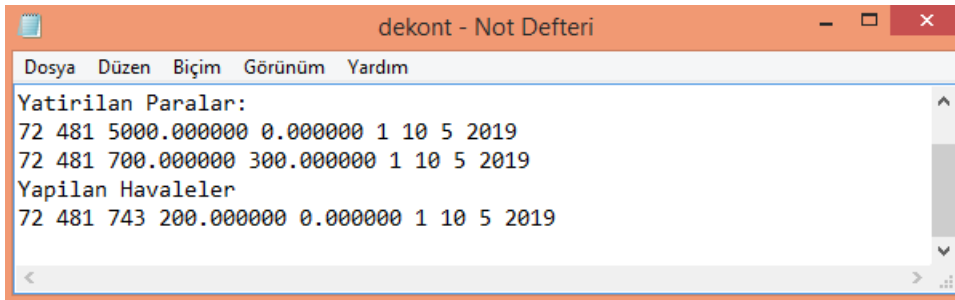
```

72 481 5196.000000 0.000000 10 5 2019 -1686795068
72 743 600.000000 0.000000 10 5 2019 -1686795068
72 481 deniz 47770345412 20 izmir 05310296576 1 5000.000000 10 5 2019
Hesap özetini görmek istediğiniz hesabın numarasını giriniz:481
Tarih Aralığını seçiniz:<YY/YY/YYYY>
Gun:09
Ay:05
Yil:2019
Gun:10
Ay:05
Yil:2019
Çekilen Paralar:
72 481 300.000000 300.000000 1 10 5 2019
Yatırılan Paralar:
72 481 5000.000000 0.000000 1 10 5 2019
72 481 700.000000 300.000000 1 10 5 2019
Yapılan Havaleteler
72 481 743 200.000000 0.000000 1 10 5 2019

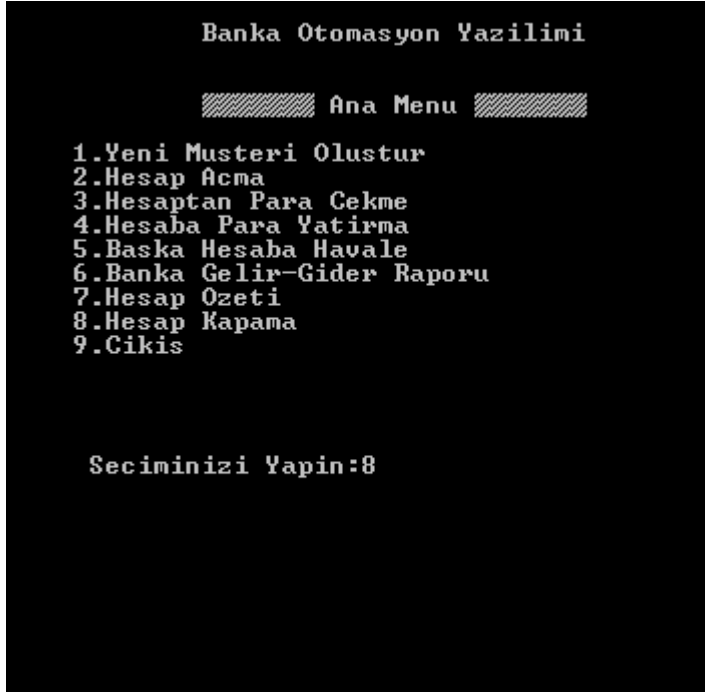
      Ana menuye donmek icin:1 Sistemden cikmak icin:0'a Basiniz:

```

- Gelir-gider raporu ile ilgili dosya güncellendi.



- Sisteme hesap kapatmak için tekrar giriş yapıyoruz.



- Müşteri kapatmak istediği hesabın hesap numarasını giriyor. Hesap bakiyesi 0 dan büyükse hesaptaki parayı çekmesi taktirde hesabı kapatıp hesaplar dosyasından siliyor.



Hesaplar - Not Defteri							
Dosya	Düzen	Biçim	Görünüm	Yardım			
72	481	5196.000000	300.000000	1	10	5	2019