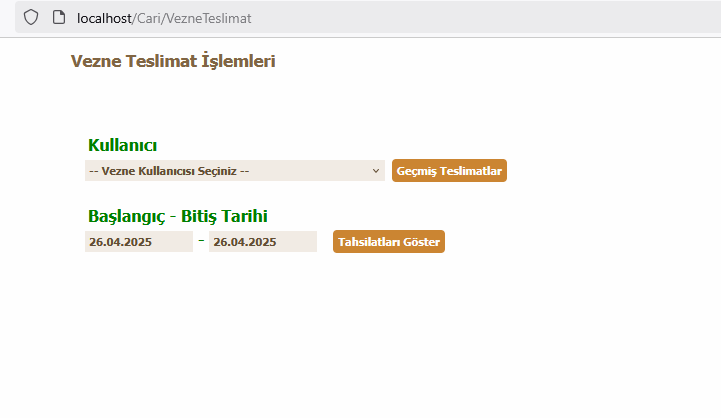
Yiğit:

Redis bilgisayarına kurulacak ve konfigürasyon ayarları yapılacak (eğer zorlanırsan şamili ara ve birlikte bakın 😊)

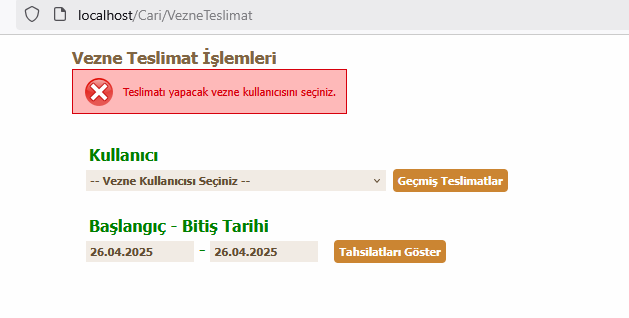
Güncel veritabanı yeri:

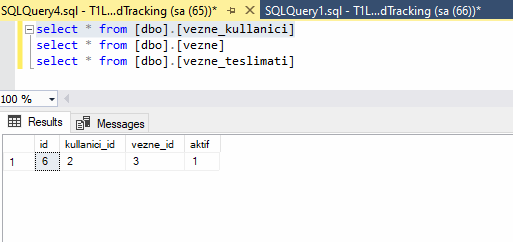
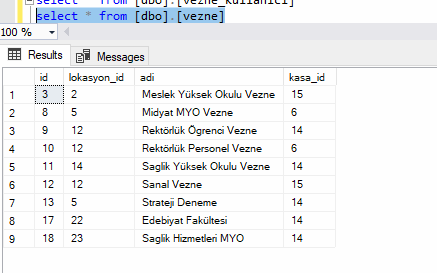
FoodTrackingSoftware/Food.Tracking.Domain.Context.FoodTrackingV2.bak

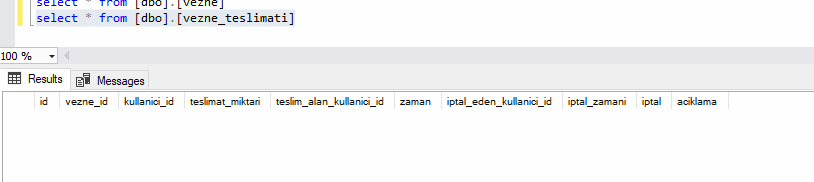
VezneTeslimat:



Vezne kullanıcısı seçilmez ise hata verilecek:

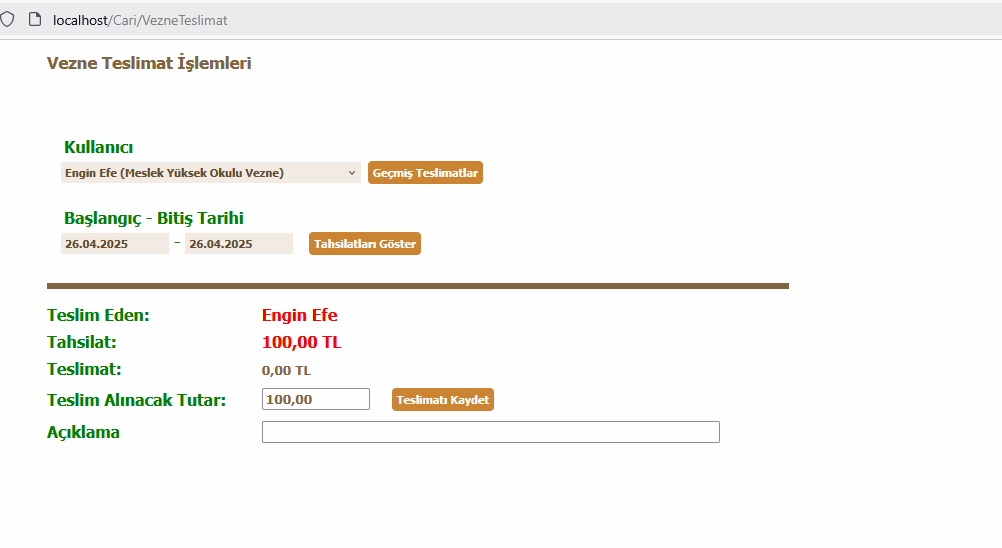


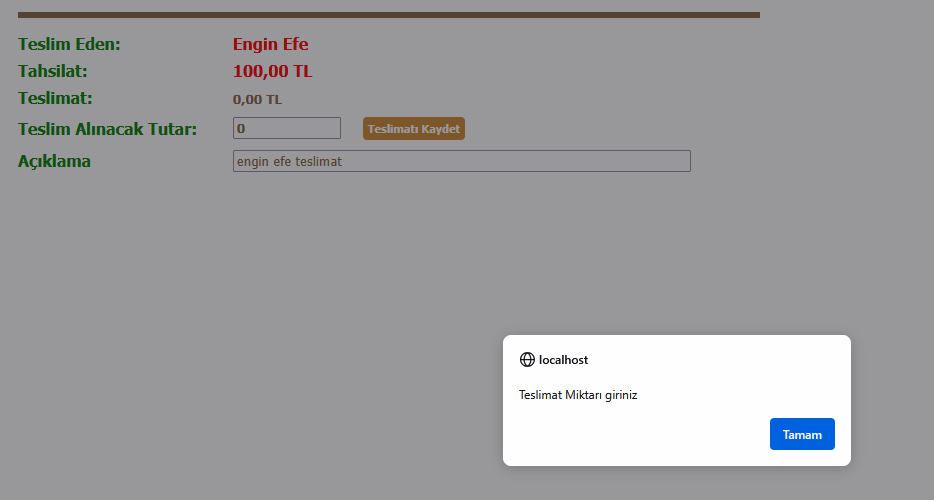


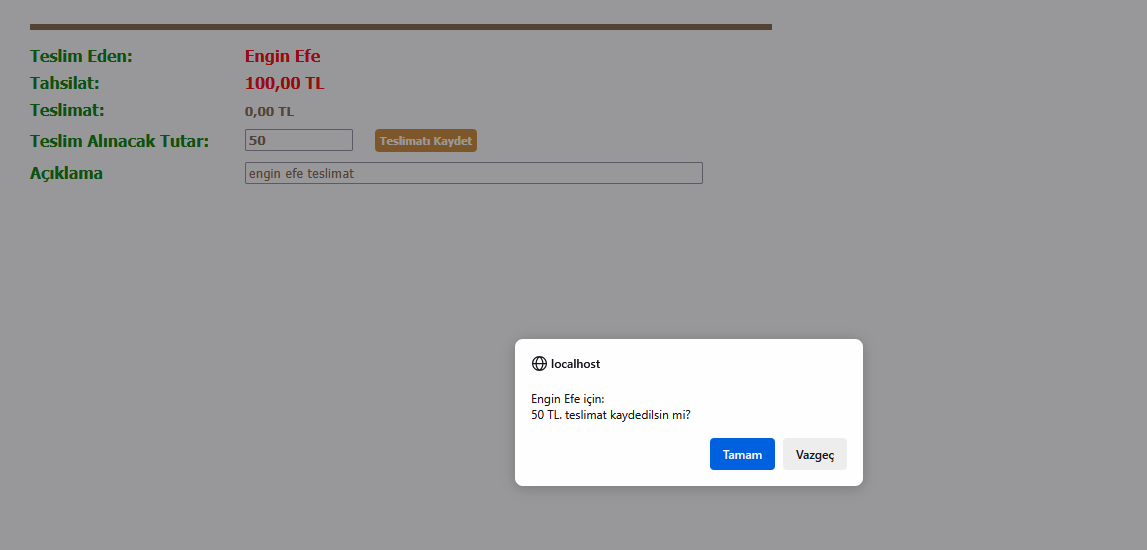


select b.id as vezne\_id , c.adi + ' ' + c.soyadi + ' (' + b.adi + ')' from [dbo].[vezne\_kullanici] a inner join [dbo].[vezne] b on a.vezne\_id = b.id

inner join kullanici c on a.kullanici\_id = c.id







Tahsilat:

public static double GetVezneTahsilatToplami(Kullanici kullanici, DateTime baslangicTarihi, DateTime bitisTarihi)

{

double? result = Session.Query<HesapHareketi>()

.Where(h => h.IslemYapanKullanici.Id == kullanici.Id)

.Where(h => h.IslemZamani >= baslangicTarihi && h.IslemZamani < bitisTarihi.AddDays(1))

.Where(h=> h.IslemTipi == IslemTurleri.KREDI\_YUKLEME || h.IslemTipi == IslemTurleri.KREDI\_GERI\_IADE)

.Where(h=> h.Iptal != 1)

.Sum(h=> (double?)(h.Kredi - h.Harcama));

return result.GetValueOrDefault();

}

Teslimat:

public static double GetVezneTeslimatToplami(Kullanici kullanici, DateTime baslangicTarihi, DateTime bitisTarihi)

{

double? result = Session.Query<VezneTeslimati>()

.Where(h => h.Kullanici.Id == kullanici.Id)

.Where(h => h.Zaman >= baslangicTarihi && h.Zaman < bitisTarihi.AddDays(1))

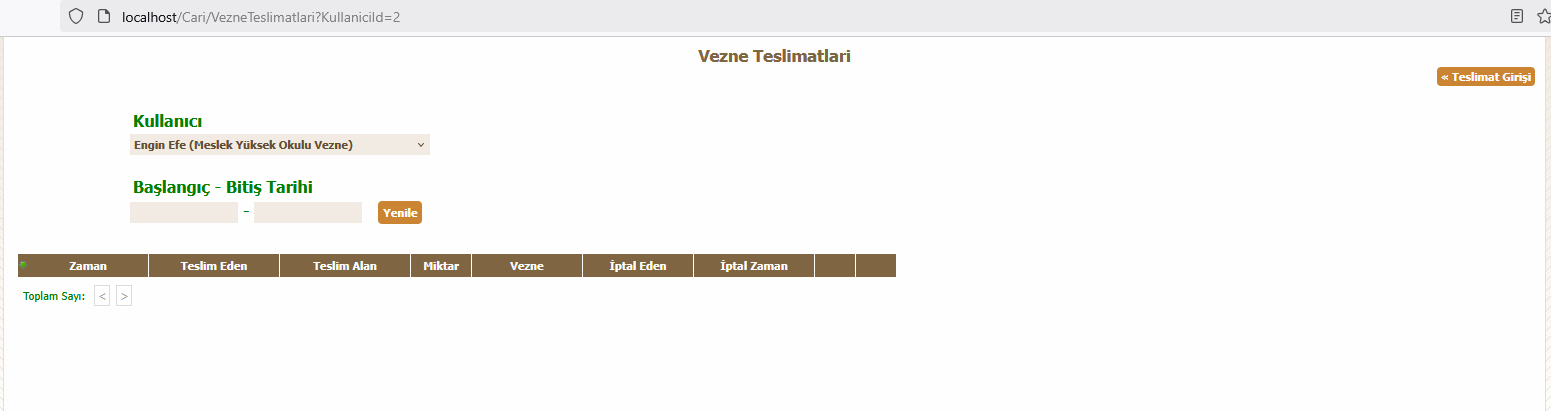
.Where(h => !h.Iptal)

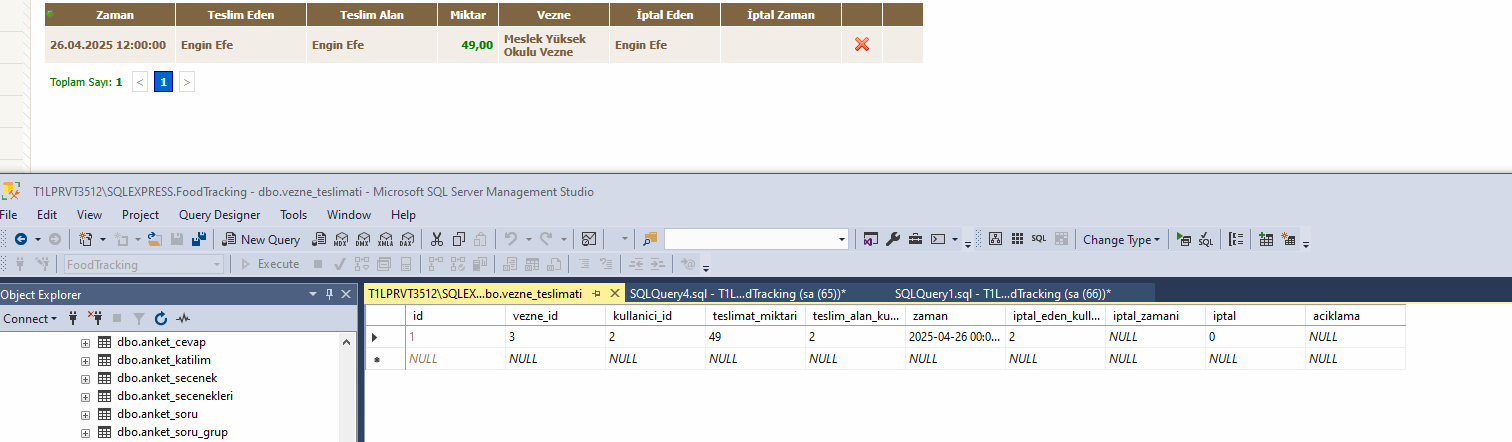
.Sum(h => (double?)h.TeslimatMiktari);

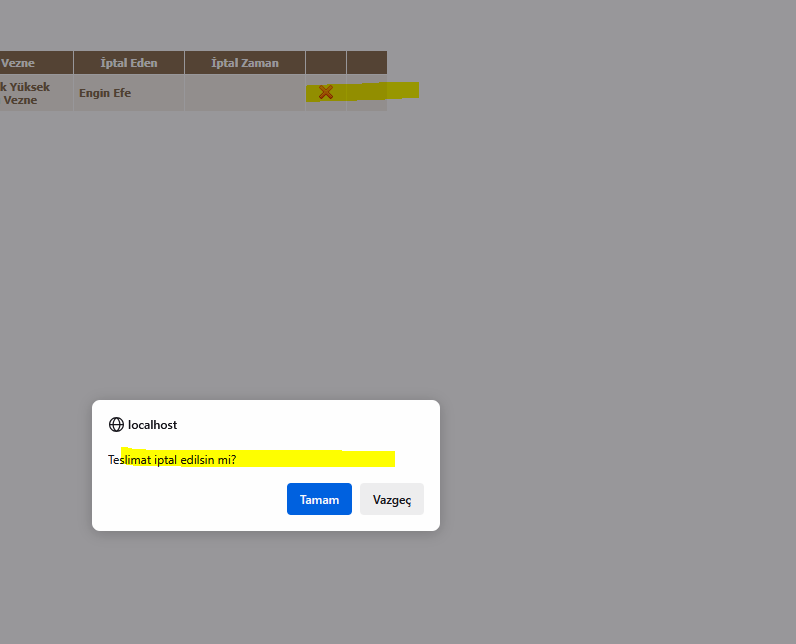
return result.GetValueOrDefault();

}

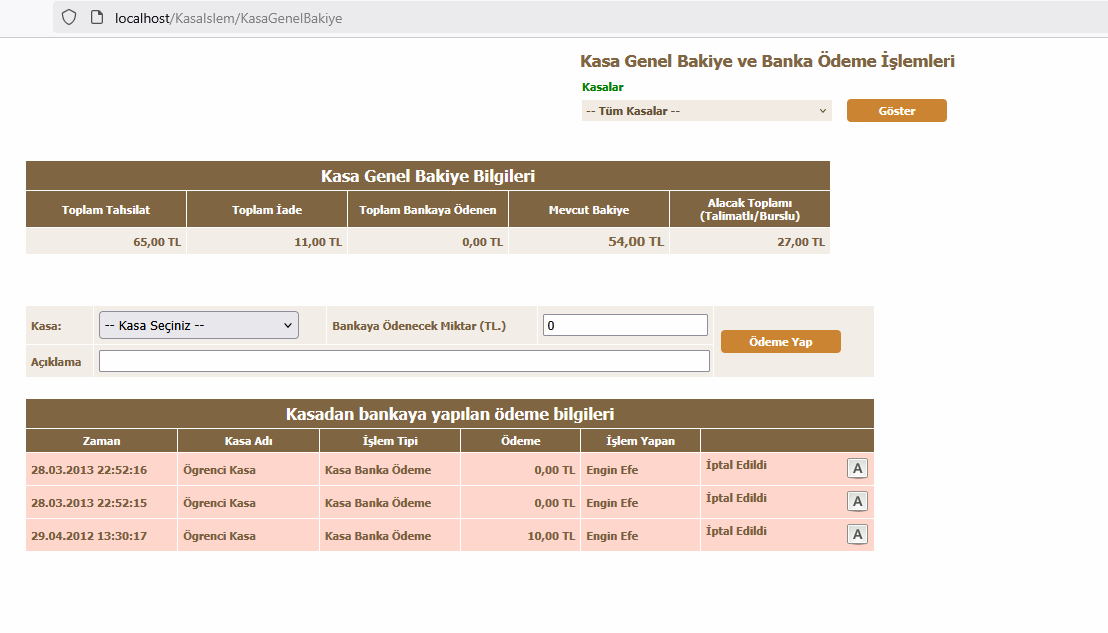
Geçmiş teslimatlar:

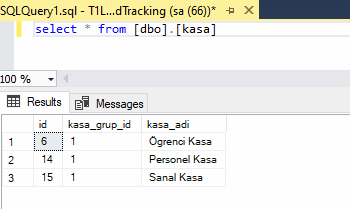






Şamil:





select \* from [dbo].[kasa];

select kasa\_id,islem\_tipi,sum(kredi) as ToplamTahsilat, sum(harcama) as Toplamiade from [dbo].[hesap\_hareketi] where kasa\_id = 6 and iptal = 0

group by kasa\_id,islem\_tipi;

select \* from [dbo].[hesap\_hareketi] where kasa\_id = 6;

select sum(h.Kredi) as ToplamTahsilat, sum(h.Kredi- h.Harcama) as MevcutBakiye, sum(h.Harcama) as ToplamIade from [hesap\_hareketi] h where 1=1 and kasa\_id= 6

and (h.islem\_tipi='K' or h.islem\_tipi='KG') and not (h.iptal=1);

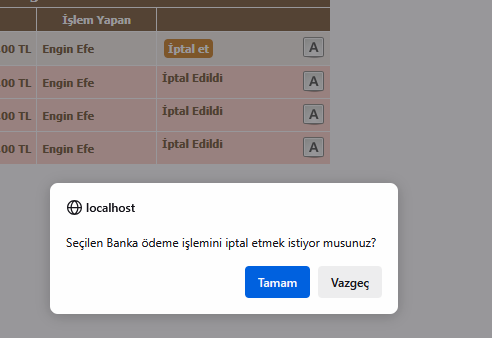
select sum(h.Kredi) as ToplamTahsilat, sum(h.Kredi- h.Harcama) as MevcutBakiye, sum(h.Harcama) as ToplamIade from [hesap\_hareketi] h

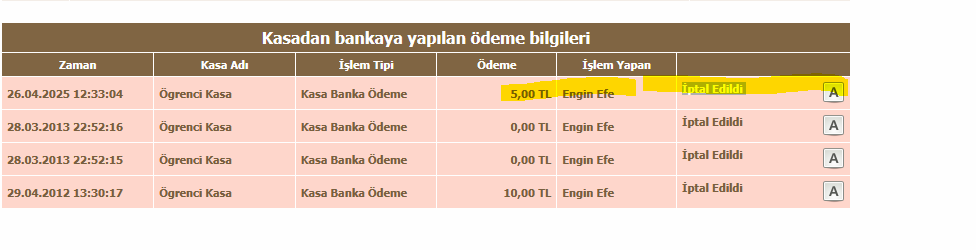
where (h.odeme\_talimatli=1 and h.iptal<>1)

select \* from [dbo].[banka\_islemi]



Iptal et:





public static class IslemTurleri

{

public static readonly string REZERVASYON = "R";

public static readonly string REZERVASYON\_IPTAL = "RI";

public static readonly string KREDI\_YUKLEME = "K";

public static readonly string KREDI\_TOPLU\_ODEME = "KT"; // Personel için yapılan toplu ödemeler

public static readonly string KREDI\_GERI\_IADE = "KG";

public static readonly string KASADAN\_BANKAYA\_ODEME = "BO";

public static readonly Dictionary<string, string > IslemTuruListesi = new Dictionary<string, string>

{

{ REZERVASYON, "Rezervasyon" },

{ REZERVASYON\_IPTAL, "Rezervasyon İptal" },

{ KREDI\_YUKLEME, "Kredi Yükleme"},

{ KREDI\_TOPLU\_ODEME, "Toplu Tahsilat"},

{ KREDI\_GERI\_IADE, "Kredi İade" },

{ KASADAN\_BANKAYA\_ODEME, "Kasa Banka Ödeme" }

};

}

public static class Parametreler

{

public static readonly string KREDI\_YUKLEME\_ALT\_LIMIT = "KREDI\_YUKLEME\_ALT\_LIMIT";

//public static readonly string KREDI\_YUKLEME\_UST\_LIMIT = "KREDI\_YUKLEME\_UST\_LIMIT";

//public static readonly string KREDI\_IPTAL\_GERIYE\_GUN\_SAYISI = "KREDI\_IPTAL\_GERIYE\_GUN\_SAYISI";

public static readonly string REZERVASYON\_ISLEM\_SURESI = "REZERVASYON\_ISLEM\_SURESI"; // OgunTakvimi.BaslangicZamani ndan geriye saat (yeni rezervasyon ve iptal)

public static readonly string REZERVASYON\_ILERIYE\_YONELIK\_GUN\_SAYISI = "REZERVASYON\_ILERIYE\_YONELIK\_GUN\_SAYISI";

public static readonly string KASA\_LIMITI\_GUNLUK = "KASA\_LIMITI\_GUNLUK";

public static readonly string MISAFIR\_ICIN\_TAM\_UCRET = "MISAFIR\_ICIN\_TAM\_UCRET";

public static readonly string KOTADAN\_REZERVASYON\_ICIN\_TAM\_UCRET = "KOTADAN\_REZERVASYON\_ICIN\_TAM\_UCRET";

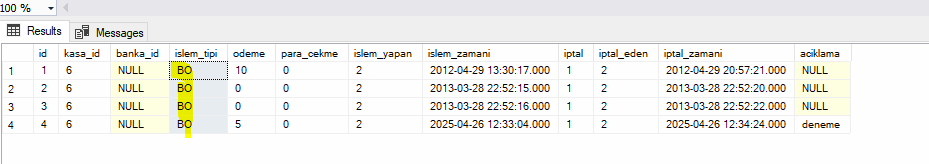
public static readonly string LOKASYON\_KOTA\_YUZDESI = "LOKASYON\_KOTA\_YUZDESI";

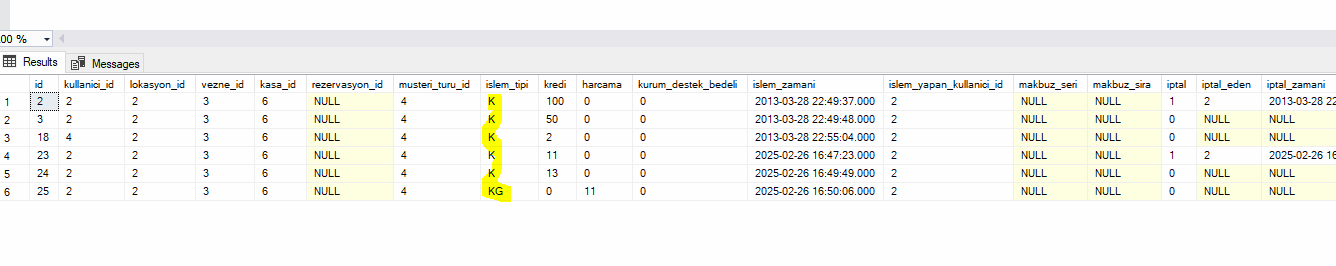
public static readonly string PROXIMITY\_PORT\_NAME = "PROXIMITY\_PORT\_NAME";

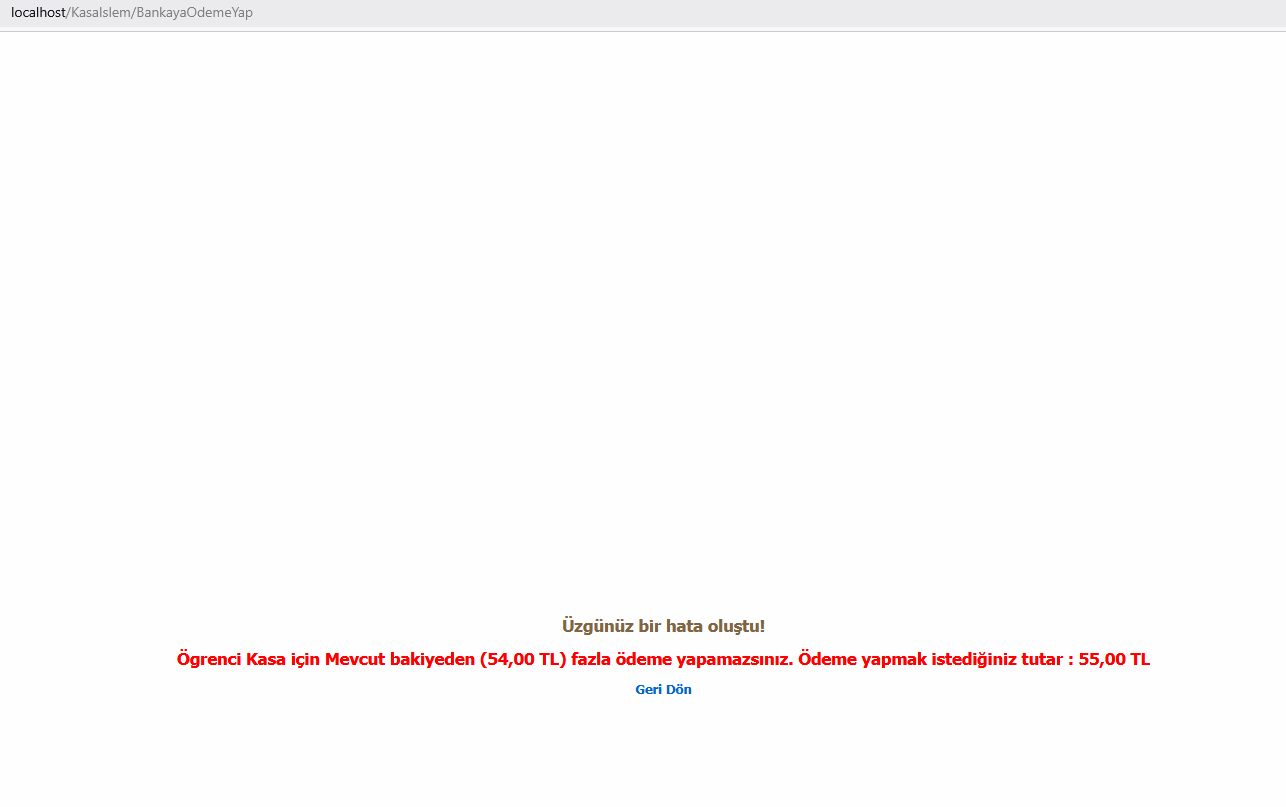
public static readonly string KIOSK\_ONAY\_TOLERANSI = "KIOSK\_ONAY\_TOLERANSI";

public static readonly string VEZNE\_KASA\_SECEBILIR = "VEZNE\_KASA\_SECEBILIR"; // Kredi Yüklemede kasa seçimi için

}







Bankaya Ödenecek Miktar (TL.) tutar Mevcut Bakiyeden büyük ise hata verilecek. Javascript de hata mesajı verebilirsiniz yada ayrı bir sayfaya yönlendirerek de olur size bağlı.