

GRUP-17

**NESNE YÖNELİMLİ PROGRAMLAMA II DERSİ DÖNEM PROJESİ TANITIMI OLUŞUM SÜRECİ**

A screenshot of a computer

Description automatically generated with low confidence

C# Form ile veritabanı arasında kod aracılığla bağlantı sağladım .

VERİTABANI İLE BAĞLANTI



FORM 1.DESİGN :

Kullanıcı Ekleme için =

Label ve textboxlara veritabanını oluştururken kullanmış olduğum bilgileri tanımladım. (Kullanıcı Ad, Şifre, E-mail, Telefon No, Departman, T.C No)

Butonlara kullanacak olduğumuz metodları yazdım . (Kullanıcı Kaydet, Kullanıcı Sil , Kullanıcıları Göster )

ListBox1 de Kaydettiğim verileri görüntüledikten sonra kullanıcı sil komutunu çalıştırdım.

FAKÜLTE SEKRETERİ İÇİN PROGRAMA KULLANICI EKLEME

FORM 1.DESİGN :

Evrak Kaydı Oluşturmak için =

Label ve textboxlara veritabanını oluştururken kullanmış olduğum bilgileri tanımladım.

Butonlara kullanacak olduğumuz metodları yazdım .

ListBox3 de evrakları tarih bazında listeleme yaptım.

ARŞİVE EVRAK KAYDI OLUŞTURMA



FORM 1.DESİGN :

Personel kaydı oluşturma=

Label ve textboxlara veritabanını oluştururken kullanmış olduğum bilgileri tanımladım.

Butonlara kullanacak olduğumuz metodu yazdım .

PERSONEL KAYDI OLUŞTURMA

FORM 1.DESİGN :

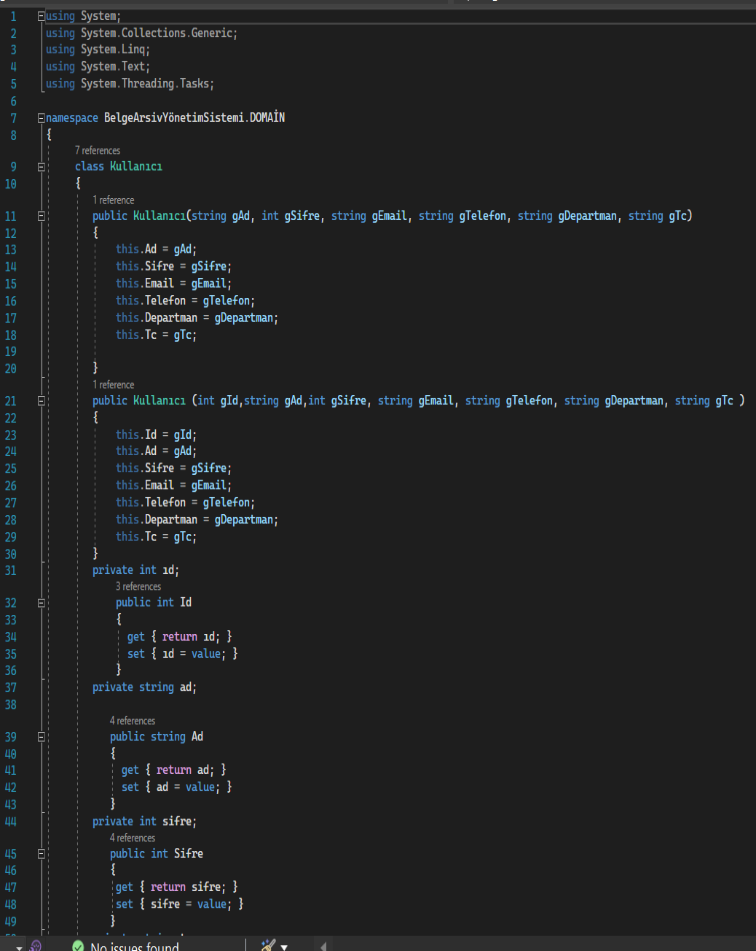
Emanet İşleri Kaydı Oluşturmak için =

Label ve textboxlara veritabanını oluştururken kullanmış olduğum bilgileri tanımladım.

Butonlara kullanacak olduğumuz metodları yazdım .

ListBox2 de Emanet verilen evrakları listeledim.

EMANET İŞLERİ İÇİN KAYIT OLUŞTURMA

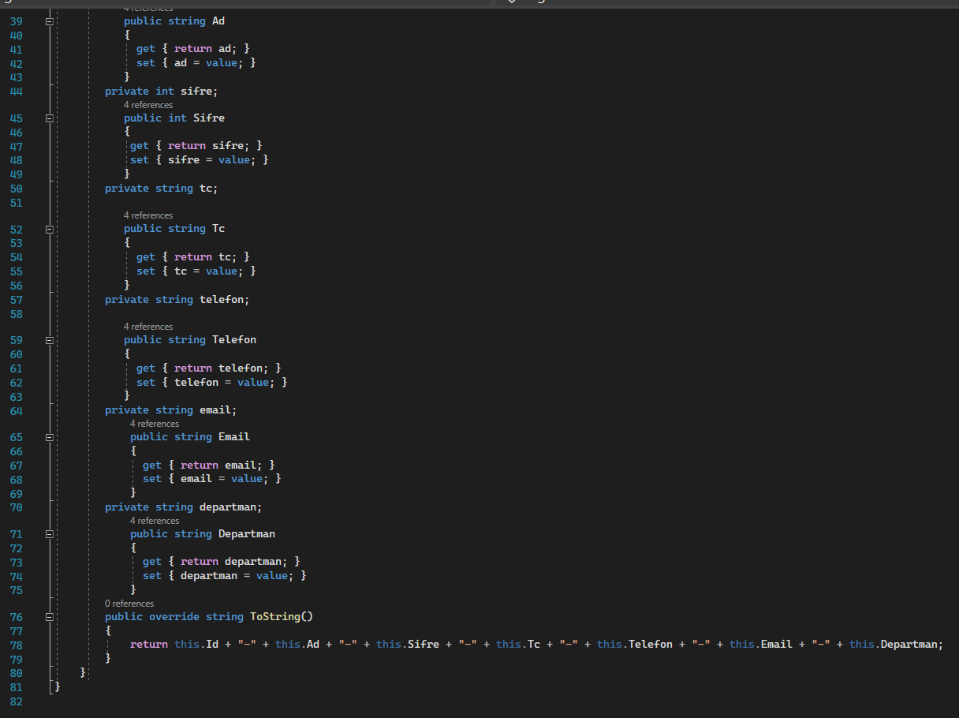


Domain klasörüne kullanıcı.cs diye Class ekledim.

Encapsulation yaptım ( nesnenin belirli özellik ve metotlarının erişiminin kısıtlanması ve saklanması için . Bu kısıtlamalar ile kötü amaçlı kullanım ve istenmeyen veri girişlerini engelledim )

Constructor metod oluşturdum (Tasarlanan sınıftan nesne üretildiğinde anında çalışan metod olduğu için yani bir nesne oluşturduğumuzda ilk olarak Constructor Method çalışır. Nesneleride sınıflardan ürettiğimize göre bu yapının sınıf içerisinde olduğunu tahmin edebiliriz. Dolayısıyla, her sınıfta mutlaka bir Constructor method vardır)

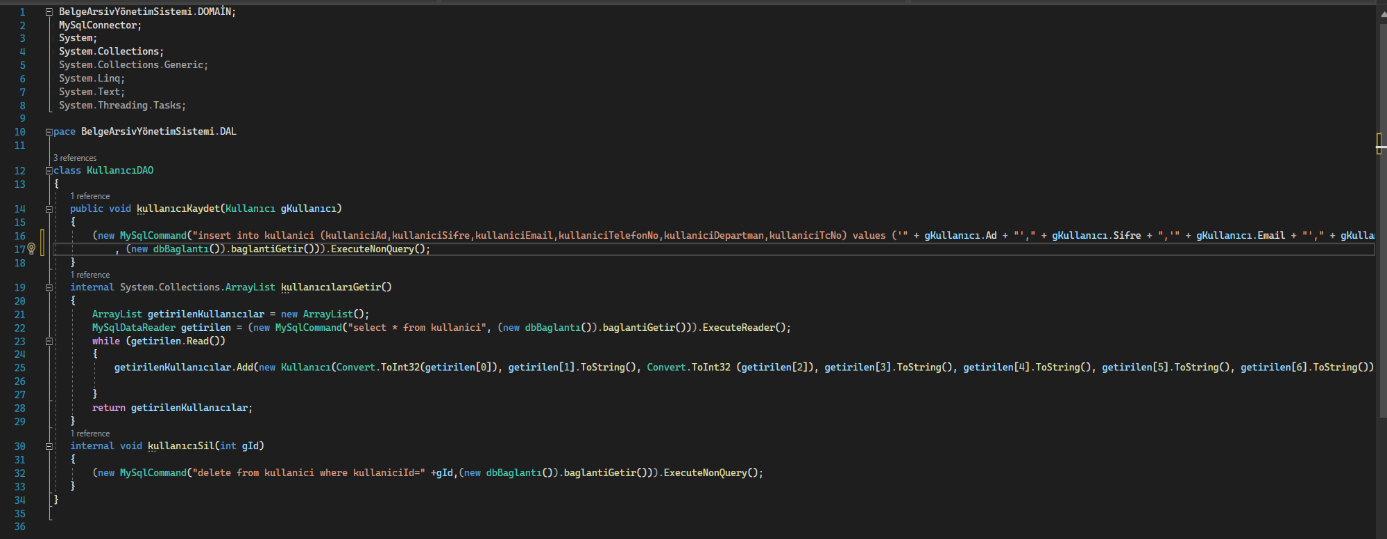
KULLANICI.CS



Daha sonra dal klasörüne yani dao ya gidip kullanıcı dao diye class ekledim çünkü dao veri tabanı ile ilgili olan işlemleri yapacak metodları barındırır.

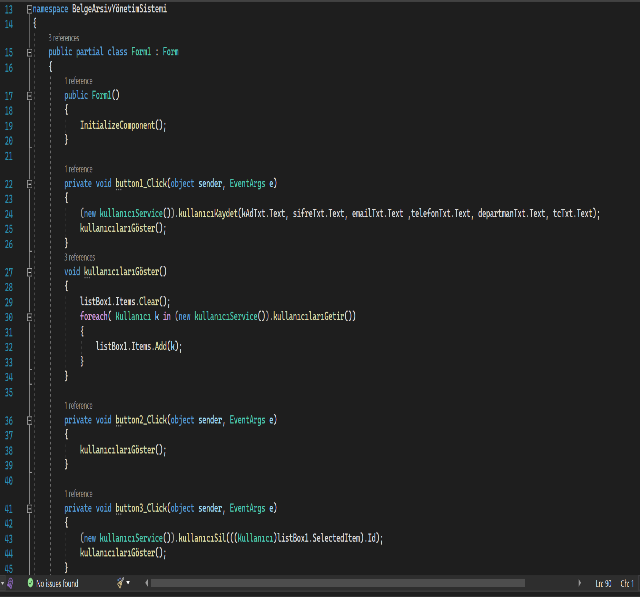
Kullanıcı dao ya using ….. Domain yazdım amacım tanımladığım classları kullanmaktı.

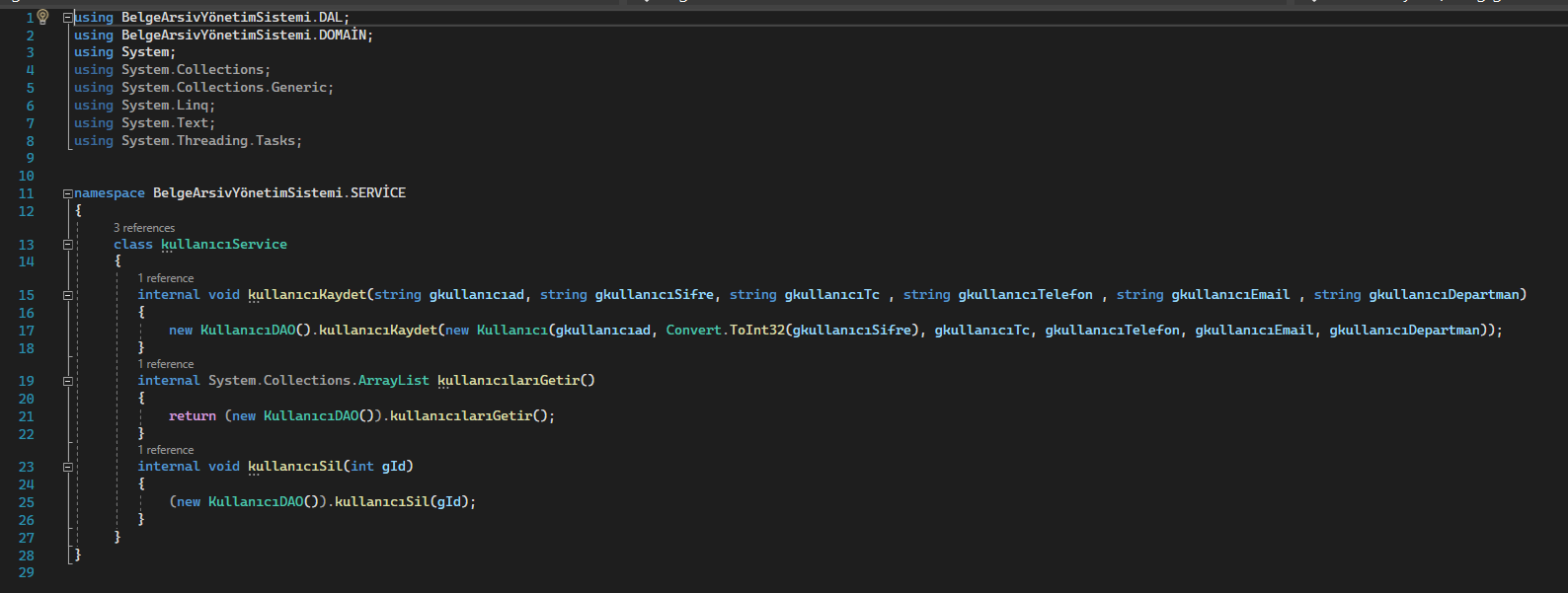
KULLANICI.DAO



Form1 cs design kısmına geldim ilgili butona click yaptım çünkü istek yapacağım servis elemanı onun içindeydi.

KULLANICI EKLEMEK İÇİN FORM1.CS

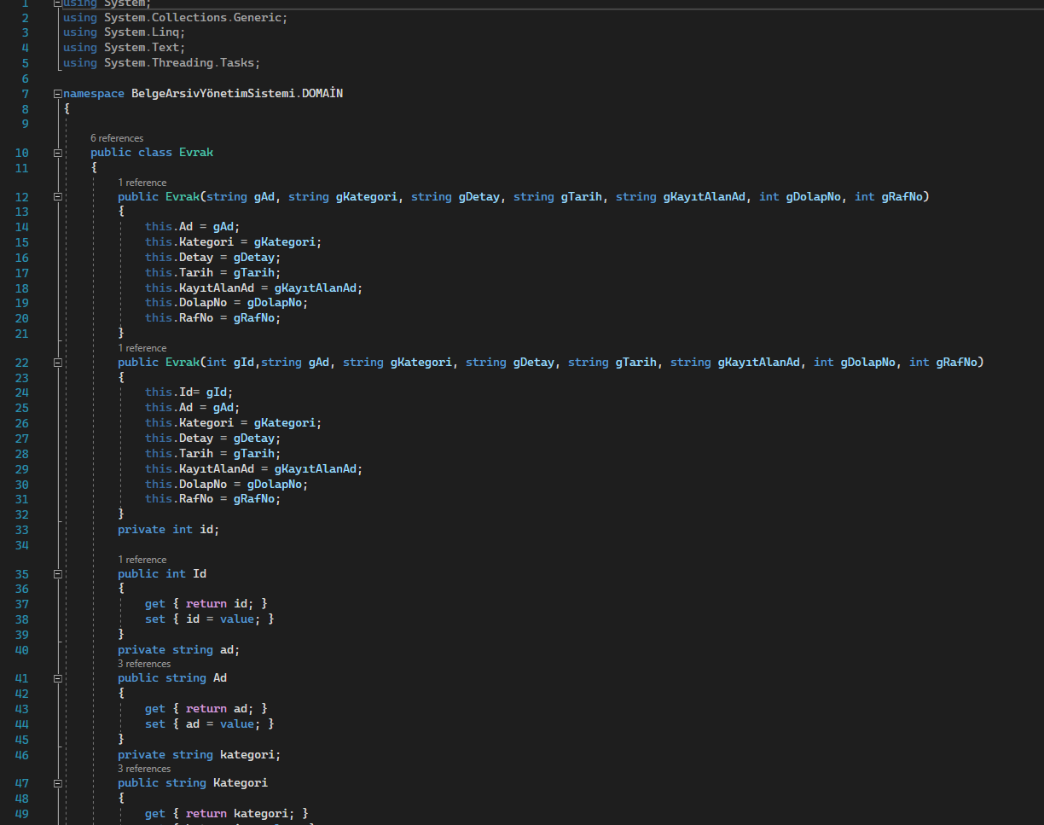




Kullanıcı service diye class oluşturdum ilgili alanların kodlarını yazdım ve kullanıcı ekleme kısmı aktif bir şekilde çalıştı.

Kullanıcı service using…. DAL ve using….DOMAİN yazdım amacım tanımladığım classları kullanıcı service kullanmaktı.

KULLANICI SERVİCE



Evrak kategori diye bir alan oluşturdum. Bunu oluşturmamdaki amaç ise bize verilen :

-Öğrenci sınav kağıtları

-Öğrenci Evrakları

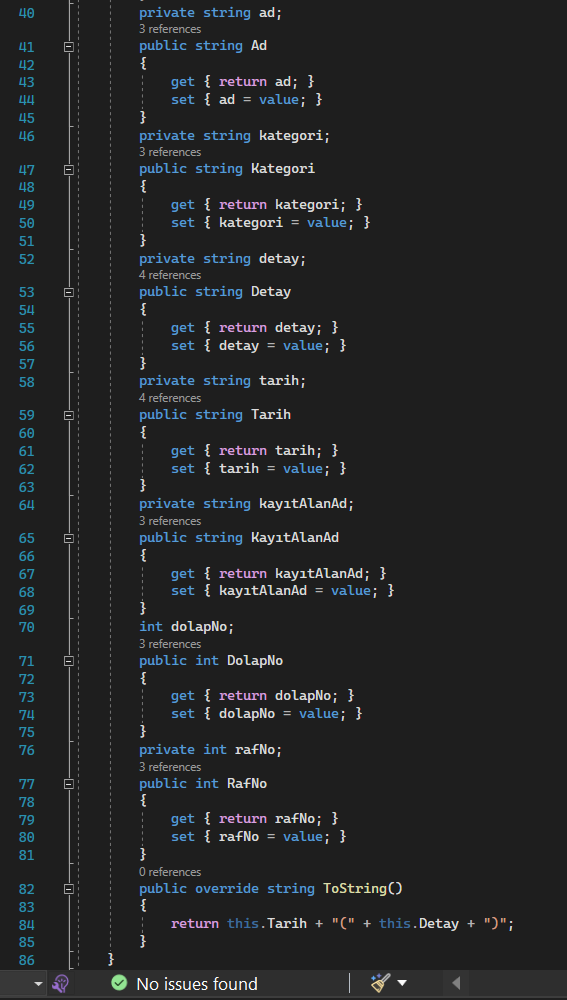
-Personel Evrakları

-Muhasebe ve Satın Alma Evrakları

-Bölüm Evraklarını bu kısımda kullanmak için.

Constructor metodlardan evrak tarih ve evrak detayı override ederek evrakları tarih bazında listeleme yaptım.

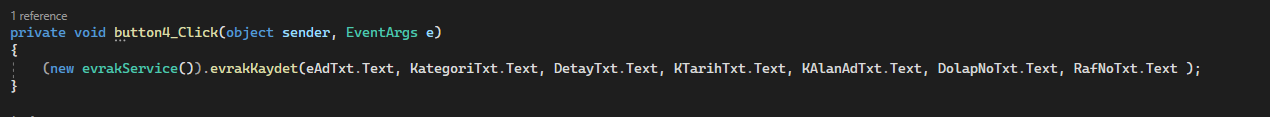
EVRAK.CS

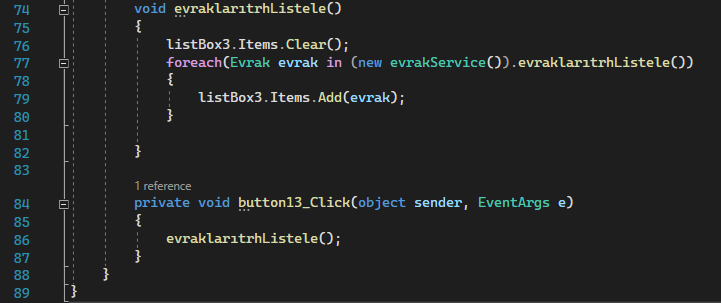


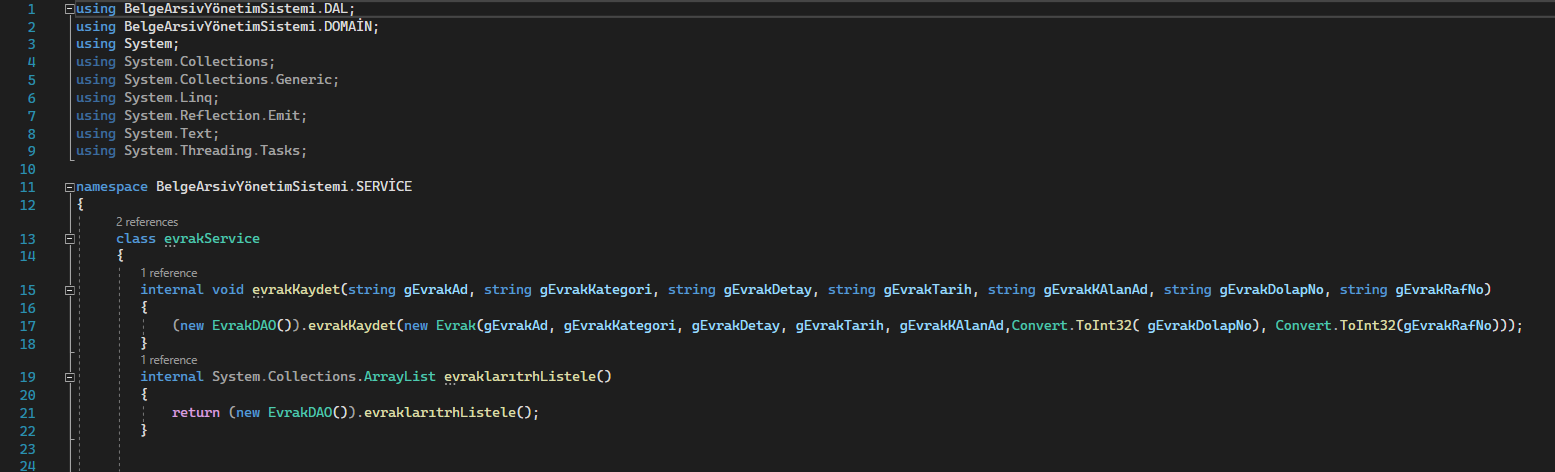
KULLANICIDA YAPTIĞIM İŞLEMLERİN AYNISI EVRAK İÇİN YUKARIDA YAZILI OLANLANLARDA YAPTIM.

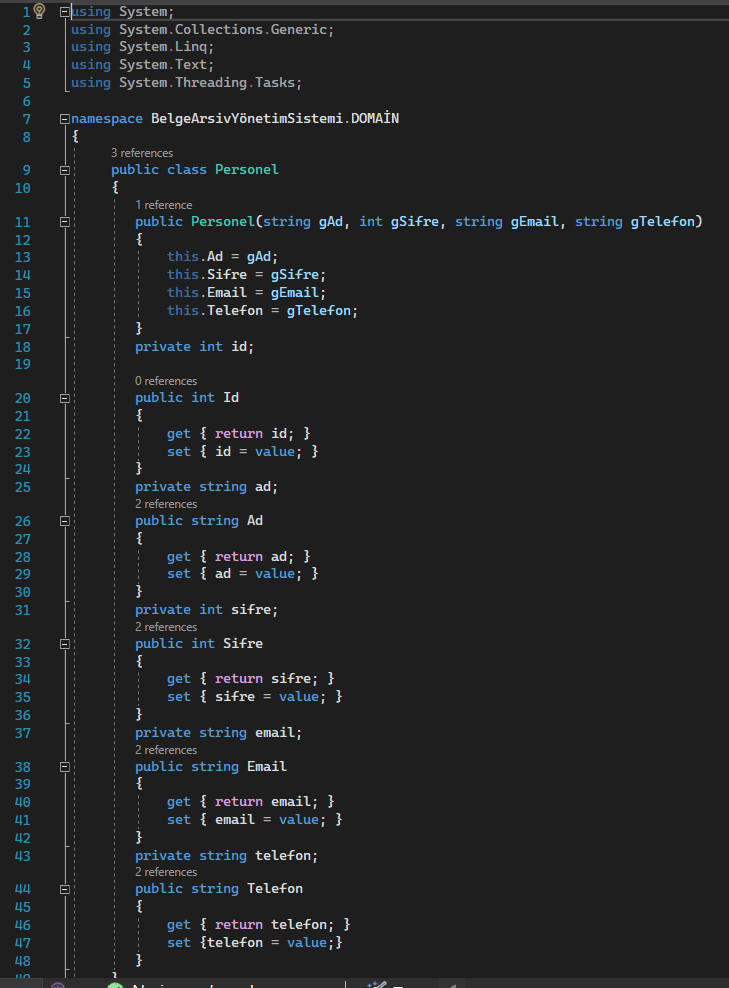
EVRAK.DAO, FORM1, EVRAK SERVİCE







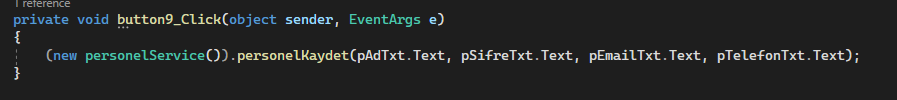


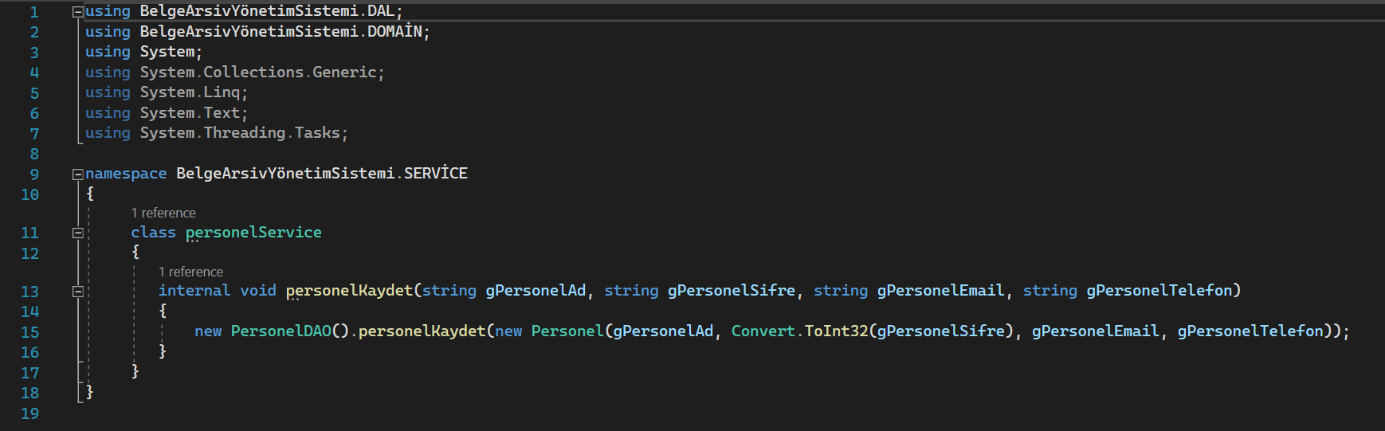


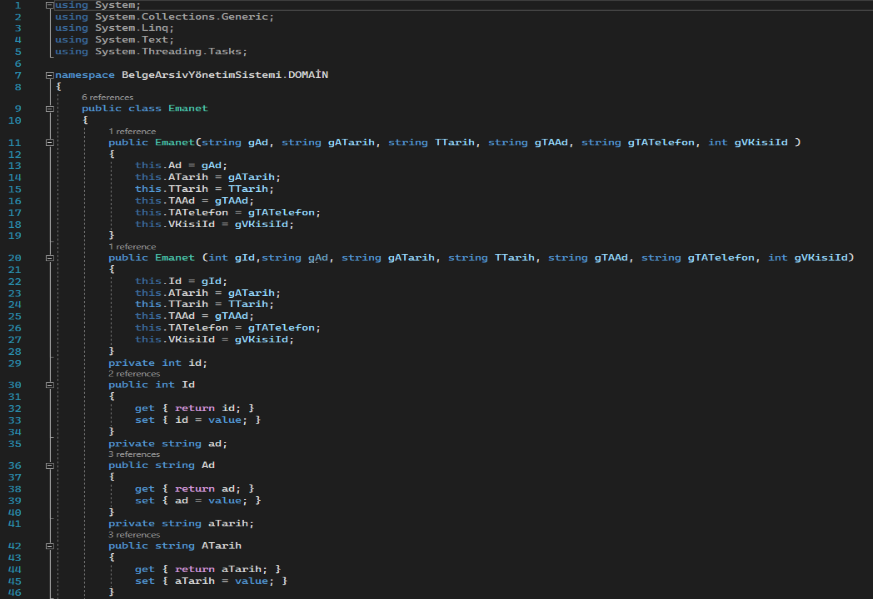
KULLANICI VE EVRAKTA YAPTIĞIM İŞLEMLERİ PERSONEL İÇİNDE UYGULADIM.

PERSONEL









Emanet içinde aynı işlemleri uyguladım.

EMANET İŞLERİ





