*BELAL ALHAMID*

*G221210592*

*WEP PROJESI*

*Kod raporu*

*-----------------------------------------------*

Login 1

Bu C# kodu, bir Windows Forms uygulaması içindeki giriş formunu tanımlar. Aşağıda kodun açıklamaları bulunmaktadır:

1. using ifadeleri: Gerekli .NET kütüphanelerini ve namespace'leri içe aktarır.
2. namespace Progekt\_Berber\_belal: Kodun bulunduğu namespace'i tanımlar.
3. public partial class Login : Form: Login adında bir form sınıfı tanımlar ve Form sınıfından türetilir.
4. public Login(): Login sınıfının yapıcı metodunu tanımlar. Bu metot, formun başlatılmasını sağlar.
5. private void ClrBtn\_Click(object sender, EventArgs e): "Temizle" butonunun tıklanma olayını işler. Kullanıcı adı ve şifre alanlarını temizler.
6. private void CrossBtn\_Click(object sender, EventArgs e): "Kapat" butonunun tıklanma olayını işler. Formu kapatır.
7. private void LoginBtn\_Click(object sender, EventArgs e): "Giriş" butonunun tıklanma olayını işler. Kullanıcı adı ve şifrenin doğruluğunu kontrol eder ve uygun mesajları görüntüler.
8. MessageBox.Show(): Bir iletişim kutusu görüntüler.
9. Home obj = new Home(); obj.Show(); this.Hide();: Eğer kullanıcı adı ve şifre doğru ise, ana pencereyi (Home) gösterir ve giriş formunu gizler.

Bu kod, bir kullanıcı giriş formunu oluşturur ve kullanıcının giriş yapmasını sağlar.

3.5

========================================================================================

HOME 2

 using ifadeleri: Gerekli .NET kütüphanelerini ve namespace'leri içe aktarır.

 namespace Progekt\_Berber\_belal: Kodun bulunduğu namespace'i tanımlar.

 public partial class Home : Form: Home adında bir form sınıfı tanımlar ve Form sınıfından türetilir.

 public Home(): Home sınıfının yapıcı metodunu tanımlar. Bu metot, formun başlatılmasını sağlar.

 private void LogoutBtn\_Click(object sender, EventArgs e): "Çıkış" butonunun tıklanma olayını işler. Kullanıcıyı giriş ekranına yönlendirir ve mevcut formu gizler.

 private void CrossBtn\_Click(object sender, EventArgs e): "Kapat" butonunun tıklanma olayını işler. Formu kapatır.

 private void DocBtn\_Click(object sender, EventArgs e): "Doktor" butonunun tıklanma olayını işler. Doktor formunu gösterir ve mevcut formu gizler.

 private void PatientBtn\_Click(object sender, EventArgs e): "Hasta" butonunun tıklanma olayını işler. Hasta formunu gösterir ve mevcut formu gizler.

 private void DiagBtn\_Click(object sender, EventArgs e): "Teşhis" butonunun tıklanma olayını işler. Teşhis formunu gösterir ve mevcut formu gizler.

 private void Home\_Load(object sender, EventArgs e): Form yüklendiğinde yapılacak işlemleri tanımlar. Bu metot şu anda boş olarak bırakılmıştır.

 using ifadeleri: Gerekli .NET kütüphanelerini ve namespace'leri içe aktarır.

 namespace Progekt\_Berber\_belal: Kodun bulunduğu namespace'i tanımlar.

 public partial class Home : Form: Home adında bir form sınıfı tanımlar ve Form sınıfından türetilir.

 public Home(): Home sınıfının yapıcı metodunu tanımlar. Bu metot, formun başlatılmasını sağlar.

 private void LogoutBtn\_Click(object sender, EventArgs e): "Çıkış" butonunun tıklanma olayını işler. Kullanıcıyı giriş ekranına yönlendirir ve mevcut formu gizler.

 private void CrossBtn\_Click(object sender, EventArgs e): "Kapat" butonunun tıklanma olayını işler. Formu kapatır.

 private void DocBtn\_Click(object sender, EventArgs e): "Doktor" butonunun tıklanma olayını işler. Doktor formunu gösterir ve mevcut formu gizler.

 private void PatientBtn\_Click(object sender, EventArgs e): "Hasta" butonunun tıklanma olayını işler. Hasta formunu gösterir ve mevcut formu gizler.

 private void DiagBtn\_Click(object sender, EventArgs e): "Teşhis" butonunun tıklanma olayını işler. Teşhis formunu gösterir ve mevcut formu gizler.

 private void Home\_Load(object sender, EventArgs e): Form yüklendiğinde yapılacak işlemleri tanımlar. Bu metot şu anda boş olarak bırakılmıştır.

================================

KUAFÖRLER 3

 **SqlConnection Tanımı**: Con adında bir SqlConnection nesnesi oluşturulur. Bu nesne, belirli bir veritabanına bağlanmak için kullanılacaktır. Bağlantı dizesi bir .mdf dosyasına işaret ediyor ve yerel bir SQL Server veritabanına bağlanmak için kullanılıyor gibi görünüyor.

 **Doktor Sınıfı ve Constructor**: Doktor sınıfı, bir Windows Form'u olarak tanımlanmıştır. Constructor, forma yüklenirken çalışacak olan DoktorlariGoster() yöntemini çağırır.

 **DoktorlariGoster Metodu**: Bu metod, veritabanındaki Berber tablosundaki tüm kayıtları almak ve bunları bir DataGridView'e yüklemek için kullanılır. Bu metot, SqlDataAdapter ve DataSet kullanarak verileri alır ve bunları DataGridView kontrolünde görüntüler. Veritabanı bağlantısı açıldıktan sonra, select \* from Berber sorgusu çalıştırılarak tüm kayıtlar alınır ve DataSet nesnesine doldurulur. Ardından, bu DataSet DataGridView'e atanır ve bağlantı kapatılır.

 **KapatBtn\_Click Metodu**: Bu metot, formun kapatılması butonunun tıklama olayını işler. Basitçe, this.Close() çağrısı kullanılarak form kapatılır.

 **AnaSayfaBtn\_Click Metodu**: Bu metot, ana sayfa butonunun tıklama olayını işler. Ana sayfa formunu gösterir ve mevcut formu gizler.

 **EkleBtn\_Click Metodu**: Bu metot, yeni bir doktor kaydı eklemek için kullanılır. Metot, formdaki giriş kutularından alınan verilerle bir insert sorgusu oluşturur ve bu sorguyu veritabanına gönderir. Eğer gerekli alanlar boşsa, bir uyarı gösterilir.

 **Doktor\_Load Metodu**: Bu metot, form yüklendiğinde çalışacak işlemleri tanımlar. Bu durumda, DoktorlariGoster() metodu çağrılır ve doktor kayıtları DataGridView'de görüntülenir.

 **TemizleBtn\_Click Metodu**: Bu metot, formdaki giriş kutularını temizlemek için kullanılır. Giriş kutularının içeriği basitçe boş bir dize ile değiştirilir.

 **SilBtn\_Click Metodu**: Bu metot, seçilen bir doktor kaydını silmek için kullanılır. Seçili satırın ID'si alınır ve bir onay ile kullanıcıya gösterilir. Kullanıcı evet derse, DELETE sorgusu çalıştırılır ve kayıt silinir.

 **GuncelleBtn\_Click Metodu**: Bu metot, mevcut bir doktor kaydını güncellemek için kullanılır. Güncelleme yapılabilmesi için gerekli olan tüm alanlar dolu olmalıdır. UPDATE sorgusu ile mevcut kayıt güncellenir.

================================

MÜŞTERİLER 4

 **SqlConnection Tanımı**: Con adında bir SqlConnection nesnesi oluşturulur. Bu nesne, belirli bir veritabanına bağlanmak için kullanılacaktır. Bağlantı dizesi bir .mdf dosyasına işaret ediyor ve yerel bir SQL Server veritabanına bağlanmak için kullanılıyor gibi görünüyor.

 **Patient Sınıfı ve Constructor**: Patient sınıfı, bir Windows Form'u olarak tanımlanmıştır. Constructor, forma yüklendiğinde çalışacak olan DisplayPatient() yöntemini çağırır.

 **DisplayPatient Metodu**: Bu metod, veritabanındaki Patient tablosundaki tüm kayıtları almak ve bunları bir DataGridView'e yüklemek için kullanılır. Bu metot, SqlDataAdapter ve DataSet kullanarak verileri alır ve bunları DataGridView kontrolünde görüntüler. Veritabanı bağlantısı açıldıktan sonra, select \* from Patient sorgusu çalıştırılarak tüm kayıtlar alınır ve DataSet nesnesine doldurulur. Ardından, bu DataSet DataGridView'e atanır ve bağlantı kapatılır.

 **EkleBtn\_Click Metodu**: Bu metot, yeni bir hasta kaydı eklemek için kullanılır. Metot, formdaki giriş kutularından alınan verilerle bir insert sorgusu oluşturur ve bu sorguyu veritabanına gönderir. Eğer gerekli alanlar boşsa, bir uyarı gösterilir.

 **Patient\_Load Metodu**: Bu metot, form yüklendiğinde çalışacak işlemleri tanımlar. Bu durumda, DisplayPatient() metodu çağrılır ve hasta kayıtları DataGridView'de görüntülenir.

 **KapatBtn\_Click Metodu**: Bu metot, formun kapatılması butonunun tıklama olayını işler. Basitçe, this.Close() çağrısı kullanılarak form kapatılır.

 **GuncelleBtn\_Click Metodu**: Bu metot, mevcut bir hasta kaydını güncellemek için kullanılır. Güncelleme yapılabilmesi için gerekli olan tüm alanlar dolu olmalıdır. UPDATE sorgusu ile mevcut kayıt güncellenir.

 **SilBtn\_Click Metodu**: Bu metot, seçilen bir hasta kaydını silmek için kullanılır. Seçili satırın ID'si alınır ve bir onay ile kullanıcıya gösterilir. Kullanıcı evet derse, DELETE sorgusu çalıştırılır ve kayıt silinir.

 **TemizleBtn\_Click Metodu**: Bu metot, formdaki giriş kutularını temizlemek için kullanılır. Giriş kutularının içeriği basitçe boş bir dize ile değiştirilir.

 **AnaSayfaBtn\_Click Metodu**: Bu metot, ana sayfa butonunun tıklama olayını işler. Ana sayfa formunu gösterir ve mevcut formu gizler.

 **dataGridView2\_DoubleClick Metodu**: Bu metot, DataGridView'deki bir hücreye çift tıklama olayını işler. Seçilen kaydın verileri ilgili giriş kutularına doldurulur.

================================

BELGİLER 5

 **SqlConnection Tanımı**: Con adında bir SqlConnection nesnesi oluşturulur. Bu nesne, belirli bir veritabanına bağlanmak için kullanılacaktır. Bağlantı dizesi bir .mdf dosyasına işaret ediyor ve yerel bir SQL Server veritabanına bağlanmak için kullanılıyor gibi görünüyor.

 **Diagnosis Sınıfı ve Constructor**: Diagnosis sınıfı, bir Windows Form'u olarak tanımlanmıştır. Constructor, forma yüklendiğinde çalışacak olan DisplayDiagnosis() yöntemini çağırır.

 **DisplayDiagnosis Metodu**: Bu metod, veritabanındaki Diagnosis tablosundaki tüm kayıtları almak ve bunları bir DataGridView'e yüklemek için kullanılır. Bu metot, SqlDataAdapter ve DataSet kullanarak verileri alır ve bunları DataGridView kontrolünde görüntüler. Veritabanı bağlantısı açıldıktan sonra, select \* from Diagnosis sorgusu çalıştırılarak tüm kayıtlar alınır ve DataSet nesnesine doldurulur. Ardından, bu DataSet DataGridView'e atanır ve bağlantı kapatılır.

 **DisplayMusteriAdi Metodu**: Bu metod, belirli bir hastanın adını görüntülemek için kullanılır. ComboBox'tan seçilen hastanın ID'si kullanılarak veritabanından adı alınır ve ilgili TextBox'e yerleştirilir.

 **DisplayPatientId Metodu**: Bu metot, hasta ID'lerini bir ComboBox'a yüklemek için kullanılır. Patient tablosundan ID'leri alır ve bir ComboBox'a yükler.

 **EkleBtn\_Click Metodu**: Bu metot, yeni bir teşhis kaydı eklemek için kullanılır. Metot, formdaki giriş kutularından alınan verilerle bir insert sorgusu oluşturur ve bu sorguyu veritabanına gönderir. Eğer gerekli alanlar boşsa, bir uyarı gösterilir.

 **GuncelleBtn\_Click Metodu**: Bu metot, mevcut bir teşhis kaydını güncellemek için kullanılır. Güncelleme yapılabilmesi için gerekli olan tüm alanlar dolu olmalıdır. UPDATE sorgusu ile mevcut kayıt güncellenir.

 **SilBtn\_Click Metodu**: Bu metot, seçilen bir teşhis kaydını silmek için kullanılır. Seçili satırın ID'si alınır ve bir onay ile kullanıcıya gösterilir. Kullanıcı evet derse, DELETE sorgusu çalıştırılır ve kayıt silinir.

 **AnaSayfaBtn\_Click Metodu**: Bu metot, ana sayfa butonunun tıklama olayını işler. Ana sayfa formunu gösterir ve mevcut formu gizler.

 **ResetBtn\_Click Metodu**: Bu metot, formdaki giriş kutularını temizlemek için kullanılır. Giriş kutularının içeriği basitçe boş bir dize ile değiştirilir.

 **comboBox1\_SelectionChangeCommitted Metodu**: Bu metot, ComboBox'tan bir öğe seçildiğinde tetiklenir ve seçilen hastanın adını görüntülemek için DisplayMusteriAdi metodunu çağırır.

 **dataGridView3\_DoubleClick Metodu**: Bu metot, DataGridView'deki bir hücreye çift tıklama olayını işler. Seçilen kaydın verileri ilgili giriş kutularına doldurulur