

OTO KİRALAMA PROGRAMI

(C# SHARP ACCESS VT DATAGRIDVIEW MDIFORM)

UYGULAMA SORUSU : Bir araç kiralama firmasının işlemlerini takip eden bir programdır. Ana form MDIForm olarak tasarlanmıştır. Diğer 3 form (Müşteri formu, Araç Formu ve Kiralama Formu) bu anaform içinde açılmaktadır. Müşteri Formu ve Araç Formunda kayıt işlemleri yapılmaktadır. Kiralama formunda ise girilen müşteri ve araç bilgilerini kullanarak kiralama işlemlerinin yapıldığı bir yerdir.

The main window of the OTO KİRALAMA PROGRAMI displays two sub-forms. The MÜŞTERİ KAYIT FORMU (Customer Registration Form) is on the left, and the ARAÇ KAYIT FORMU (Vehicle Registration Form) is on the right. The MÜŞTERİ KAYIT FORMU includes fields for Müşteri No, TC No, Ad Soyad, Telefon, Adres, and E-Posta. The ARAÇ KAYIT FORMU includes fields for Oto Kayıt No, Plaka, Marka, Model, Yıl, Vites, Yakıt, Renk, and Durumu. Both forms have a table at the bottom to display the registered data.

MusNo	TC No	Ad Soyad	Telefon
1	123456	Ali Taşkıran	555 225 1214
2	254156	Nuriye Haktan	535 458 78
3	584574	Ziya Uzunlu	505 658 74

OtoKayıtNo	OtoPlaka	OtoMarka	OtoModel	OtoYıl
1	35AB125	Renault	Clio	2022
2	34CDE85	BMW	i5	2025
3	35AA457	Fiat	Egea	2021

The MÜŞTERİ KAYIT FORMU displays the registration details for a customer. The form includes fields for Müşteri No, TC No, Ad Soyad, Telefon, Adres, and E-Posta. The data is as follows:

MusNo	TcNo	AdSoyad	Telefon
1	123456	Ali Taşkıran	555 225 1214
2	254156	Nuriye Haktan	535 458 78
3	584574	Ziya Uzunlu	505 658 74

The ARAÇ KİRALAMA FORMU displays the rental details for a vehicle. The form includes fields for Kayıt No, Müşteri Ad Soyadı, Araç Plaka, Başlama Tarihi, Bitiş Tarihi, and Tutan. The data is as follows:

KayıtNo	AdSoyad	AracPlaka	BaslamaTari
3	Ziya Uzunlu	35AA457	13.05.2025
2	Ali Taşkıran	34CDE85	21.05.2025

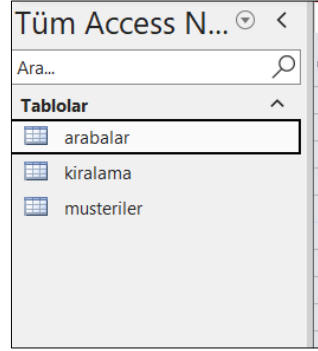
ACCESS VERİ TABANI YAPISI

otokiralama.accdb

Alan Adı	Veri Türü
OtoKayitNo	Otomatik Sayı
OtoPlaka	Kısa Metin
OtoMarka	Kısa Metin
OtoModel	Kısa Metin
OtoYil	Sayı
OtoVites	Kısa Metin
OtoYakit	Kısa Metin
OtoRenk	Kısa Metin
OtoDurumu	Evet/Hayır

Alan Adı	Veri Türü
MusNo	Otomatik Sayı
TcNo	Kısa Metin
AdSoyad	Kısa Metin
Telefon	Kısa Metin
Adres	Kısa Metin
Eposta	Kısa Metin

Alan Adı	Veri Türü
KayitNo	Otomatik Sayı
AdSoyad	Kısa Metin
AracPlaka	Kısa Metin
BaslamaTarihi	Tarih/Saat
BitisTarihi	Tarih/Saat
Tutari	Sayı



KODLAR

Vtbaglan.cs

```
rp_Otokiralama_AccessVT_Datagridview  Csharp_Otokiralama_AccessVT_Datagridview.vtbaglan  baglan0
1  using System;
2  using System.Collections.Generic;
3  using System.Data.OleDb;
4  using System.Linq;
5  using System.Text;
6  using System.Threading.Tasks;
7
8  namespace Csharp_Otokiralama_AccessVT_Datagridview
9  {
10     class vtbaglan
11     {
12         public OleDbConnection baglan()
13         {
14             OleDbConnection baglanti = new OleDbConnection("Provider=Microsoft.ACE.OLEDB.12.0;Data " +
15                 "Source=otokiralama.accdb;Persist Security Info=True");
16             baglanti.Open();
17             return (baglanti);
18         }
19     }
20 }
21
```

FORM1.CS (MDI FORM) KODLARI

```
11
12 namespace Csharp_Otokiralama_AccessVT_Datagridview
13 {
14     public partial class Form1: Form
15     {
16         public Form1()
17         {
18             InitializeComponent();
19         }
20
21         private void toolStripButton2_Click(object sender, EventArgs e)
22         {
23             Form2 frm2 = new Form2();
24             frm2.MdiParent = this;
25             frm2.Show();
26         }
27
28         private void toolStripButton1_Click(object sender, EventArgs e)
29         {
30             Form4 frm4 = new Form4();
31             frm4.MdiParent = this;
32             frm4.Show();
33         }
34
35         private void toolStripButton3_Click(object sender, EventArgs e)
36         {
37             Form3 frm3 = new Form3();
38             frm3.MdiParent = this;
39             frm3.Show();
40         }
41     }
42 }
```

FORM2.CS (ARAÇ KAYIT FORMU) KODLARI

```
11
12 namespace Csharp_Otokiralama_AccessVT_Datagridview
13 {
14     public partial class Form2: Form
15     {
16         vtbaglan AccessBaglanti = new vtbaglan();
17         DataTable AccessTablo = new DataTable();
18         OleDbCommand SqlKomut = new OleDbCommand();
19         OleDbDataAdapter Adaptor = new OleDbDataAdapter();
20
21         public Form2()
22         {
23             InitializeComponent();
24             AracListele();
25         }
26     }
```

```

26
27 void AracListele()
28 {
29     try
30     {
31         string sqlCumle = "Select * from arabalar";
32         Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
33         AccessTablo.Clear();
34         Adaptor.Fill(AccessTablo);
35         dataGridView1.DataSource = AccessTablo;
36         toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
37     }
38     catch
39     {
40         MessageBox.Show("Arabalar listelenirken bir hata oluřtu");
41     }
42 }
43
44 private void toolStripButton4_Click(object sender, EventArgs e)
45 {
46     // Listeleme
47     AracListele();
48 }

```

```

49
50 private void dataGridView1_SelectionChanged(object sender, EventArgs e)
51 {
52     if (dataGridView1.SelectedRows.Count > 0)
53     {
54         label2.Text = dataGridView1.CurrentRow.Cells[0].Value.ToString();
55         textBox1.Text = dataGridView1.CurrentRow.Cells[1].Value.ToString();
56         textBox2.Text = dataGridView1.CurrentRow.Cells[2].Value.ToString();
57         textBox3.Text = dataGridView1.CurrentRow.Cells[3].Value.ToString();
58         textBox4.Text = dataGridView1.CurrentRow.Cells[4].Value.ToString();
59         comboBox1.Text = dataGridView1.CurrentRow.Cells[5].Value.ToString();
60         comboBox2.Text = dataGridView1.CurrentRow.Cells[6].Value.ToString();
61         textBox5.Text = dataGridView1.CurrentRow.Cells[7].Value.ToString();
62         checkBox1.Checked = Convert.ToBoolean(dataGridView1.CurrentRow.Cells[8].Value);
63     }
64 }
65
66
67 private void checkBox1_CheckedChanged(object sender, EventArgs e)
68 {
69     if (checkBox1.Checked) checkBox1.Text = "Dolu"; else checkBox1.Text = "Boř";
70 }
71

```

```

private void toolStripButton5_Click(object sender, EventArgs e)
{
    // Form Temizle
    label2.Text = "";
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox4.Clear();
    textBox5.Clear();
    comboBox1.Text = "";
    comboBox2.Text = "";
    textBox1.Focus();
}

```

```

private void toolStripButton1_Click(object sender, EventArgs e)
{
    // Yeni Ekle
    try
    {
        string sqlEKLE = "Insert Into arabalar(OtoPlaka,OtoMarka,OtoModel,OtoYil,OtoVites,OtoYakit,OtoRenk,OtoDurumu) " +
            "values(@OtoPlaka,@OtoMarka,@OtoModel,@OtoYil,@OtoVites,@OtoYakit,@OtoRenk,@OtoDurumu)";
        SqlKomut = new OleDbCommand(sqlEKLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@OtoPlaka", textBox1.Text);
        SqlKomut.Parameters.AddWithValue("@OtoMarka", textBox2.Text);
        SqlKomut.Parameters.AddWithValue("@OtoModel", textBox3.Text);
        SqlKomut.Parameters.AddWithValue("@OtoYil", textBox4.Text);
        SqlKomut.Parameters.AddWithValue("@OtoVites", comboBox1.Text);
        SqlKomut.Parameters.AddWithValue("@OtoYakit", comboBox2.Text);
        SqlKomut.Parameters.AddWithValue("@OtoRenk", textBox5.Text);
        SqlKomut.Parameters.AddWithValue("@OtoDurumu", checkBox1.Checked);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Yeni bir Araç Kaydı eklendi.");
        Araclistele();
    }
    catch { MessageBox.Show("Kayıt yapılırken bir hata oluştu."); }
}

```

```

private void toolStripButton2_Click(object sender, EventArgs e)
{
    // Güncelleme
    try
    {
        string sqlGUNCELLE = "Update arabalar set OtoPlaka=@OtoPlaka,OtoMarka=@OtoMarka,OtoModel=@OtoModel," +
            "OtoYil=@OtoYil,OtoVites=@OtoVites,OtoYakit=@OtoYakit,OtoRenk=@OtoRenk,OtoDurumu=@OtoDurumu " +
            "where OtoKayitNo=@OtoKayitNo";

        SqlKomut = new OleDbCommand(sqlGUNCELLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@OtoPlaka", textBox1.Text);
        SqlKomut.Parameters.AddWithValue("@OtoMarka", textBox2.Text);
        SqlKomut.Parameters.AddWithValue("@OtoModel", textBox3.Text);
        SqlKomut.Parameters.AddWithValue("@OtoYil", textBox4.Text);
        SqlKomut.Parameters.AddWithValue("@OtoVites", comboBox1.Text);
        SqlKomut.Parameters.AddWithValue("@OtoYakit", comboBox2.Text);
        SqlKomut.Parameters.AddWithValue("@OtoRenk", textBox5.Text);
        SqlKomut.Parameters.AddWithValue("@OtoDurumu", checkBox1.Checked);
        SqlKomut.Parameters.AddWithValue("@OtoKayitNo", label2.Text);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Yeni bir Araç Kaydı Güncellendi.");
        Araclistele();
    }
    catch { MessageBox.Show("Kayıt Güncellenirken bir hata oluştu."); }
}

```

```

private void button1_Click(object sender, EventArgs e)
{
    // Arama-Filtreleme
    try
    {
        string sqlCumle = "Select * from arabalar where otoplaka like '%" + textBox1.Text + "%'";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        AccessTablo.Clear();
        Adaptor.Fill(AccessTablo);
        dataGridView1.DataSource = AccessTablo;
        toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
    }
    catch
    {
        MessageBox.Show("Arabalar listelenirken bir hata oluştu.");
    }
}

```

```

private void toolStripButton3_Click(object sender, EventArgs e)
{
    // Kayıt Sil

    DialogResult SOR = new DialogResult();

    SOR = MessageBox.Show(" Bu kaydı silmek istiyor musun?", "UYARI", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);

    if (SOR == DialogResult.Yes)
    {
        try
        {
            string sqlSIL = "Delete from arabalar Where OtoKayitNo=@OtoKayitNo";

            SqlKomut = new OleDbCommand(sqlSIL, AccessBaglanti.baglan());
            SqlKomut.Parameters.AddWithValue("@OtoKayitNo", label2.Text);
            SqlKomut.Connection = AccessBaglanti.baglan();
            SqlKomut.ExecuteNonQuery();
            SqlKomut.Connection.Close();
            MessageBox.Show("Kayıt silindi.");
            Aracliste();
        }
        catch { MessageBox.Show("Kayıt silme işlemi yapılırken bir hata oluştu."); }
    }
}

```

FORM3.CS (MÜŞTERİ KAYIT FORMU) KODLARI

```

public partial class Form3: Form
{
    vtbaglan AccessBaglanti = new vtbaglan();
    DataTable AccessTablo = new DataTable();
    OleDbCommand SqlKomut = new OleDbCommand();
    OleDbDataAdapter Adaptor = new OleDbDataAdapter();

    public Form3()
    {
        InitializeComponent();
        MusteriListele();
    }

    void MusteriListele()
    {
        try
        {
            string sqlCumle = "Select * from musteriler";
            Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
            AccessTablo.Clear();
            Adaptor.Fill(AccessTablo);
            dataGridView1.DataSource = AccessTablo;
            toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
        }
        catch
        {
            MessageBox.Show("Müşteriler listelenirken bir hata oluştu");
        }
    }
}

```

```

private void toolStripButton4_Click(object sender, EventArgs e)
{
    //Listele
    MusteriListele();
}

private void dataGridView1_SelectionChanged(object sender, EventArgs e)
{
    if (dataGridView1.SelectedRows.Count > 0)
    {
        label2.Text = dataGridView1.CurrentRow.Cells[0].Value.ToString();
        textBox1.Text = dataGridView1.CurrentRow.Cells[1].Value.ToString();
        textBox2.Text = dataGridView1.CurrentRow.Cells[2].Value.ToString();
        textBox3.Text = dataGridView1.CurrentRow.Cells[3].Value.ToString();
        textBox4.Text = dataGridView1.CurrentRow.Cells[4].Value.ToString();
        textBox5.Text = dataGridView1.CurrentRow.Cells[5].Value.ToString();
    }
}

```

```

private void toolStripButton1_Click(object sender, EventArgs e)
{
    // Yeni Ekle
    try
    {
        string sqlEKLE = "Insert Into musteriler(TcNo,AdSoyad,Telefon,Adres,Eposta) " +
            "values(@TcNo,@AdSoyad,@Telefon,@Adres,@Eposta)";
        SqlKomut = new OleDbCommand(sqlEKLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@TcNo", textBox1.Text);
        SqlKomut.Parameters.AddWithValue("@AdSoyad", textBox2.Text);
        SqlKomut.Parameters.AddWithValue("@Telefon", textBox3.Text);
        SqlKomut.Parameters.AddWithValue("@Adres", textBox4.Text);
        SqlKomut.Parameters.AddWithValue("@Eposta", textBox5.Text);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Yeni bir Müşteri Kaydı eklendi.");
        MusteriListele();
    }
    catch { MessageBox.Show("Kayıt yapılırken bir hata oluştu."); }
}

```

```

private void toolStripButton5_Click(object sender, EventArgs e)
{
    // Form Temizle
    label2.Text = "";
    textBox1.Clear();
    textBox2.Clear();
    textBox3.Clear();
    textBox4.Clear();
    textBox5.Clear();
    textBox1.Focus();
}

```



```

private void toolStripButton2_Click(object sender, EventArgs e)
{
    // Güncelleme
    try
    {
        string sqlGUNCELLE = "Update musteriler Set TcNo=@TcNo,AdSoyad=@AdSoyad,Telefon=@Telefon," +
            "Adres=@Adres, Eposta=@Eposta where MusNo=@MusNo";

        SqlKomut = new OleDbCommand(sqlGUNCELLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@TcNo", textBox1.Text);
        SqlKomut.Parameters.AddWithValue("@AdSoyad", textBox2.Text);
        SqlKomut.Parameters.AddWithValue("@Telefon", textBox3.Text);
        SqlKomut.Parameters.AddWithValue("@Adres", textBox4.Text);
        SqlKomut.Parameters.AddWithValue("@Eposta", textBox5.Text);
        SqlKomut.Parameters.AddWithValue("@MusNo", label2.Text);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Yeni bir Müşteri Kaydı Güncelledi.");
        MusteriListele();
    }
    catch { MessageBox.Show("Kayıt Güncellenirken bir hata oluştu."); }
}

```

```

private void toolStripButton3_Click(object sender, EventArgs e)
{
    // Kayıt Sil
    DialogResult SOR = new DialogResult();
    SOR = MessageBox.Show(" Bu kaydı silmek istiyor musun?", "UYARI", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
    if (SOR == DialogResult.Yes)
    {
        try
        {
            string sqlSIL = "Delete from musteriler Where MusNo=@MusNo";
            SqlKomut = new OleDbCommand(sqlSIL, AccessBaglanti.baglan());
            SqlKomut.Parameters.AddWithValue("@MusNo", label2.Text);
            SqlKomut.Connection = AccessBaglanti.baglan();
            SqlKomut.ExecuteNonQuery();
            SqlKomut.Connection.Close();
            MessageBox.Show("Kayıt silindi.");
            MusteriListele();
        }
        catch { MessageBox.Show("Kayıt silme işlemi yapılırken bir hata oluştu."); }
    }
}

```

```

private void button1_Click(object sender, EventArgs e)
{
    // Arama-Filtreleme
    try
    {
        string sqlCumle = "Select * from musteriler where adsoyad like '%" + textBox2.Text + "%'";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        AccessTablo.Clear();
        Adaptor.Fill(AccessTablo);
        dataGridView1.DataSource = AccessTablo;
        toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
    }
    catch
    {
        MessageBox.Show("Müşteriler listelenirken bir hata oluştu");
    }
}

```


FORM4.CS (KİRALAMA FORMU) KODLARI

```
using System;
using System.Collections.Generic;
using System.ComponentModel;
using System.Data;
using System.Data.OleDb;
using System.Drawing;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Windows.Forms;
using static System.Windows.Forms.VisualStyles.VisualStyleElement;

namespace Csharp_Otokiralama_AccessVT_Datagridview
{
    4 references
    public partial class Form4: Form
    {
        vtbaglan AccessBaglanti = new vtbaglan();
        DataTable AccessTablo = new DataTable();
        OleDbCommand SqlKomut = new OleDbCommand();
        OleDbDataAdapter Adaptor = new OleDbDataAdapter();

        1 reference
        public Form4()
        {
            InitializeComponent();
            KiralListele();
            araclistele();
            musterilistele();
        }
    }
}
```

```
void KiralListele()
{
    try
    {
        string sqlCumle = "Select * from kiralama";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        AccessTablo.Clear();
        Adaptor.Fill(AccessTablo);
        dataGridView1.DataSource = AccessTablo;
        toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
    }
    catch
    {
        MessageBox.Show("Kiralamalar listelenirken bir hata oluştu");
    }
}
```

```
private void dataGridView1_SelectionChanged(object sender, EventArgs e)
{
    if (dataGridView1.SelectedRows.Count > 0)
    {
        label2.Text = dataGridView1.CurrentRow.Cells[0].Value.ToString();
        comboBox1.Text = dataGridView1.CurrentRow.Cells[1].Value.ToString();
        comboBox2.Text = dataGridView1.CurrentRow.Cells[2].Value.ToString();
        dateTimePicker1.Value = Convert.ToDateTime(dataGridView1.CurrentRow.Cells[3].Value);
        dateTimePicker2.Value = Convert.ToDateTime(dataGridView1.CurrentRow.Cells[4].Value);
        maskedTextBox1.Text = dataGridView1.CurrentRow.Cells[5].Value.ToString();
    }
}
```

```
1 reference
void musterilistele()
{
    try
    {
        DataTable dt = new DataTable();
        dt.Columns.Add("adsoyad", typeof(string));
        string sqlCumle = "Select adsoyad from musteriler";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        dt.Clear();
        Adaptor.Fill(dt);
        comboBox1.DataSource = dt;
        comboBox1.DisplayMember = "adsoyad";
        comboBox1.ValueMember = "adsoyad";
    }
    catch
    {
        MessageBox.Show("Müşteri adsoyad listelenirken bir hata oluştu");
    }
}
```

```
1 reference
void araclistele()
{
    try
    {
        DataTable dt = new DataTable();
        dt.Columns.Add("otoplaka", typeof(string));
        string sqlCumle = "Select otoplaka from arabalar";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        dt.Clear();
        Adaptor.Fill(dt);
        comboBox2.DataSource = dt;
        comboBox2.DisplayMember = "otoplaka";
        comboBox2.ValueMember = "otoplaka";
    }
    catch
    {
        MessageBox.Show("Araç plakası listelenirken bir hata oluştu");
    }
}
```

```

1 reference
private void toolStripButton1_Click(object sender, EventArgs e)
{
    // Yeni Ekle

    try
    {
        string sqlEKLE = "Insert Into kiralama(AdSoyad,AracPlaka,BaslamaTarihi,BitisTarihi,Tutari) " +
            "values(@AdSoyad,@AracPlaka,@BaslamaTarihi,@BitisTarihi,@Tutari)";
        SqlKomut = new OleDbCommand(sqlEKLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@AdSoyad", comboBox1.Text);
        SqlKomut.Parameters.AddWithValue("@AracPlaka", comboBox2.Text);
        SqlKomut.Parameters.AddWithValue("@BaslamaTarihi", dateTimePicker1.Value);
        SqlKomut.Parameters.AddWithValue("@BitisTarihi", dateTimePicker2.Value);
        SqlKomut.Parameters.AddWithValue("@Tutari", maskedTextBox1.Text);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Yeni bir Kiralama Kaydı eklendi.");
        KiraListele();
    }
    catch { MessageBox.Show("Kayıt yapılırken bir hata oluştu."); }
}

```

```

1 reference
private void toolStripButton2_Click(object sender, EventArgs e)
{
    // Güncelle

    try
    {
        string sqlGÜNCELLE = "Update kiralama set AdSoyad= @AdSoyad,AracPlaka=@AracPlaka,BaslamaTarihi=@BaslamaTarihi," +
            "BitisTarihi=@BitisTarihi,Tutari=@Tutari where KayitNo=@KayitNo";

        SqlKomut = new OleDbCommand(sqlGÜNCELLE, AccessBaglanti.baglan());
        SqlKomut.Parameters.AddWithValue("@AdSoyad", comboBox1.Text);
        SqlKomut.Parameters.AddWithValue("@AracPlaka", comboBox2.Text);
        SqlKomut.Parameters.AddWithValue("@BaslamaTarihi", dateTimePicker1.Value);
        SqlKomut.Parameters.AddWithValue("@BitisTarihi", dateTimePicker2.Value);
        SqlKomut.Parameters.AddWithValue("@Tutari", maskedTextBox1.Text);
        SqlKomut.Parameters.AddWithValue("@KayitNo", label2.Text);

        SqlKomut.Connection = AccessBaglanti.baglan();
        SqlKomut.ExecuteNonQuery();
        SqlKomut.Connection.Close();
        MessageBox.Show("Kiralama Kaydı Güncellendi.");
        KiraListele();
    }
    catch { MessageBox.Show("Kayıt yapılırken bir hata oluştu."); }
}

```

```

1 reference
private void toolStripButton3_Click(object sender, EventArgs e)
{
    // Kayıt Sil
    DialogResult SOR = new DialogResult();
    SOR = MessageBox.Show(" Bu kaydı silmek istiyor musun?", "UYARI", MessageBoxButtons.YesNo, MessageBoxIcon.Warning);
    if (SOR == DialogResult.Yes)
    {
        try
        {
            string sqlSIL = "Delete from kiralama Where KayitNo=@KayitNo";
            SqlKomut = new OleDbCommand(sqlSIL, AccessBaglanti.baglan());
            SqlKomut.Parameters.AddWithValue("@KayitNo", label2.Text);
            SqlKomut.Connection = AccessBaglanti.baglan();
            SqlKomut.ExecuteNonQuery();
            SqlKomut.Connection.Close();
            MessageBox.Show("Kayıt silindi.");
            KiraListele();
        }
        catch { MessageBox.Show("Kayıt silme işlemi yapılırken bir hata oluştu."); }
    }
}

```

```

1 reference
private void toolStripButton5_Click(object sender, EventArgs e)
{
    // Form Temizle
    label2.Text = "";
    comboBox1.Text = "";
    comboBox2.Text = "";
    dateTimePicker1.Value = DateTime.Now;
    dateTimePicker2.Value = DateTime.Now;
    maskedTextBox1.Clear();
    comboBox1.Focus();
}

```

1 reference

```
private void button2_Click(object sender, EventArgs e)
{
    // Arama filtreleme
    try
    {
        string sqlCumle = "Select * from kiralama where adsoyad like '%" + comboBox2.Text + "%'";
        Adaptor = new OleDbDataAdapter(sqlCumle, AccessBaglanti.baglan());
        AccessTablo.Clear();
        Adaptor.Fill(AccessTablo);
        dataGridView1.DataSource = AccessTablo;
        toolStripStatusLabel2.Text = AccessTablo.Rows.Count.ToString();
    }
    catch
    {
        MessageBox.Show("Kiralamalar listelenirken bir hata oluştu");
    }
}
```

1 reference

```
private void button1_Click(object sender, EventArgs e)
{
    // Ücret Hesapla
    int gunFarki = (dateTimePicker2.Value - dateTimePicker1.Value).Days;
    maskedTextBox1.Text = (gunFarki * 1000).ToString();
}
```