

```
using System;
using System.Collections.Generic;
using System.Linq;
using System.Text;
using System.Threading.Tasks;
using System.Data.OleDb;
using System.Data;
using System.Windows.Forms;
using System.Collections;
```

```
namespace SınavListesi
```

```
{
```

```
    class VeriTabaniIslemleri
```

```
    {
```

```
        OleDbConnection baglanti;
```

```
        public VeriTabaniIslemleri(string ConnectionString)
```

```
        {
```

```
            baglanti = new OleDbConnection(ConnectionString);
```

```
        }
```

```
        public void OgrenciSil()
```

```
        {
```

```
            OleDbCommand komut = new OleDbCommand("Delete from Ogrenciler", baglanti);
```

```
            baglanti.Open();
```

```
            komut.ExecuteNonQuery();
```

```
            baglanti.Close();
```

```
        }
```

```
        public bool OgrenciVarMi(string OgrNo, string DersNo)
```

```
{  
    bool BuMetodKullandi = false;  
  
    OleDbCommand komut = new OleDbCommand("Select Ad from Ogrenciler where  
((Ogrenciler.OgrNo=?)and(Ogrenciler.DersNo=?));", baglanti);  
  
    if (baglanti.State == ConnectionState.Closed)  
    {  
        baglanti.Open();  
  
        BuMetodKullandi = true;  
    }  
  
    komut.Parameters.AddWithValue("OgrNo", OgrNo);  
    komut.Parameters.AddWithValue("DersNo", DersNo);  
  
  
    OleDbDataReader oku = komut.ExecuteReader();  
  
    if (oku.HasRows)  
    {  
        oku.Close();  
        komut.Dispose();  
  
        if (BuMetodKullandi)  
            baglanti.Close();  
  
        return true;  
    }  
    else  
    {  
        oku.Close();  
        komut.Dispose();  
  
        if (BuMetodKullandi)  
            baglanti.Close();  
    }  
}
```

```

        return false;
    }

}

public bool DersVarMi(string DersNo)
{
    bool BuMetodKullandi = false;

    OleDbCommand komut = new OleDbCommand("Select DersNo from Dersler where
(Dersler.DersNo=?);", baglanti);

    if (baglanti.State == ConnectionState.Closed)
    {
        baglanti.Open();

        BuMetodKullandi = true;
    }

    komut.Parameters.AddWithValue("DersNo", DersNo);

    OleDbDataReader oku = komut.ExecuteReader();

    if (oku.HasRows)
    {
        oku.Close();

        komut.Dispose();

        if (BuMetodKullandi)
            baglanti.Close();

        return true;
    }
    else

```

```

    {
        oku.Close();
        komut.Dispose();
        if (BuMetodKullandi)
            baglanti.Close();
        return false;
    }

}

public void DersEkle(string DersNo, string DersAdi = "Dersin Adi Verilmemiş")
{
    if (DersVarMi(DersNo) == false)
    {
        bool BuMetodKullandi = false;

        OleDbCommand komut = new OleDbCommand("Insert into Dersler(DersNo,DersAdi)
Values(?,?);", baglanti);

        if (baglanti.State == ConnectionState.Closed)
        {
            baglanti.Open();

            BuMetodKullandi = true;
        }

        komut.Parameters.AddWithValue("DersNo", DersNo);
        komut.Parameters.AddWithValue("DersAdi", DersAdi);
        komut.ExecuteNonQuery();
        komut.Dispose();
        if (BuMetodKullandi)
            baglanti.Close();
    }
}

```

```

    }
}

public string OgrYukle(string[] veri, ToolStripProgressBar p)
{
    string HataliSatirlar = "Dosyadan veritabanına aktarılamayan satırlar\n";
    baglanti.Open();
    p.Maximum = veri.Length;
    for (int i = 1; i < veri.Length; i++)
    {
        p.Value = i;

        OleDbCommand komut = new OleDbCommand("Insert into
Ogrenciler(OgrNo,Ad,Soyad,DersNo,SinifNo) Values(?,?,?,?,?);", baglanti);

        string[] veri2 = veri[i].Replace("\r", ' ').Split(';');
        if (veri2.Length >= 4)
        {
            if (OgrenciVarMi(veri2[0], veri2[3]) == false)
            {
                komut.Parameters.AddWithValue("OgrNo", veri2[0]);
                komut.Parameters.AddWithValue("Ad", veri2[1]);
                komut.Parameters.AddWithValue("Soyad", veri2[2]);
                komut.Parameters.AddWithValue("DersNo", veri2[3]);
                komut.Parameters.AddWithValue("SinifNo", "1");
                DersEkle(veri2[3]);
                komut.ExecuteNonQuery();
                komut.Dispose();
            }
            else
            {
                HataliSatirlar = HataliSatirlar + i + "\t" + "Tekrarlı Veri  " + veri[i] + "\n";
            }
        }
    }
}

```

```

        }
    }
    else
    {
        HataliSatirlar = HataliSatirlar + i + "\t" + "Eksik Veri  " + veri[i] + "\n";
    }

}

baglanti.Close();

return HataliSatirlar;
}

public int DersiAlanOgrenciSayisi(string DersNo)
{
    bool BuMetodKullandi = false;

    OleDbCommand komut = new OleDbCommand("Select Count (*) from Ogrenciler Where
(Ogrenciler.DersNo=?);", baglanti);

    if (baglanti.State == ConnectionState.Closed)
    {
        baglanti.Open();

        BuMetodKullandi = true;
    }

    komut.Parameters.AddWithValue("DersNo", DersNo);

    int toplam = Convert.ToInt16(komut.ExecuteScalar());

    komut.Dispose();

    if (BuMetodKullandi)
        baglanti.Close();

    return toplam;
}

```

```

public string DersListesi()
{
    bool BuMetodKullandi = false;

    OleDbCommand komut = new OleDbCommand("Select * from Dersler ;", baglanti);

    if (baglanti.State == ConnectionState.Closed)
    {
        baglanti.Open();

        BuMetodKullandi = true;
    }

    OleDbDataReader oku = komut.ExecuteReader();

    string Cevap = "";

    while (oku.Read())
    {
        Cevap = Cevap + oku["DersNo"] + ":" + oku["DersAdi"] + ":" +
DersiAlanOgrenciSayisi(oku["DersNo"].ToString()) + "\n";
    }

    Cevap = Cevap.Remove(Cevap.Length - 1);

    oku.Close();

    komut.Dispose();

    if (BuMetodKullandi)
        baglanti.Close();

    return Cevap;
}

public string SinifListesi()
{
    bool BuMetodKullandi = false;

    OleDbCommand komut = new OleDbCommand("Select * from Siniflar ;", baglanti);

    if (baglanti.State == ConnectionState.Closed)
    {

```

```

        baglanti.Open();

        BuMetodKullandi = true;
    }

    OleDbDataReader oku = komut.ExecuteReader();

    string Cevap = "";
    while (oku.Read())
    {
        Cevap = Cevap + oku["SinifNo"] + ":" + oku["SinifAdi"] + ":" + oku["Kontejan"] + "\n";
    }

    Cevap = Cevap.Remove(Cevap.Length - 1);
    oku.Close();
    komut.Dispose();
    if (BuMetodKullandi)
        baglanti.Close();
    return Cevap;
}

public void SinifDoldur(string DersNo, ArrayList Siniflar)
{
    ArrayList Ogrenciler = new ArrayList();

    OleDbCommand komut = new OleDbCommand("Select OgrNo from Ogrenciler Where (Ogrenciler.DersNo=?);", baglanti);

    baglanti.Open();

    komut.Parameters.AddWithValue("DersNo", DersNo);

    OleDbDataReader oku = komut.ExecuteReader();

    while (oku.Read())
    {
        Ogrenciler.Add(oku["OgrNo"]);
    }

    oku.Close();
}

```



```

Random r = new Random();
for(int i = 0; i < Siniflar.Count; i++)
{
    string[] Sinif = (string[])Siniflar[i];
    int kontejan = Convert.ToInt16(Sinif[2]);
    for(int j = 0; j < kontejan; j++)
    {
        if (Ogrenciler.Count == 0)
            break;

        int satirno = r.Next(Ogrenciler.Count);
        string OgrNo = Ogrenciler[satirno].ToString();
        Ogrenciler.RemoveAt(satirno);

        OleDbCommand komut2 = new OleDbCommand("UPDATE Ogrenciler Set SinifNo=? Where
((Ogrenciler.OgrNo=?) and (Ogrenciler.DersNo=?));", baglanti);

        komut2.Parameters.AddWithValue("SinifNo", Sinif[0]);
        komut2.Parameters.AddWithValue("OgrNo", OgrNo);
        komut2.Parameters.AddWithValue("DersNo", DersNo);
        komut2.ExecuteNonQuery();
        komut2.Dispose();
    }
}

baglanti.Close();
}

public string SinifListesi2()
{
    bool BuMetodKullandi = false;
    OleDbCommand komut = new OleDbCommand("Select * from Siniflar ;", baglanti);
    if (baglanti.State == ConnectionState.Closed)
    {

```

```

        baglanti.Open();

        BuMetodKullandi = true;
    }

    OleDbDataReader oku = komut.ExecuteReader();

    string Cevap = "";

    while (oku.Read())
    {
        Cevap = Cevap + oku["SinifAdi"]+" ";

    }

    Cevap = Cevap.Remove(Cevap.Length - 1);

    oku.Close();

    komut.Dispose();

    if (BuMetodKullandi)
        baglanti.Close();

    return Cevap;
}

public string DersListesi2()
{
    bool BuMetodKullandi = false;

    OleDbCommand komut = new OleDbCommand("Select * from Dersler ;", baglanti);

    if (baglanti.State == ConnectionState.Closed)
    {
        baglanti.Open();

        BuMetodKullandi = true;
    }

    OleDbDataReader oku = komut.ExecuteReader();

    string Cevap = "";

```

```
while (oku.Read())
{
    Cevap = Cevap + oku["DersNo"] + oku["DersAdi"] + " ";
}
Cevap = Cevap.Remove(Cevap.Length - 1);
oku.Close();
komut.Dispose();
if (BuMetodKullandi)
    baglanti.Close();
return Cevap;
}
}
}
```