**ListBox Özellikleri**:

Items.Add Özelliği: listBox’a eleman eklemek için kullanılır.

Örnek:

private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

}

Text Özelliği: listBox’ın girilen , seçilen veya gözüken elemanı temsil eder.

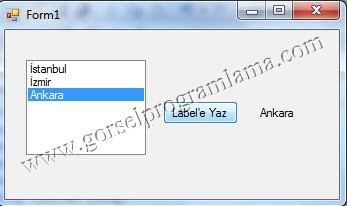
Örnek:

private void button1\_Click(object sender, EventArgs e)

{

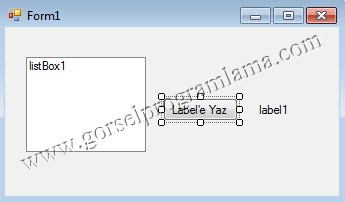
label1.Text =listBox1.Text;

}



Items Özelliği: listBox’ın eleman içeriğini öğrenmek için kullanılır.

Örnek:



private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

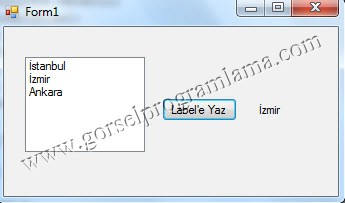
}

private void button1\_Click(object sender, EventArgs e)

{

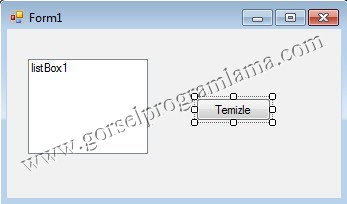
label1.Text = listBox1.Items[1].ToString();

}



Items.Clear Özelliği:listBox’ı temizlemek  için kullanılır.

Örnek:



private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

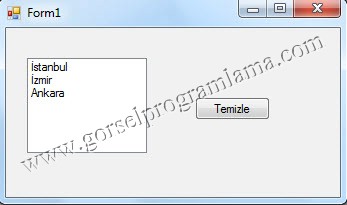
}

private void button1\_Click(object sender, EventArgs e)

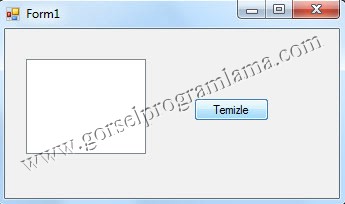
{

listBox1.Items.Clear();

}

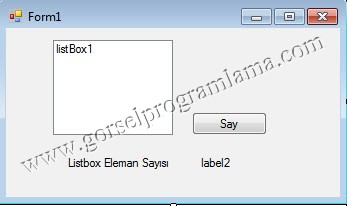


Temizle tıklanınca sonuç.



Items.Count Özelliği: listBox’taki  eleman sayısını verir.

Örnek:



private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

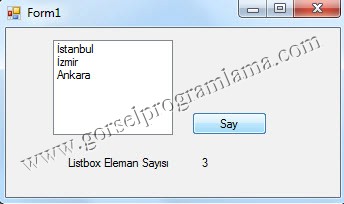
}//Daha fazla bilgi için : www.gorselprogramlama.com

private void button1\_Click(object sender, EventArgs e)

{

label1.Text= listBox1.Items.Count.ToString();

}



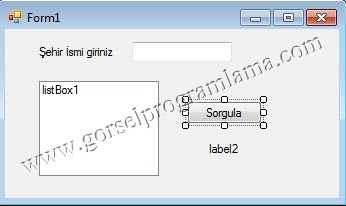
Daha fazla bilgi için : [www.gorselprogramlama.com](http://www.gorselprogramlama.com/)

Items.IndexOf  Özelliği:listbox’ta eleman aramak için kullanılır.Eğer belirtilen eleman var ise sıra numarasını ,  yok ise -1  değerini döndürür.

**Not:** Elemanların sıra numaralarının sıfırdan başladığını unutmayın.Yani 0.eleman–>İstanbul 1.eleman–>İzmir gibi.

**İpucu :** Mesela  ekrana elaman sayısını yazdırmak istiyorsunuz. 0.elemanı 1. eleman olarak yazdırmak istiyorsanız.Bunu için  komutsatırı+1 derseniz  istediğinizi yapmış olursunuz.

Örnek:



private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

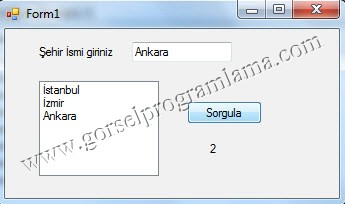
}

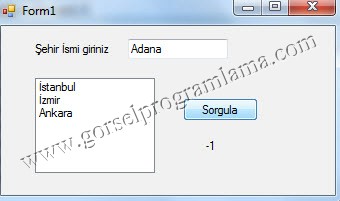
private void button1\_Click(object sender, EventArgs e)

{

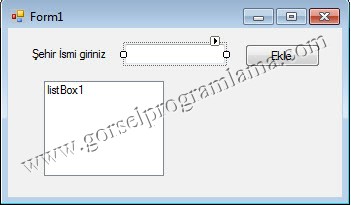
label2.Text = listBox1.Items.IndexOf(textBox1.Text).ToString();

}





Örnek:



private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

}

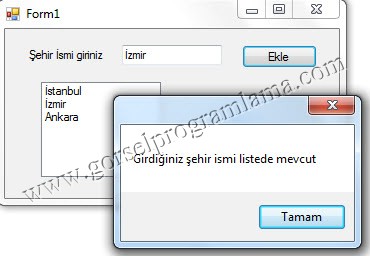
private void button1\_Click(object sender, EventArgs e)

{

if (listBox1.Items.IndexOf(textBox1.Text) != -1) MessageBox.Show("Girdiğiniz şehir ismi açılır listede mevcut");

else listBox1.Items.Add(textBox1.Text);

}//Daha fazla bilgi için : www.gorselprogramlama.com



Yukarıda gördüğümüz gibi textbox’a girilen şehir  listbox’ta  var ise eklemez.Bu örnek Items.Contains ve FindStringExact ilede yapılabilir.

Items.Contains Özelliği: listBox’ta eleman aramak için kullanılır.Eğer belirtilen eleman var ise true , yok ise false  değerini döndürür.

Örnek:

private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

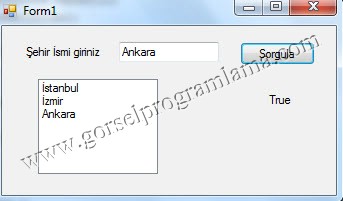
}

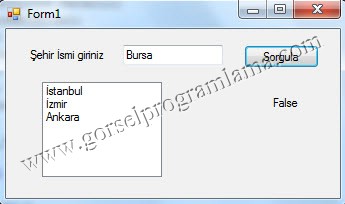
private void button1\_Click(object sender, EventArgs e)

{

label2.Text = listBox1.Items.Contains(textBox1.Text).ToString();

}





FindStringExact Özelliği:listBox’ta eleman aramak için kullanılır.Eğer belirtilen eleman var ise sıra numarasını , yok ise -1  değerini döndürür.

Örnek:

private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

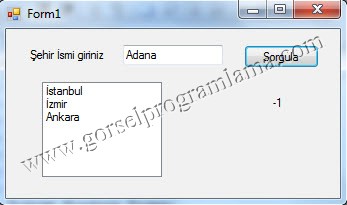
}

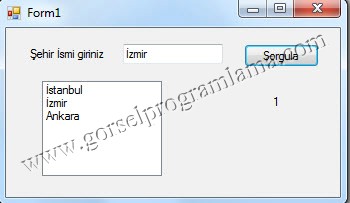
private void button1\_Click(object sender, EventArgs e)

{

label2.Text = listBox1.FindStringExact(textBox1.Text).ToString();

}}//Daha fazla bilgi için : www.gorselprogramlama.com





Not : Arama işlemini belirli bir elemandan başlatabilirsiniz.

Örnek : FindStrintExact(“gorsel”,3)  —>gorsel kelimesini 4. (elemanların sıfırdan başladığını unutmayın.) elemandan itibaren başlayarak ara.(var ise sıra numarasını yok ise -1 değerini döndürür.)

FindString Özelliği: listbox’ta eleman aramak için kullanılır.Eğer belirtilen metnin karakter sıraları listBox’taki harhangi bir  elemanın karakter sıraları ile eşleşiyorsa  ise sıra numarasını , tersi ise -1  değerini döndürür.Örnek ile daha iyi anlayacağınızı düşünüyorum.

Örnek:

private void Form1\_Load(object sender, EventArgs e)

{

listBox1.Items.Add("İstanbul");

listBox1.Items.Add("İzmir");

listBox1.Items.Add("Ankara");

}

private void button1\_Click(object sender, EventArgs e)

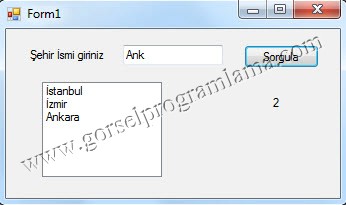
{

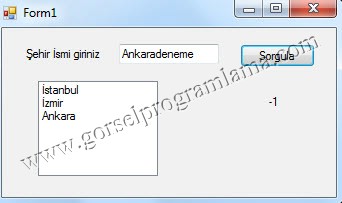
label2.Text = listBox1.FindString(textBox1.Text).ToString();

}

Aşağıdaki textbox  yazacağımız ifadelere göre çıkış değerleri:

İ=1   İz=1   İzm=1  İzmi=1  İzmir=1  İzmirx=-1  İzmirxx=-1  İstan=0  İstanb=0  İstanbul=0  İstanbulffff=-1  Anka=2  Ankara=2 Ankarayy=-1 Nevşehir=-1  gibi





Not : Arama işlemini belirli bir elemandan başlatabiliriz.

Örnek:  FindString(“program”,6)

**ListBox**

            Listeleri oluşturabileceğimiz bir kontroldür. Bir grup elmanı listelemek için kullanılır.

**Items**

            Liste içinde bulundurulacak elamanlar bu özellik ile belirlenir. Collection tipinde olan bu özellik çok sayıda alt özellik ve metot bulunur. Visual C#’ta diziler sıfırdan başladığı gibi liste içindeki indekslerde sıfırdan başlar.

**Items.Add(Elaman)**

            Listeye elman eklemek için kullanılan bir metottur. Metottan geriye dönen değer sadece liste içindeki sıra numarasıdır.

private void button1\_Click(object sender, EventArgs e)

        {

            listBox1.Items.Add("Ankara");

            listBox1.Items.Add("İstanbul");

            listBox1.Items.Add("İzmir");

        }

**Items.AddRange**

      Bir grup elamanı tek seferde listeye eklemek için bu metot kullanılır.

Örnek ListBox1’deki elamanları başka bir liste kutusuna aktaran programı yazınız.

private void button1\_Click(object sender, EventArgs e)

        {

            listBox2.Items.AddRange(listBox1.Items);

        }

**Örnek**

string[] x={“aaa”,”bbb”,”ccc”,”dddd”};

listBox2.Items.AddRange(x);

**Items.Insert(No,Eleman)**

            Items.Add  eklenen elemanı listenin sonuna ekler. Araya eleman eklemek için bu metot kullanılır.

private void button2\_Click(object sender, EventArgs e)

        {

            listBox1.Items.Add("Ankara");

            listBox1.Items.Add("İstanbul");

            listBox1.Items.Add("İzmir");

            listBox1.Items.Insert(1, "Bolu");

        }

**Items.RemoveAt(No)**

Numarası verilen elamanı siler.

**Items.Remove(Eleman)**

Verilen elemanı listeden siler. RemoveAt listedeki sıra numarasına göre siler Remove içeriğe göre siler.

listBox1.Items.Remove (“Ankara”);

**Örnek:**

private void button1\_Click(object sender, EventArgs e)

        {

            listBox1.Items.RemoveAt(0); //ilk elamanı siler

        }

Elemanı listeden çıkardığı için bütün elamanlar bir yukarı kayar. Bu nedenle döngü ile elamanları silerken dikkatli olmak gerekir.

Örnek: 5-10 arası elamanları listeden çıkarmak isteyelim.

for (int i = 5; i <= 10; i++)

            {

                listBox1.Items.RemoveAt(i);

            }

Bu kod doğru gibi görünüyor ama yanlıştır.

private void button2\_Click(object sender, EventArgs e)

        {

            for (int i = 0; i < 20; i++)

            {

                listBox1.Items.Add(i);

            }

        }

        private void button1\_Click(object sender, EventArgs e)

        {

            for (int i = 5; i < 10; i++)

            {

                listBox1.Items.RemoveAt(i);

            }

        }

Gördüğünüz gibi 5.6.7.8.9.10 yerine 5.7.9.11.13.15 silindi. Bu şekildeki silme işlemini;

for (int i = 5; i < 10; i++)

            {

                listBox1.Items.RemoveAt(5);

            }

Yada

for (int i =10; i >=5; i--)

            {

                listBox1.Items.RemoveAt(i);

            }

Listeden kullanıcının seçtiği elamanı silmek için;

private void button3\_Click(object sender, EventArgs e)

        {

            listBox1.Items.RemoveAt(listBox1.SelectedIndex);

        }

**Itmes.Clear**

      Listedeki bütün elemanları siler.

      Listbox1.Items.Clear();

**Items.Count**

      Listedaki eleman sayısını verir.

**Items[No]**

      Listede numarası verilen elemanın içeriğini öğrenmek veya değiştirmek için kullanılır.

**Items.IndexOf(Eleman)**

      Listede bir elemanı aramak için kullanılır. Aranan bulunursa sıra numarasını bulunmazsa -1 değerini verir

**Örnek:**

      İnt yer;

      Yer=listBox1.Items.IndexOf(“Ankara”);

      İf (yer<0)

            MessageBox.Show(“Ankara bulınamadı”);

      else

      MessageBox.Show(“Listede “+(yer+1).ToString()+”.eleman”);

**Örnek**

private void button1\_Click(object sender, EventArgs e)

        {

            listBox1.Items.Add(textBox1.Text);

            label2