1. Diziler

***Dizi:*** *Bellekte aynı türden sıralanmış verilerin oluşturduğu yapıya dizi denir.* *Aynı tipte veriyi bir arada tutar. Birden fazla boyutlu olabilir. Dizi değişkenleri, aynı tipte birçok veriyi bir arada tutmayı sağlar.* *Diziler tek boyutlu, çok boyutlu ve karmaşık (düzensiz) diziler olarak üç farklı şekilde tanımlayabilir ve kullanabiliriz.*

* 1. Tek Boyutlu Diziler

Diziler ile tek bir tanımlama yapılarak istenilen sayıda veri girişi sağlanabilir. Dizi elemanları bir indis numarasına sahiptir. İndis numaraları varsayılan olarak sıfırdan başlar. Dizinin kaç elemanlı olacağı dizinin tanımlandığı satırda veya daha sonra da belirtilebilir. Ayrıca dizi tanımlaması yapıldığında dizinin eleman sayısı, kullanılacak eleman sayısından az ise dizi yeniden boyutlandırılabilir. Dizi değişkenin eleman sayısını belirtmek için new metodu kullanılır.

Dizilimi ☞

değişken\_Tipi [ ] değişken;

değişken= new değişken\_Tipi[Eleman sayısı];

☞ Örnek

*5 elemanlı bir rakam dizisi tanımlanacak olursa;*

int [ ] rakam;

rakam= new int[5]; komut satırları yazılarak tanımlama yapılmış olur**.**

Ayrıca tanımlama aşağıdaki gibide yapılabilir.

int [ ] rakam= new int[5];

Komut satırları tek satırda yazılarak tanımlama yapılmış olur.

☞ Örnek

*String tipinde tek boyutlu dizi tanımlayın. Dizinin ismi “isimler” olsun. Bu dizi değişkene 4(dört) eleman atansın.*

# Uygulama kodu ☞

string [ ] isimler = new string[4];

# Uygulama kodu ☞

string [ ] isimler;

isimler = new string[4];

Dizilerin indisleri sıfırdan başlar. Örnekteki isimler dizisinin 3 tane String tipinden elemanı vardır. Dizilerin elemanlarına ulaşmak için, istenilen elemanın indisi verilmesi gerekir.

☞ Örnek

*String tipinde tek boyutlu dizi tanımlayın. Dizinin ismi “isimler” olsun. Bu dizi değişkene 3(üç) eleman atansın. 2. dizindeki elemanı Messagebox’ da görüntülesin.*

string [ ] isimler = new string[3];

isimler[0] = "Ali";

isimler[1] = "Kemal";

isimler[2] = "İsmail";

MessageBox.Show(isimler[2]);

Dizilere tek tek değer atanabildiği gibi, tanımlarken de başlangıç değerleri atanabilir.

string[ ] isimler = { "Mehmet", "Kemal", "İsmail", "Ayşe" };

☞ Örnek

*Butona tıklandığında, 5 elemanlı bir int dizi tanımlayarak dizi elemanlarını InputBox’tan okutan ve bu elemanları liste kutusunda listeleyen program kodunu yazınız.*

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

*{*

*byte i;*

*int[ ] dizi1 = new int[5];*

*{*

*for (i = 0; i < 5 ; i++)*

*dizi1[i] = Convert.ToInt32(Microsoft.VisualBasic.Interaction.InputBox("Yazılı Notunu Giriniz", "Bilgi Girişi", "", 10, 10));*

*{*

*for (i = 0; i < 5; i++)*

*listBox1.Items.Add(dizi1[i]);*

*}*

☞ Örnek

*4 elemanlı string bir dizi tanımlayın. Dizi elemanlarının ilk değerleri dizi satırda verilsin. Program çalıştırıldığında bu değerler ComboBox() ta listelensin. ComboBox’ tan seçim yapıldığı zaman seçilen metni (metin kutusunda) textBox1 içinde gösteren program kodu.*

string[ ] dersler =new string[4] {"Visual BASIC.NET","C#.NET","ASP:NET","3D GRAFİK ANİMASYON"};

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

{

comboBox1.Items.Add(dersler[0]);

comboBox1.Items.Add(dersler[1]);

comboBox1.Items.Add(dersler[2]);

comboBox1.Items.Add(dersler[3]);

}

private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)

{

if (comboBox1.SelectedIndex == 0)

textBox1.Text = dersler[0];

if (comboBox1.SelectedIndex == 1)

textBox1.Text = dersler[1];

if (comboBox1.SelectedIndex ==2)

textBox1.Text = dersler[2];

if (comboBox1.SelectedIndex == 3)

textBox1.Text = dersler[3];

}

NOT:*Aynı çalışmayı aşağıdaki dizi tanımlama satırındaki değişiklik ile de yazabiliriz. Burada Ayrıca, ilk değer ataması yapılacak olan dizilerde eleman sayısını belirtmeye gerek yoktur.*

string [ ] dersler= {"Visual BASIC.NET","C#.NET","ASP:NET","3D GRAFİK ANİMASYON"};

Bu tanımlamaya göre new metodunu kullanmayabilirsiniz. Çünkü dizi değişkenin eleman sayısı verilen ilk değerlerin sayısına göre belirlenir.

☞ Örnek

*4 elemanlı string bir dizi tanımlayın. Dizi elemanlarının ilk değerleri dizi satırda verilsin. Program çalıştırıldığında bu değerler ComboBox() ta listelensin. ComboBox’ tan seçim yapıldığı zaman seçilen metni (metin kutusunda) textBox1 içinde gösteren program kodu*

*string [ ] dersler= {"Visual BASIC.NET","C#.NET","ASP:NET","3D GRAFİK ANİMASYON"};*

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

*{*

*comboBox1.Items.Add(Mevsimler[0]);*

*comboBox1.Items.Add(Mevsimler[1]);*

*comboBox1.Items.Add(Mevsimler[2]);*

*comboBox1.Items.Add(Mevsimler[3]);*

*}*

*private void comboBox1\_SelectedIndexChanged(object sender, EventArgs e)*

*{*

*if (comboBox1.SelectedIndex == 0)*

*textBox1.Text = Mevsimler[0];*

*if (comboBox1.SelectedIndex == 1)*

*textBox1.Text = Mevsimler[1];*

*if (comboBox1.SelectedIndex ==2)*

*textBox1.Text = Mevsimler[2];*

*if (comboBox1.SelectedIndex == 3)*

*textBox1.Text = Mevsimler[3];*

*}*

☞ Örnek

*Bir boyutlu 7 elemanlı dizi tanımlayın. Buton1’e basıldığı zaman isimleri listbox1’de görüntülesin. Tek boyutlu dizi ile birlikte Foreach döngü komutunu kullanın.*

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

*{*

*string[] ad = new string[7];*

*ad[0] = "ismail";*

*ad[1] = "kemal";*

*ad[2] = "mehmet";*

*ad[3] = "ali";*

*ad[4] = "veli";*

*ad[5] = "can";*

*ad[6] = "canan";*

*foreach (string i in ad)*

*{*

*listBox1.Items.Add(i); } }*

☞ Örnek

*Bir boyutlu 7 elemanlı dizi tanımlayın. Buton1’e basıldığı zaman isimleri listbox1’de görüntülesin. Tek boyutlu dizi ile birlikte For döngü komutunu kullanın.*

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

*{*

*string[] ad = new string[7];*

*ad[0] = "ismail";*

*ad[1] = "kemal";*

*ad[2] = "mehmet";*

*ad[3] = "ali";*

*ad[4] = "veli";*

*ad[5] = "can";*

*ad[6] = "canan";*

*for(int i=0;i<ad.Length;i++)*

*{ listBox1.Items.Add(ad[i]); } }*

☞ Örnek

*Bir boyutlu 7 elemanlı dizi tanımlayın. Program çalıştığı zaman isimler listbox1’de, buton1’e basıldığı zaman toplam eleman sayısı MesajBoxda görüntülensin.*

*string[] ad = new string[7];*

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

*{*

*ad[0] = "ismail";*

*ad[1] = "kemal";*

*ad[2] = "mehmet";*

*ad[3] = "ali";*

*ad[4] = "veli";*

*ad[5] = "can";*

*ad[6] = "canan";*

*for (int i = 0; i < ad.Length; i++)*

*{*

*listBox1.Items.Add(ad[i]);*

*}*

*}*

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

*{ int adet = ad.Length;*

*MessageBox.Show("ad dizinde toplam" + adet.ToString() + " adet eleman vardır"); }*

☞ Örnek

*Bir boyutlu 6 elemanlı dizi tanımlayın. Program çalıştığı zaman 1 den 49 kadar 6 adet sayı üretip Listbox1 içinde görüntülesin.*

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

*{ Random rastgelesayi = new Random();*

*int[] dizi = new int[6];*

*for (int i = 0; i < dizi.Length; i++)*

*{*

*dizi[i] = rastgelesayi.Next(1,49);*

*listBox1.Items.Add(dizi[i]); } }*

* 1. Çok Boyutlu Diziler

Diziler tek boyutlu olduğu gibi, birkaç boyutlu diziler de tanımlanabilir. İki veya daha çok boyutlu bilgileri saklamak için kullanılan dizi türleridir. Tek boyutlu dizilerde olduğu gibi çok boyutlu dizilerde de ilk değer ataması yapılabilir. Çok boyutlu dizilerde işlem yapabilmek için genellikle iç içe döngü yapıları kullanılmaktadır.

**Dizilimi ☞**

Değişken\_Tipi [ **,** ] dizi\_değişken\_ismi;

dizi1 = new int[1.Boyut, 2.Boyut]; şeklinde tanımlama yapılır.

[,] şeklindeki bu gösterim dizinin iki boyutlu olduğunu belirtir. (,) virgül sayısı arttıkça dizinin boyutu da artar. Aşağıda 2 boyutlu “3x4” dizi tanımlamaları yapılmıştır. Bildirmeyi (declare) ve tanımlamayı (initialize) aynı anda yapabildiğimiz yöntem;

int[,] matris = new int[3, 4];

Bildirme ve tanımlamayı ayrı satırlarda yapabildiğimiz yöntem;

int[,] matris;

matris = new int[3, 4];

Çok boyutlu dizi içerisine değer atamalarını da yaptığımız yöntem;

int[,] matris = {

{ 0, 1, 2, 3} ,

{ 4, 5, 6, 7} ,

{ 8, 9, 10, 11}

};

int[,] array2D = new int[,] {

{ 1, 2 },

{ 3, 4 },

{ 5, 6 },

{ 7, 8 }

};

int[,] array = new int[4, 2] {

{ 1, 2 },

{ 3, 4 },

{ 5, 6 },

{ 7, 8 }

};

string[,] array2 = new string[3, 2] {

{ "one", "two" },

{ "three", "four" },

{ "five", "six" }

};

int[, ,] array3 = new int[,,] {

{

{ 1, 2, 3 },

{ 4, 5, 6 }

},

{

{ 7, 8, 9 },

{ 10, 11, 12 }

}

};

int[, ,] array3Da = new int[2, 2, 3] {

{

{ 1, 2, 3 },

{ 4, 5, 6 }

},

{

{ 7, 8, 9 },

{ 10, 11, 12 }

}

};

☞ Örnek

*2\*2’lik bir matrise ilk değer ataması gibi int[,] Matris ={ { 1, 0 }, { 0, 1 } }; yapılabilir. Bir dizinin boyutu öğrenilmek istenirse Rank () metodu kullanılır.*

*public partial class Form1 : Form*

*{*

*Random random = new Random();*

*string dizi\_ciktisi;*

*public Form1()*

*{*

*InitializeComponent();*

*}*

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

*{*

*int[,] iki\_boyutlu\_dizi = new int[3, 5];*

*for (int i = 0; i < 3; i++)*

*{*

*for (int j = 0; j < 5; j++)*

*{*

*iki\_boyutlu\_dizi[i, j] = random.Next(1, 100);*

*} }*

*for (int i = 0; i < 3; i++)*

*{*

*for (int j = 0; j < 5; j++)*

*{*

*dizi\_ciktisi += iki\_boyutlu\_dizi[i, j].ToString("0\t");*

*}*

*listBox1.Items.Add(dizi\_ciktisi);*

*dizi\_ciktisi = ""; } }}}*

☞ Örnek

“*Matrisin elemanlarını gir” düğmesine tıklandığında 3x2’lik bir matrise eleman girişi InputBox() ile yapılmaktadır. Dizi giriş işlemi bittikten sonra girilen elemanlar “Multiline” özelliğine sahip metin kutusuna girilerek satır-sütun halinde yazdıran program kodunu yazınız*.



*İlk boyutunda* ***5****, İkinci boyutunda* ***6 int*** *değeri olan* ***2 boyutlu*** *dizi*

int[,] matris = new int[5, 6];

*NOT: Burada dizinin ilk boyutunda 5 tane eleman vardır. İlk boyuttaki her eleman için İkinci boyutta 6 eleman bulunur. Dolayısıyla toplam 30 elemanlı bir dizidir. Bu dizide bir boyut daha olsaydı, o boyutun her elemanı için diğer boyutlardaki 30 eleman bulunacaktı. Çok boyutlu dizilerin eleman sayıları boyutlarındaki eleman sayılarını çarparak hesaplanabilir*.

int [,,,,] dizi = new int[boyut1,boyut2,boyut3,... ,boyutn];

Çok boyutlu dizilere başlangıç değerleri, dizinin boyutu dikkate alınarak verilmelidir. Boyutlardaki elemanlar küme parantezleri ile gruplanmalıdır.

☞ Örnek

*// İlk boyutunda 4, ikinci boyutunda 4 eleman olan*

*// 2 boyutlu dizi*

*int[ , ] matris = { { 1, 2, 3, 4 }, { 5, 6, 7, 8 } };*

Çok boyutlu dizilerin elemanlarına ulaşmak için, her boyut için indis göstermek gerekir. Matris [0, 0] = 1;

*public partial class Form1 : Form*

*{*

*Random random = new Random();*

*string dizi\_ciktisi;*

*public Form1()*

*{*

*InitializeComponent();*

*}*

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

*{*

*int[,] iki\_boyutlu\_dizi = new int[2, 10];*

*int [] tek\_boyutlu\_dizi = new int[10];*

*for (int i = 0; i < 2; i++)*

*{*

*for (int j = 0; j < 10; j++)*

*{*

*iki\_boyutlu\_dizi[i, j] = random.Next(1, 1000);*

*}*

*}*

*for (int i = 0; i < 2; i++)*

*{*

*for (int j = 0; j < 10; j++)*

*{*

*dizi\_ciktisi += iki\_boyutlu\_dizi[i, j].ToString("0\t"); }*

*listBox1.Items.Add(dizi\_ciktisi);*

*dizi\_ciktisi = "";}*

*for (int i = 0; i < 2; i++)*

*{*

*for (int j = 0; j < 10; j++)*

*{*

*tek\_boyutlu\_dizi[j] = iki\_boyutlu\_dizi[i, j]; }*

*Array.Sort(tek\_boyutlu\_dizi);*

*for (int j = 0; j < 10; j++)*

*{*

*iki\_boyutlu\_dizi[i, j] = tek\_boyutlu\_dizi[j]; }}*

*for (int i = 0; i < 2; i++)*

*{*

*for (int j = 0; j < 10; j++)*

*{*

*dizi\_ciktisi += iki\_boyutlu\_dizi[i, j].ToString("0\t");}*

*listBox2.Items.Add(dizi\_ciktisi);*

*dizi\_ciktisi = ""; }}} }*

☞ Örnek

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

*{*

*int[,] matris = new int[4, 4];*

*int i, k;*

*for (i = 0; i <= 3; i++)*

*{*

*for (k = 0; k <= 3; k++)*

*{*

*matris[i, k] = Convert.ToInt32(textBox1.Text);*

*textBox2.Text = textBox2.Text + " " + matris[i, k].ToString();*

*}*

*textBox2.Text = textBox2.Text + "\r\n";*

*textBox2.Text = textBox2.Text + "\r\n";*

*} } } }*

☞ Örnek

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

*{*

*int[, ,] matris = new int[4, 4, 4];*

*int m, i, k;*

*for (m = 0; m <= 3; m++)*

*{*

*for (i = 0; i <= 3; i++)*

*{*

*for (k = 0; k <= 3; k++)*

*{*

*matris[m, i, k] = Convert.ToInt32(textBox1.Text);*

*textBox2.Text = textBox2.Text + " " + matris[m, i, k].ToString();*

*}*

*textBox2.Text = textBox2.Text + "\r\n";*

*textBox2.Text = textBox2.Text + "\r\n";*

*textBox2.Text = textBox2.Text + "\r\n";*

*} } } }*

☞ Örnek

*Satır ve sütun sayıları klavyeden girilmek kaydıyla iki boyutlu diziyi oluşturan ve oluşturan dizi içinden rasgele sayıları textbox içinde görüntüleyen program kodunu yazınız.*

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

*{*

*int a = int.Parse(Microsoft.VisualBasic.Interaction.InputBox("Satır sayısını gir",""));*

*int b = int.Parse(Microsoft.VisualBasic.Interaction.InputBox("Sütun sayısını girr",""));*

*Random rnd = new Random();*

*int[,] dizi2 = new int[a, b];*

*for (int i = 0; i < a; i++)*

*{*

*for (int j = 0; j < b; j++)*

*{*

*dizi2[i, j] = rnd.Next(0, 50);*

*textBox1.Text = textBox1.Text + " " + dizi2[i, j].ToString();*

*}*

*textBox1.Text = textBox1.Text + "\r\n";*

*textBox1.Text = textBox1.Text + "\r\n";*

*} }*

* 1. Karışık (Düzensiz) Diziler

Bu dizilere Jagged diziler de denir. Karışık diziler sütun sayısı birbirinden farklı olan dizilerdir. Bir başka deyişle, farklı sütun sayılarına ait diziler dizisidir.

* Bir dizinin her bir elemanı ayrı bir dizi ise,
* Ve bunlar bir dizi içeriyorsa

İşte bu dizilere zaman düzensiz diziler denir. Bu dizi grubunda her bir satırdaki dizi farklı boyut olarak tanımlanabileceği için matris dizisi formunda olması gerekmez.

int[ ][ ] deneme\_dizi = new int[3][ ];

Bu tanıma göre 1. boyut 3 elemanlı olarak tanımlanmışken diğer boyutun kaç olacağı belli değildir. Tanımlanmayan diğer boyutun eleman sayısı ilk değer ataması sonucunda belirlenir. Buna göre; değer atamalarının aşağıdaki gibi yapıldığını düşünürsek,

deneme\_dizi[0] = new int[3];

deneme\_dizi[1] = new int[4];

deneme\_dizi[2] = new int[5];

burada deneme\_dizi[3][] tek boyut ve 3 eleman olarak tanımlanmış.İkinci dizinin boyutu belli değil. Ancak 1. Boyut 3 eleman atanmış olmasına rağmen dizinin her elemanını bir dizi içerdiğini düşünerekten indis değerlerine atama yaparak gösterecek olursak, bunu aşağıdaki gibi gösterebiliriz.

int[ ][ ] deneme\_dizi = new int[3][ ];

burada;

deneme\_dizi[0] = new int[3];

deneme\_dizi[1] = new int[4];

deneme\_dizi[2] = new int[5];

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| deneme\_dizi[3] |  | deneme\_dizi[0] | **1** | **2** | **3** |  |  |
| *NOT: Dizinin her bir elemanı ayrı bir dizi ise, bunlarda ayrı bir dizi içeriyor.* |  |  |  |  |  |  |  |
|  | deneme\_dizi[1] | **1** | **2** | **3** | **4** |  |
|  |  |  |  |  |  |  |
|  | deneme\_dizi[2] | **1** | **2** | **3** | **4** | **5** |

☞ Örnek

*1. boyutu 3 elemanlı diğer boyutunun eleman sayısı belli olmayan düzensiz dizi tanımlayın. 2. boyutunun elemanları form yüklendiğinde diziye girilmektedir. Her bir sütun elemanlarını ayrı ayrı liste kutusuna listeleyen ve bu dizinin boyut sayısını Button1’e basıldığında mesaj kutusunda gösteren program kodu aşağıdaki gibidir.*

*string[][] deneme = new String[3][];*

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

*{ deneme[0]=new string[] {"Kemal","Bora","ismail","ilhan" };//ilk dizin 4 elemanlı*

*deneme[1] = new string[] { "Kamil", "Behçet", "Hülya", "Gulçin", "Didem" };//ikinci dizin 5 elemanlı*

*deneme[2] = new string[] { "Engin", "Varol", "ismet" };//üçüncü dizin 3 elemanlı*

*for (int j = 0; j < 4; j++)*

*listBox1.Items.Add(deneme[0][j]);*

*for (int i = 0; i < 5; i++)*

*listBox2.Items.Add(deneme[1][i]);*

*for (int l = 0; l < 3; l++)*

*listBox3.Items.Add(deneme[2][l]);*

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

*{ MessageBox.Show(("Deneme Dizi="+deneme.Rank).ToString()+"boyutlu");//Rank Metodu=dizinin boyut sayısını veren özelliktir } } }*

☞ Örnek

*Bir boyutlu 3 elemanlı dizi tanımlayın.*

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

*{*

*int[][] tek\_boyut = new int[3][];*

*tek\_boyut[0] = new int[1];*

*tek\_boyut[1] = new int[3];*

*tek\_boyut[2] = new int[5];*

*for (int i = 0; i < tek\_boyut.Length; i++)*

*{*

*for (int j = 0; j < tek\_boyut[i].Length; j++)*

*{*

*tek\_boyut[i][j] = i + (1 + (i + j));*

*} }*

*foreach (int[] sayilar in tek\_boyut)*

*{*

*foreach (int sayi in sayilar)*

*{ listBox1.Items.Add(" " + sayi.ToString() + " ");*

*} } } } }*

1. Dizi Özellikleri Ve Metotları

Dizi işlemleri, programlarda çok sık kullanılan programlama öğelerindendir.

* *IndexOf()*
* *LastIndexOf()*
* *Sort()*
* *Reverse()*

Array sınıfı yardımıyla kullanılırlar. Diziler, .NET Framework içinde tanımlı Array sınıfı temsil eder. Tüm diziler Array sınıfında tanımlı özellikleri ve metotları kullanırlar.

* 1. IndexOf() :

Aramayı bastan başlatır ve aranan elemanın kaçıncı sırada olduğunu değer olarak dondurur. İlk parametrede verilen dizide, ikinci parametrede verilen değeri arar.

Dizilimi ☞

Array.IndexOf (1.Parametre , 2.Parametre)

☞ Örnek

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

*{*

*float[ ] notlar = {* ***100****,* ***100****,* ***100****,* ***100*** *};*

*float maxNot = 100;*

*MessageBox.Show(Array.IndexOf(notlar, maxNot).ToString());*

*//Sonuç =1 dır.*

*// Program, aramaya sol bastan yan1, 78 sayısından baslar.*

*//100'in bulunduğu yer numarası ise bastan index değeri olarak indis [1] dır. }*

☞ Örnek

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

*{*

*int[] dizi = {* ***1****, 2, 3, 4 };*

*MessageBox.Show(Array.IndexOf(dizi, 1).ToString());*

*//Sonuç =0 dır.*

*// Program, aramaya sol bastan yan, 1 sayısından baslar.*

*//1'in bulunduğu yer numarası ise bastan index değeri olarak indis [0] dır.*

}

* 1. LastIndexOf()

Aramayı sondan başlatarak aranan elemanın bastan kacıncı sırada olduğunu değer olarak dondurur.

Dizilimi ☞

Array.LastIndexOf ( 1.Parametre , 2.Parametre )

☞ Örnek

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

*{*

*float[] notlar = { 78, 100, 89, 99, 78, 100, 89, 99 };*

*float maxNot = 100;*

*MessageBox.Show(Array.LastIndexOf(notlar, maxNot).ToString());*

*// Arama sağdan sola yani sondan basa doğrudur.*

*// Ancak yer bildirimini bastan bastan itibaren baslamıstır*

*// bulunduğu yer numarası ise bastan index değeri olarak indis [5] dır.*

}

☞ Örnek

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

*{*

*float[ ] dizi = { 1, 2, 3, 1, 2, 3 }; ;*

*MessageBox.Show(Array.LastIndexOf(dizi, 1).ToString());*

}

*NOT: Aranan değer dizide bulunursa indisi, bulunamazsa( -1) döndürür.*

* 1. Array.Sort()

Dizinin elemanlarını küçükten büyüğe doğru sıralar.

Dizilimi ☞

Array.Sort (Dizi)

☞ Örnek

*Buton tıklandığı zaman, tek boyutlu dizi içinde tanımlanan 7 adet elemanı listbox1 içinde düzensiz, listbox2 içinde ise sıralanmış olarak görüntüleyen program kodunu yazınız.*

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

*{ int[] dizi = { 10,25,53,85,1,5,44 };*

*for (int i = 0; i < dizi.Length; i++)*

*{*

*listBox1.Items.Add(dizi[i].ToString());*

*// dizi düzensiz olarak listbox1 içinde görüntüleniyor.*

*}*

*Array.Sort(dizi); // dizi array.sort komutu ile sıralanıyor.*

*for (int i = 0; i < dizi.Length; i++)*

*{*

*listBox2.Items.Add(dizi[i].ToString());*

*// dizi düzenli olarak listbox2 içinde görüntüleniyor. } }*

☞ Örnek

*Buton tıklandığı zaman, tek boyutlu dizi içinde tanımlanan5 adet elemanı listbox1 içinde düzensiz, listbox2 içinde ise sıralanmış olarak görüntüleyen program kodunu yazınız.*

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

*{*

*string[] dizi = { "mine","can","mehmet","kemal","yağız" };*

*for (int i = 0; i < dizi.Length; i++)*

*{*

*listBox1.Items.Add(dizi[i].ToString());*

*// dizi düzensiz olarak listbox1 içinde görüntüleniyor.*

*}*

*Array.Sort(dizi);*

*// dizi array.sort komutu ile sıralanıyor.*

*for (int i = 0; i < dizi.Length; i++)*

*{*

*listBox2.Items.Add(dizi[i].ToString());*

*// dizi düzenli olarak listbox2 içinde görüntüleniyor. }} } }*

* 1. Length()

Dizinin bütün boyutlarındaki toplam eleman sayısını veren özelliktir.

Dizilimi ☞

Nesne.Length veya Değişken.Length

☞ Örnek

*private void button4\_Click(object sender, EventArgs e)*

*{*

*string[] dizi = { "A", "B", "C", "D", "E" };*

*listBox4.Items.Add("ELEMAN SAYISI=" + " " + dizi.Length.ToString());*

*}*

* 1. GetLength()

İndisi verilen boyutun kaç elemanlı olduğunu gösterir. İndis sıfırdan başlar.

Dizilimi ☞

değişken\_ismi.GetLength(İndis\_no)

☞ Örnek

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

*{*

*int[, , , ,] dizi = new int[5, 10, 6, 15, 7];*

*MessageBox.Show(dizi.GetLength(3).ToString());*

*// Sonuç = GetLength(3)ile dizi içinde 3. indise karşılık gelen değer 15 sayısıdır.*

*}*

* 1. Rank()

Dizinin boyut sayısını veren özelliktir.

**Dizilimi ☞**

değişken\_ismi.Rank( )

☞ Örnek

*private void button8\_Click(object sender, EventArgs e)*

*{*

*int[, , , ,] dizi = new int[10, 40, 50, 80, 90];*

*listBox8.Items.Add("BOYUTU=" + " " + dizi.Rank.ToString());*

*}*

* 1. Reverse()

Parametre olarak verilen dizinin eleman sırasını tersine çevirir. Dizinin tüm elemanlarının veya belirli indis aralığındaki elemanlarının sırası tersine çevrilebilir.

Dizilimi ☞

Array.Reverse(dizi\_değişken\_ismi)

☞ Örnek

*string[ ] harfler = { "A", "B", "C" };*

*Array.Reverse(harfler);*

*MessageBox.Show(harfler[2]);*

*// Sonuç = A*

☞ Örnek

*string[ ] harfler = { "A", "B", "C" };*

*Array.Reverse(harfler, 0, 1);*

*MessageBox.Show(harfler[2]);*

*// Sonuç = C*

* 1. Clear

Parametre olarak verilen dizinin, belirtilen indis aralığındaki tüm değerleri temizler. Temizleme işleminde atanan değer, dizi elemanlarının tiplerine göre değişir. int tipinde tanımlı bir dizinin elemanları temizlenirse 0 değerini alacaktır. Buna karşın String tipindeki elemanlar “” (boş yazı) değerini alır.

**Dizilimi ☞**

Array.Clear(dizi\_değişken\_ismi,silinecek\_başlangıç\_indexno,silinecek\_eleman\_sayısı)

☞ Örnek

*int[] dizi = { 12, 13, 14, 15 };*

*// 1. indisten başlayarak, 3 elemanı temizle*

*Array.Clear(dizi, 1, 3);*

*MessageBox.Show(dizi[2].ToString());*

*//Sonuç = 0*

*// Dizinin tüm elemanlarını temizler*

*Array.Clear(dizi, 0, dizi.Length);*

* 1. ArrayList

Klasik dizilerle çalışırken karşılaşabileceğimiz temel sorunlar şunlardır.

* Dizilerin sınırları sabittir.
* Dizilerin elemanlarını tipi aynı türden olmalıdır.
* Fazladan tanımlanan dizi elemanlarının kullanılmaması nedeniyle bellek alanları gereksiz yere işgal edilmektedir.

ArrayList Metotları ve Özellikleri

|  |  |
| --- | --- |
| Add() | Nesneyi ArrayList'in sonuna ekler. |
| BinarySearch() | Sıralanmış bir ArrayList içinde bir nesne aramak için kullanılır. |
| Clear() | ArrayList'in tüm elemanlarını siler. |
| Contains() | Nesnenin ArrayList'in elemanı olup olmadığını kontrol eder. |
| Insert() | Dizinin sonuna debilde istediğimiz bir yerine indeksini belirterek eklememizi sağlar. |
| Remove() | Herhangi bir elemanı diziden siler. |
| Reverse() | Diziyi ters çevirir. |
| Sort() | Diziyi sıralar. |

☞ Örnek

using System;

using System.Collections.Generic;

using System.ComponentModel;

using System.Data;

using System.Drawing;

using System.Linq;

using System.Text;

using System.Windows.Forms;

namespace WindowsFormsApplication6

{

public partial class Form1 : Form

{

public Form1()

{

InitializeComponent();

}

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

{

float[] notlar = { 8, 20, 18, 19 };

float maxNot = 20;

listBox1.Items.Add("İNDİS DEĞERİ = " + Array.IndexOf(notlar, maxNot).ToString());

//Sonuç =1 dır.

// Program, aramaya sol bastan yan1, 20 sayısından baslar.

//20'in bulunduğu yer numarası ise bastan index değeri olarak indis [1] dır.

}

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

{

float[] dizi = { 1, 2, 3, 1, 2, 3 }; ;

listBox2.Items.Add("İNDİS DEĞERİ = " + Array.LastIndexOf(dizi, 1).ToString());

// Arama sağdan sola yani sondan basa doğrudur.

// Ancak yer bildirimini bastan bastan itibaren baslamıstır

// yeri ise 0 dan itibaren 3. sıradadır.

}

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

{ string[] dizi = { "G", "C", "M", "K", "Y" };

for (int i = 0; i < dizi.Length; i++)

{ listBox3.Items.Add(dizi[i].ToString()); // dizi düzensiz olarak listbox1 içinde görüntüleniyor.

}

Array.Sort(dizi); // dizi array.sort komutu ile sıralanıyor.

for (int i = 0; i < dizi.Length; i++)

{ listBox11.Items.Add(dizi[i].ToString()); // dizi düzenli olarak listbox2 içinde görüntüleniyor.

}

}

private void button4\_Click(object sender, EventArgs e)

{

string[] dizi = { "A", "B", "C", "D", "E" };

listBox4.Items.Add("ELEMAN SAYISI=" + " " + dizi.Length.ToString());

}

private void button5\_Click(object sender, EventArgs e)

{

int[, , , ,] dizi = new int[10, 40, 50, 80, 90];

listBox5.Items.Add("B. E. SAYISI=" + " " + dizi.GetLength(4).ToString());

}

private void button8\_Click(object sender, EventArgs e)

{

int[, , , ,] dizi = new int[10, 40, 50, 80, 90];

listBox8.Items.Add("BOYUTU=" + " " + dizi.Rank.ToString());

}

private void button9\_Click(object sender, EventArgs e)

{

string[] harfler = { "G", "C", "M", "K", "Y" };

Array.Reverse(harfler);

listBox9.Items.Add(harfler[4]);

}