

String (Dizgi) İşlemleri

BİL131 - Bilişim Teknolojileri ve Programlama

Hakan Ezgi Kızılöz

String (Dizgi) Sınıfı

- **String** bir karakter dizisidir.
- **String** Java'nın temel veri türlerinden değildir ve sınıf olarak tasarlanmıştır.
- Aşağıda örnekler verilmiştir:

```
String iletisi = "TOBB Ekonomi ve Teknoloji Universitesi";  
System.out.print(iletisi);
```

veya

```
System.out.print("TOBB Ekonomi ve Teknoloji Universitesi");
```

Çıktı:

```
TOBB Ekonomi ve Teknoloji Universitesi
```

Dizgilerin Eklenmesi

- İki dizginin art arda eklenmesi + ile yapılır.
- Aşağıda örnek verilmiştir:

```
String birinciParca = "TOBB";  
String ikinciParca = "Ekonomi ve Teknoloji";  
String tobbEtu = birinciParca + ikinciParca +  
"Universitesi";  
System.out.print(tobbEtu);
```

Çıktı:

TOBBEkonomi ve TeknolojiUniversitesi

Dizgilerin Boyutu

- Dizgilerin uzunluğunu bulmak için dizgi sınıfının `length()` metodu kullanılır.
- Aşağıda örnek verilmiştir:

```
int uzunluk = "TOBB ETU".length();  
System.out.print(uzunluk);
```

veya

```
String tobbEtu = "TOBB ETU";  
int uzunluk = tobbEtu.length();  
System.out.println(uzunluk);
```

Çıktı:

8

Karakterlerin Dizgideki Konumu

- * Dizgiler, karakterlerin art arda eklenmesiyle oluşmuş cümlelerdir.
- * Cümleler içerisinde karakterlere teker teker erişilebilmektedir.
- * Karakterlerin yerini bulabilmek için saymaya 1'den değil 0'dan başlamak gerekmektedir.

0	1	2	3	4	5	6	7	8	9	10	11	12	13
m	e	r	h	a	b	a		d	u	n	y	a	.

Dizgi Metotları

Metot	İşlevi
<code>.toLowerCase()</code>	Dizgi küçük harfe çevrilir. String ileti = "MERHABA"; String kucukHarfli = ileti.toLowerCase(); // merhaba olur
<code>.toUpperCase()</code>	Dizgi büyük harfe çevrilir. String ileti = "merhaba"; String buyukHarfli = ileti.toUpperCase(); // MERHABA olur

Dizgi Metotları

Metot	İşlevi
<code>.substring(Başlangıç)</code>	Dizgide Başlangıçtan itibaren en sona kadar kopyalama yapar. String ileti = "merhaba"; String parca = ileti.substring(4); // aba olur
<code>.substring(Başlangıç, Son)</code>	Dizgide Başlangıçtan Sona kadar kopyalama yapar. String ileti = "merhaba"; String parca = ileti.substring(1, 4); // erh olur

Dizgi Metotları

Metot	İşlevi
<code>.indexOf(Bir_Dizgi)</code>	Dizgi içerisinde Bir_Dizgi'yi arar; varsa başlangıç sırasını yoksa -1 değerini döndürür. <pre>String ileti = "mudur mudur mudur?"; int mudurNerede = ileti.indexOf("mudur"); // mudurNerede değişkeninin değeri 0 olur</pre>

Dizgi Metotları

Metot	İşlevi
<code>.indexOf(Bir_Dizgi, Başlangıç)</code>	Dizgi içerisinde Bir_Dizgi'yi Başlangıç'tan itibaren arar; varsa başlangıç sırasını yoksa -1 değerini döndürür. <pre>String ileti = "mudur mudur mudur?"; int mudurNerede = ileti.indexOf("mudur", 4); // mudurNerede değişkeninin değeri 6 olur</pre>

Dizgi Metotları

Metot	İşlevi
<code>.lastIndexOf(Bir_Dizgi)</code>	Dizgi içerisinde Bir_Dizgi'yi sondan başa doğru arar; varsa başlangıç sırasını yoksa -1 değerini döndürür. <pre>String ileti = "mudur mudur mudur?"; int mudurNerede = ileti.lastIndexOf("mudur"); // mudurNerede değişkeninin değeri 12 olur</pre>

Dizgi Metotları

Metot	İşlevi
<code>.trim()</code>	Dizginin başındaki ve sonundaki boşlukları siler. <pre>String ileti = " merhaba "; int bosluklarAtilmis = ileti.trim(); // bosluklarAtilmis değişkeni değeri "merhaba" olur</pre>

Dizgi Metotları

Metot	İşlevi
<code>.charAt (Konum)</code>	Dizgide yer alan Konum numaralı karakteri geri verir. <pre>String ileti = "merhaba"; char konumdakiKarakter = ileti.charAt(3); // konumdakiKarakter değişkeninin değeri 'h' olur</pre>

Dizgi Metotları

Metot	İşlevi
<code>.compareTo(Bir_Dizgi)</code>	Dizgiyi Bir_Dizgi ile alfabetik olarak karşılaştırır. Eğer Dizgi sözlükte Bir_Dizgi'den önce geliyorsa negatif bir değer, sonra geliyorsa pozitif bir değer döndürür. Dizgi ile Bir_Dizgi birbirine eşitse 0 değerini döndürmektedir. <pre>String iletı = "merhaba"; int esıtlık = iletı.compareTo("merhaba"); // esıtlık değışkenının değeri 0 olur</pre>

Dizgi Metotları

```
import java.util.Scanner;
public class StringDenemesi
{
    public static void main (String [] args)
    {
        String isim, baslik, altIsim;
        Scanner klavye = new Scanner(System.in);
        isim = klavye.next();
        System.out.print ("Isimdeki harf sayisi = ");
        System.out.println(isim.length());
        System.out.println(isim.toLowerCase());
        baslik = "    JAVA    ";
        System.out.println(baslik);
        System.out.println(baslik.trim());
        System.out.println(isim.charAt(0));
        altIsim = isim.substring(3);
        System.out.println(altIsim);
        altIsim = isim.substring(0,4);
        System.out.println(altIsim);
        int nerede = isim.indexOf("TURK");
        System.out.println(nerede);
    }
}
```

Çıktı:

```
TURKIYE // Klavyeden TURKIYE okundu
Isimdeki simge sayısı = 7
turkiye
    JAVA
JAVA
T
KIYE
TURK
0
```

Dizgi Eşitliği

- İki dizginin eşit olup olmadığı kontrolü `equals()` ve `equalsIgnoreCase()` metotları ile yapılmaktadır.
- `equals()` metodu, iki dizginin birebir aynı olup olmadıklarını kontrol etmektedir.
- Karşılaştırma yaparken büyük/küçük harf ayrımı yapılması istenmiyorsa `equalsIgnoreCase()` metodu kullanılmalıdır.
- Çıkan sonuç boolean türündedir ve iki dizgi aynı ise `true`, farklı ise `false` değerini taşır.

Dizgi Eşitliği

```
public class EqualsDenemesi
{
    public static void main (String [] args)
    {
        String birinci = "TOBB";
        String ikinci = "ETU";
        String ucuncu = "etu";

        System.out.println(birinci.equals(birinci));
        System.out.println(birinci.equals(ikinci));
        System.out.println(ikinci.equalsIgnoreCase(ucuncu));
        System.out.println(ikinci.equals(ucuncu.toUpperCase()));
    }
}
```

Çıktı:

```
true
false
true
true
```

Örnek Program

```
import java.util.*;
public class StringIslemleri
{
    public static void main (String [] args)
    {
        String isim, baslik, altIsim;
        int nerede;
        Scanner klavye = new Scanner(System.in);
        System.out.println("Bir cumle giriniz...");
        isim = klavye.nextLine();
        System.out.print ("Bu cumlede bosluklar dahil ");
        System.out.println(isim.length() + " karakter var.");
        System.out.print ("Cumlenin buyuk harfle yazilmis ");
        System.out.println("hali...");
        System.out.println(isim.toUpperCase());
        baslik = "    TOBB ETU    ";
        System.out.println("Baslik: " + baslik);
        System.out.print("Basligin trimden sonraki hali: ");
        System.out.println(baslik.trim());
        altIsim = isim.substring(3);
        System.out.println("Alt baslik: " + altIsim);
        altIsim = isim.substring(1,4);
        System.out.println("Alt baslik: " + altIsim);
        nerede = isim.lastIndexOf("T");
        System.out.print ("T'nin bulundugu son konum: ");
        System.out.println(nerede);
    }
}
```

Çıktı:

```
Bir cumle giriniz...
TOBB etu    // klavyeden girilmiştir
Bu cümlede bosluklar dahil 8 karakter var.
Cumlenin buyuk harfle yazilmis hali...
TOBB ETU
Baslik:      TOBB ETU
Basligin trimden sonraki hali: TOBB ETU
Alt baslik: B etu
Alt baslik: OBB
T'nin bulundugu son konum: 0
```

Özel Karakterler

Metot	İşlevi
<code>\"</code>	<code>"</code> işaretinin String içinde kullanımı
<code>\'</code>	<code>'</code> işaretinin String içinde kullanımı
<code>\\</code>	<code>\</code> işaretinin String içinde kullanımı
<code>\n</code>	String içinde yeni satıra geçilir
<code>\r</code>	String içinde sayır başına geçilerek yazmaya devam eder
<code>\t</code>	String içinde sekme (tab) yapar

Özel Karakterler

- Java'da özel karakterlerin örnek kullanımları aşağıda verilmiştir:

```
char tekTirnak = '\\';  
System.out.println("\\r\\t\\n"); // kullanım örneği  
System.out.println("\\n\\n\\n"); // 4 satır alta geçilir  
System.out.println("TOBB\\nEkonomi ve Teknoloji Üniversitesi");
```

Çıktı:

```
TOBB  
Ekonomi ve Teknoloji Üniversitesi
```