

Nesne Yönelimli Programlama

Hazırlayan: M.Ali Akcayol
Gazi Üniversitesi
Bilgisayar Mühendisliği Bölümü

Not: Bu dersin sunumları, "Java Programlama Dili ve Yazılım Tasarımı, Altuğ B. Altıntaş, Papatya Yayıncılık, 2016" kitabı kullanılarak hazırlanmıştır.

Konular

- **Diziler**
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Diziler

- **Diziler nesnedir ve içerisinde belirli sayıda eleman bulunur.**
- **Dizi elemanlarına** tam sayı ile ifade edilen bir **indis** kullanılarak erişilir.
- Bir dizide **n tane eleman** varsa **dizinin uzunluğu n'dir.**
- Dizilerdeki **ilk elemanın** indisi/konumu **0 ile başlar, son elemanı** ise **n-1** sırasına sahiptir.
- **Dizinin elemanları aynı türde olmak zorundadır.**
- Eğer dizi içerisindeki elemanların türü double ise, bu dizinin türü için double denilir.
- double türünde olan diziye String tipinde bir nesne atanmak istenirse hata ile karşılaşılır.
- **Diziler temel veri tipinde** veya herhangi **bir sınıf tipinde olabilir.**

3

Diziler

- **Dizi nesnelerini oluşturmak için new anahtar sözcüğü kullanılması gereklidir.**

```
double [] dd; // double tipindeki dizi
double dd []; // double tipindeki dizi
float [] fd; // float tipindeki dizi
Object [] ao ; // Object tipindeki dizi
```

- Yukarıda tanımlama yapıldı ancak hafızada yer ayrılmadı.

```
double [] d = new double[20]; // 20 elemanlı double tipindeki dizi
double dd []= new double[20]; // 20 elemanlı double tipindeki dizi
float [] fd = new float [14]; // 14 elemanlı float tipindeki dizi
Object []ao = new Object[17]; // 17 elemanlı Object tipindeki dizi
String [] s = new String[25]; // 25 elemanlı String tipindeki dizi
```

4

Diziler

Örnek

- Ölçülmüş 7 günlük hava sıcaklıklarını kullanarak istatistiksel veriler isteniyor.
- Günlere ait sıcaklık değerleri için ayrı ayrı **7 tane int türünde değişken ile tanımlanabilir.**

```
int gun1, gun2, gun3, gun4, gun5, gun6, gun7;
```

- Veri sayısı arttıkça değişken sayısı artacağından kullanışlı bir yaklaşım değildir. Bunun yerine **dizi ile bir değişken tanımlanarak çok sayıda değer saklanabilir.**

```
int [] sicaklik = new int [7];
```

```
int sicaklik [] = new int [7];
```

5

Diziler

- **Dizinin her elemanına ayrı ayrı değer atanabilir.**

```
sicaklik[0] = 7;  
sicaklik[1] = 4;  
sicaklik[2] = -1;  
sicaklik[3] = -3;  
sicaklik[4] = 0;  
sicaklik[5] = 2;  
sicaklik[6] = 3;
```

- Diziye girilen değerlerin toplamı istenirse aşağıdaki gibi **bir döngüyle atama ve hesaplama yapılabilir.**

```
for (int i = 0; i < 7; i++) {  
    sicaklik[i] = klavye.nextInt();  
    toplam = toplam + sicaklik[i];  
}
```

6

Diziler

```
/*
 * Bu program girilen sıcaklıkların ortalamasını
 * almakta, en düşük ve en yüksek sıcaklıkları yazmaktadır.
 */

public class DiziOrnekler {

    public static void main(String[] args) {
        Scanner klavye = new Scanner(System.in);
        int sicaklik[] = new int[30];
        int endusuksicaklik, enyukseksicaklik;
        int gunendusuk, gunenyukse, toplam = 0;
        double ortalama;

        for (int i = 0; i < 30; i++) {
            System.out.print((i + 1) + ". gün sıcaklık = ");
            sicaklik[i] = klavye.nextInt();
            toplam += sicaklik[i];
        }

        ortalama = toplam / 30.0;
    }
}
```

7

Diziler

```
System.out.println("Ay sıcaklık ortalamasi = " + ortalama);
endusuksicaklik = sicaklik[0];
gunendusuk = 0;
enyukseksicaklik = sicaklik[0];
gunenyukse = 0;

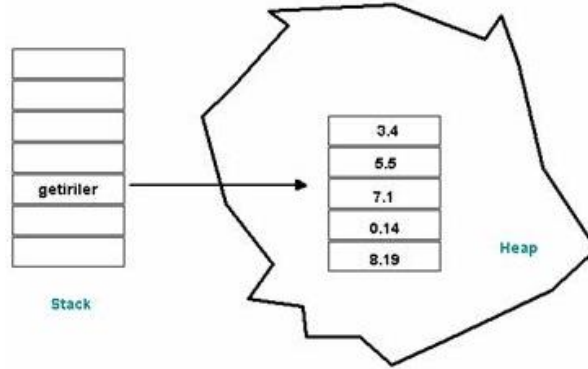
for (int i = 1; i < 30; i++) {
    if (endusuksicaklik > sicaklik[i]) {
        endusuksicaklik = sicaklik[i];
        gunendusuk = i;
    } else if (enyukseksicaklik < sicaklik[i]) {
        enyukseksicaklik = sicaklik[i];
        gunenyukse = i;
    }
}

System.out.print("En düşük sıcaklık = ");
System.out.println(endusuksicaklik + " Gun = " + (gunendusuk + 1));
System.out.print("En yüksek sıcaklık = ");
System.out.println(enyukseksicaklik + " Gun = " + (gunenyukse + 1));
}

1. gün sıcaklık = 12
2. gün sıcaklık = 13
3. gün sıcaklık = 14
4. gün sıcaklık = 11
5. gün sıcaklık = 10
6. gün sıcaklık = 15
7. gün sıcaklık = 16
8. gün sıcaklık = 12
9. gün sıcaklık = 11
10. gün sıcaklık = 14
11. gün sıcaklık = 13
12. gün sıcaklık = 15
13. gün sıcaklık = 12
14. gün sıcaklık = 10
15. gün sıcaklık = 16
16. gün sıcaklık = 17
17. gün sıcaklık = 18
18. gün sıcaklık = 10
19. gün sıcaklık = 11
20. gün sıcaklık = 12
21. gün sıcaklık = 13
22. gün sıcaklık = 15
23. gün sıcaklık = 14
24. gün sıcaklık = 16
25. gün sıcaklık = 12
26. gün sıcaklık = 13
27. gün sıcaklık = 12
28. gün sıcaklık = 10
29. gün sıcaklık = 12
30. gün sıcaklık = 13
Ay sıcaklık ortalamasi = 13.06666
En düşük sıcaklık = 10 Gun = 5
En yüksek sıcaklık = 18 Gun = 17
```

Diziler

- Diziler tanımlandığında ilk değerleri atanabilir.
`double[] getiriler = {3.4, 5.5, 7.1, 0.14, 8.19};`
- Temel double türündeki getiriler referansı yığın (**stack**) alanında yer almaktadır.
- Referansın bağlı olduğu dizi nesnesi ise **heap alanında yer almaktadır.**



9

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Dizi İşlemleri

Dizi boyutu

- Dizi boyutu tanımlandıktan sonra değiştirilemez.

```
int liste[] = new int[5];  
liste = new int[15]; // yeni bir dizi nesnesi bağlandı
```

- Yukarıda liste isimli referansa yeni bir dizi bağlandı.
- Önceki 5 elemanlı dizi garbage collector tarafından temizlenecektir.

11

Dizi İşlemleri

Dizi elemanlarına erişim

- Java dilinde eğer tanımlanan **dizinin sınırları aşılsa**, **çalışma zamanı hatası (runtime error)** ile karşılaşılır.
- Örneğin 20 elemanlı bir double dizisi tanımlanmış olsun.
- Eğer bu dizinin 21. elemanına ulaşmak istenirse (böyle bir indisli eleman yok) çalışma zamanı hatası ortaya çıkar (**ArrayIndexOutOfBoundsException**).
- Böylece dizi için ayrılmış bellek alanından dışarı çıkılıp başka verilere müdahale edilmesi engellenmiş olunur.

12

Dizi İşlemleri

- `length` değişkeni **dizinin büyüklüğünü (eleman sayısını)** gösterir.

```
double [] dersnotu = new double[50];  
System.out.println("Dizinin uzunluğu = " + dersnotu.length);
```

- **Dizinin uzunluğu atama ile değiştirilemez.**

```
dersnotu.length = 100; // atama yapılamaz
```

13

Dizi İşlemleri

```
package DiziOrnekler;  
  
public class DiziOrnekler {  
  
    public static void main(String args[]) {  
        double[] d = { 2.1, 3.4, 4.6, 1.1, 0.11 };  
        String[] s = { "defter", "kalem", "sarman", "tekir", "boncuk" };  
  
        // double tipindeki dizimizi ekrana yazdırıyoruz  
        for (int i = 0; i < d.length; i++) {  
            System.out.println("d[" + i + "] = " + d[i]);  
            // System.out.println("d["+78+"] = " + d[78] ); // Hata !  
        }  
  
        System.out.println("-----");  
  
        // String tipindeki dizimizi ekrana yazdırıyoruz  
        for (int x = 0; x < s.length; x++) {  
            System.out.println("s[" + x + "] = " + s[x]);  
            // System.out.println("s["+78+"] = " + s[78] ); // Hata !  
        }  
    }  
}
```

```
d[0] = 2.1  
d[1] = 3.4  
d[2] = 4.6  
d[3] = 1.1  
d[4] = 0.11  
-----  
s[0] = defter  
s[1] = kalem  
s[2] = sarman  
s[3] = tekir  
s[4] = boncuk
```

14

Dizi İşlemleri

```
public class DiziOrnekler {  
    double[] d;  
    String[] s;  
    public DiziOrnekler() { // double tipindeki diziyi eleman atanıyor  
        d = new double[5];  
        d[0] = 2.1;  
        d[1] = 3.4;  
        d[2] = 4.6;  
        d[3] = 1.1;  
        d[4] = 0.11;  
        // d[5] = 0.56 ; // Hata !  
        // String tipindeki diziyi eleman atanıyor  
        s = new String[5];  
        s[0] = new String("defter");  
        s[1] = new String("kalem");  
        s[2] = new String("sarman");  
        s[3] = new String("tekir");  
        s[4] = new String("boncuk");  
        // s[5] = new String("duman"); // Hata !  
    }  
    public void ekranaBas() { // double tipindeki diziyi ekrana yazdırıyoruz  
        for (int i = 0; i < d.length; i++) {  
            System.out.println("d[" + i + "] = " + d[i]);  
        }  
        System.out.println("-----"); // String dizi ekrana yazdırılıyor  
        for (int x = 0; x < s.length; x++) {  
            System.out.println("s[" + x + "] = " + s[x]);  
        }  
    }  
    public static void main(String args[]) {  
        DiziOrnekler deg = new DiziOrnekler();  
        deg.ekranaBas();  
    }  
}
```

d[0] = 2.1
d[1] = 3.4
d[2] = 4.6
d[3] = 1.1
d[4] = 0.11

s[0] = defter
s[1] = kalem
s[2] = sarman
s[3] = tekir
s[4] = boncuk

15

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

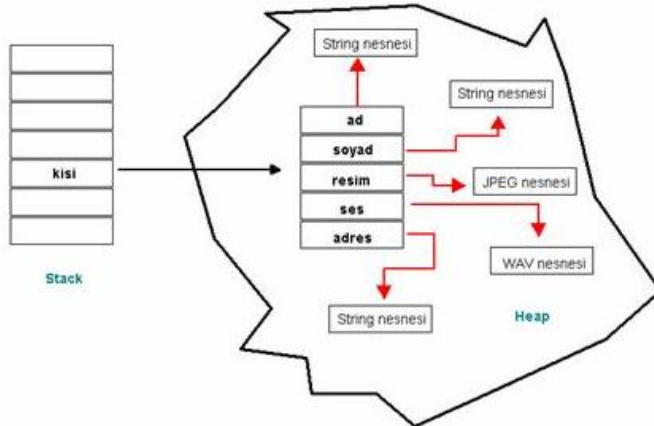
Nesne Dizileri

- **Nesnelerden oluşan bir dizi, temel türlerden oluşan bir diziden çok daha farklıdır.**
- **Nesnelerden meydana gelmiş bir dizi** oluşturulduğu zaman, bu dizinin elemanlarının **içerisinde, ilgili nesnelere ait referanslar tutulur.**
- Örneğin aşağıdaki gibi bir dizi yapısı bulunsun:
 - **Ad**, String sınıfı tipinde
 - **Soyad**, String sınıfı tipinde
 - **Resim**, JPEG sınıfı tipinde (böyle bir sınıf olduğunu varsayalım)
 - **Ses**, WAV sınıfı tipinde (böyle bir sınıf olduğunu varsayalım)
 - **Adres**, String sınıfı tipinde

17

Nesne Dizileri

- Ad, String sınıfı tipinde
- Soyad, String sınıfı tipinde
- Resim, JPEG sınıfı tipinde (böyle bir sınıf olduğunu varsayalım)
- Ses, WAV sınıfı tipinde (böyle bir sınıf olduğunu varsayalım)
- Adres, String sınıfı tipinde



18

Nesne Dizileri

```
class KahveFincani {  
    public KahveFincani(int i) {  
        System.out.println((i+1) + ". kahvefincani nesnesi başlatıldı...");  
    }  
}  
  
public class NesneOrnekler {  
    public static void main(String[] args) {  
        KahveFincani [] kahvefincani = new KahveFincani[5];  
        for (int i = 0; i < 5; i++)  
            kahvefincani[i] = new KahveFincani(i);  
    }  
}
```

```
1. kahvefincani nesnesi başlatıldı...  
2. kahvefincani nesnesi başlatıldı...  
3. kahvefincani nesnesi başlatıldı...  
4. kahvefincani nesnesi başlatıldı...  
5. kahvefincani nesnesi başlatıldı...
```

19

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- **Fonksiyonlara Dizi Gönderilmesi**
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Fonksiyonlara Dizi Gönderilmesi

- Bir fonksiyona dizi elemanlarının tamamı doğrudan dizi ismiyle gönderilir.
- Fonksiyon başlık tanımındaki dizi değişkeni ile gönderilen dizi elemanlarının türü aynı olmalıdır.
- Diziler referanslı olarak çağrılan fonksiyonlara gönderilirler.
- Bir fonksiyon dizi üzerinde değişiklik yaparsa, orijinal dizideki eleman değeri de değişir.

21

Fonksiyonlara Dizi Gönderilmesi

```
public class DiziOrnekler {  
  
    public static double ortalamaHesapla(int sicaklik[]) {  
        int toplamsicaklik = 0;  
        for (int i = 0; i < sicaklik.length; i++) {  
            toplamsicaklik = toplamsicaklik + sicaklik[i];  
        }  
        return (double) toplamsicaklik / sicaklik.length;  
    }  
  
    public static void main(String[] args) {  
        Scanner klavye = new Scanner(System.in);  
        int sicaklik[] = new int[30];  
  
        for (int i = 0; i < 30; i++) {  
            System.out.print((i + 1) + ". gün sıcaklık = ");  
            sicaklik[i] = klavye.nextInt();  
        }  
  
        double ortalama = ortalamaHesapla(sicaklik);  
  
        System.out.printf("Ay sıcaklık ortalaması = %.2f", ortalama);  
    }  
}
```

```
1. gün sıcaklık = 1  
2. gün sıcaklık = 2  
3. gün sıcaklık = 3  
4. gün sıcaklık = 2  
5. gün sıcaklık = 1  
6. gün sıcaklık = 3  
7. gün sıcaklık = 5  
8. gün sıcaklık = 6  
9. gün sıcaklık = 4  
10. gün sıcaklık = 2  
11. gün sıcaklık = 1  
12. gün sıcaklık = 3  
13. gün sıcaklık = 5  
14. gün sıcaklık = 4  
15. gün sıcaklık = 1  
16. gün sıcaklık = 2  
17. gün sıcaklık = 5  
18. gün sıcaklık = 6  
19. gün sıcaklık = 2  
20. gün sıcaklık = 1  
21. gün sıcaklık = 5  
22. gün sıcaklık = 4  
23. gün sıcaklık = 8  
24. gün sıcaklık = 9  
25. gün sıcaklık = 2  
26. gün sıcaklık = 1  
27. gün sıcaklık = 3  
28. gün sıcaklık = 2  
29. gün sıcaklık = 5  
30. gün sıcaklık = 2  
Ay sıcaklık ortalaması = 3,33
```

22

Fonksiyonlara Dizi Gönderilmesi

```
public class DiziOrnekler {  
  
    public static void donustur(double sicaklik[]) {  
        for (int i = 0; i < sicaklik.length; i++)  
            sicaklik[i] = sicaklik[i] * 1.8 + 32;  
    };  
  
    public static void main(String[] args) {  
  
        double[] sicaklik = new double[3];  
  
        Scanner klavye = new Scanner(System.in);  
  
        for (int i = 0; i < sicaklik.length; i++) {  
            System.out.print((i + 1) + ". gün sıcaklık (derece) = ");  
            sicaklik[i] = klavye.nextInt();  
        }  
  
        donustur(sicaklik);  
  
        for (int i = 0; i < sicaklik.length; i++) {  
            System.out.println((i+1) + ". gün sıcaklık (fahrenheit) = " +sicaklik[i]);  
        }  
    }  
}
```

1. gün sıcaklık (derece) = 1
2. gün sıcaklık (derece) = 2
3. gün sıcaklık (derece) = 3
1. gün sıcaklık (fahrenheit) = 33.8
2. gün sıcaklık (fahrenheit) = 35.6
3. gün sıcaklık (fahrenheit) = 37.4

23

Fonksiyonlara Dizi Gönderilmesi

```
public class DiziOrnekler {  
  
    public static boolean esit(int sicaklik1[], int sicaklik2[]) {  
        boolean esit = true;  
        for (int i = 0; i < sicaklik1.length; i++) {  
            esit = (sicaklik1[i] == sicaklik2[i]) ? true : false;  
            if (!esit) break;  
        }  
        return esit;  
    }  
  
    public static void main(String[] args) {  
        Scanner klavye = new Scanner(System.in);  
        int sicaklik1[] = new int[3];  
        int sicaklik2[] = new int[3];  
  
        System.out.print("1. dizinin değerleri giriliyor...\n");  
        for (int i = 0; i < sicaklik1.length; i++) {  
            System.out.print((i + 1) + ". değer = ");  
            sicaklik1[i] = klavye.nextInt();  
        }  
  
        System.out.print("2. dizinin değerleri giriliyor...\n");  
        for (int i = 0; i < sicaklik2.length; i++) {  
            System.out.print((i + 1) + ". değer = ");  
            sicaklik2[i] = klavye.nextInt();  
        }  
    }  
}
```

24

Fonksiyonlara Dizi Gönderilmesi

```
System.out.println("\nsicaklik1 == sicaklik2 karşılaştırmasının sonucu");
if (sicaklik1 == sicaklik2)
    System.out.println("Diziler birbirine eşit");
else System.out.println("Diziler birbirine eşit değil");

System.out.println("\nesit() fonksiyonu karşılaştırma sonucu");
if (esit(sicaklik1, sicaklik2))
    System.out.println("Diziler birbirine eşit");
else System.out.println("Diziler birbirine eşit değil");
}
```

```
1. dizinin değerleri giriliyor...
1. değer = 1
2. değer = 2
3. değer = 3
2. dizinin değerleri giriliyor...
1. değer = 1
2. değer = 2
3. değer = 3

sicaklik1 == sicaklik2 karşılaştırmasının sonucu
Diziler birbirine eşit değil

esit() fonksiyonu karşılaştırma sonucu
Diziler birbirine eşit
```

25

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- **Fonksiyonlardan Dizi Döndürülmesi**
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Fonksiyonlardan Dizi Döndürülmesi

- Bir fonksiyondan dizi elemanlarının tamamı doğrudan dizi ismiyle döndürülür.

```
public class DiziOrnekler {  
  
    public static boolean esit(int sicaklik1[], int sicaklik2[]) {  
        boolean esit = true;  
        for (int i = 0; i < sicaklik1.length; i++) {  
            esit = (sicaklik1[i] == sicaklik2[i]) ? true : false;  
            if (!esit) break;  
        }  
        return esit;  
    }  
  
    public static int[] sicaklikgirisi() {  
        Scanner klavye = new Scanner(System.in);  
        int sicaklik[] = new int [3];  
        for (int i = 0; i < sicaklik.length; i++) {  
            System.out.print((i + 1) + ". deger = ");  
            sicaklik[i] = klavye.nextInt();  
        }  
        return sicaklik;  
    }  
}
```

27

Fonksiyonlardan Dizi Döndürülmesi

```
public static void main(String[] args) {  
    int sicaklik1[] = new int[3];  
    int sicaklik2[] = new int[3];  
  
    System.out.print("1. dizinin degerleri giriliyor...\n");  
    sicaklik1 = sicaklikgirisi();  
  
    System.out.print("2. dizinin degerleri giriliyor...\n");  
    sicaklik2 = sicaklikgirisi();  
  
    System.out.println("\nsicaklik1 == sicaklik2 karšılařtırmasının sonucu");  
    if (sicaklik1 == sicaklik2)  
        System.out.println("Diziler birbirine eřit");  
    else System.out.println("Diziler birbirine eřit deęil");  
  
    System.out.println("\nesit() fonksiyonu karšılařtırma sonucu");  
    if (esit(sicaklik1, sicaklik2))  
        System.out.println("Diziler birbirine eřit");  
    else System.out.println("Diziler birbirine eřit deęil");  
}
```

28

Fonksiyonlardan Dizi Döndürülmesi

```
1. dizinin değerleri giriliyor...
1. değer = 1
2. değer = 2
3. değer = 3
2. dizinin değerleri giriliyor...
1. değer = 1
2. değer = 2
3. değer = 3

sicaklik1 == sicaklik2 --> karşılaştırması sonucu
Diziler birbirine eşit değil

esit() --> fonksiyonu sonucu
Diziler birbirine eşit.
```

29

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- **Dizilerde Sıralama**
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Dizilerde Sıralama

- Dizi elemanlarını büyükten küçüğe doğru sıralamak için `java.util` paketini altındaki **Arrays sınıfı kullanılabilir.**
- Bu sınıfın statik **sort() yordamı** sayesinde dizilerin içerisindeki **elemanlar sıralanabilir.**
- Dizi elemanları **kapsayan sınıf (wrapper class)** şeklinde oluşturulursa sıralama yönü de değiştirilebilir.

31

Dizilerde Sıralama

```
import java.util.*; // java.util kütüphanesini kullanmak için

public class DiziOrnekler {
    public static void ekranaBas(Double[] d) {
        for (int i = 0; i < d.length; i++) {
            System.out.println("d[" + i + "] = " + d[i]);
        }
    }

    public static void main(String args[]) {
        Double d[] = new Double[9];
        d[0] = 2.45;
        d[1] = 3.45;
        d[2] = 4.78;
        d[3] = 1.45;
        d[4] = 15.12;
        d[5] = 1.0;
        d[6] = 9.0;
        d[7] = 15.32;
        d[8] = 78.17;

        System.out.println("Sırasız dizi");
        ekranaBas(d);

        Arrays.sort(d);
        System.out.println("Artan Sırada Sıralanmış dizi");
        ekranaBas(d);

        Arrays.sort(d, Collections.reverseOrder());
        System.out.println("Azalan Sırada Sıralanmış dizi");
        ekranaBas(d);
    }
}
```

```
Sırasız dizi
d[0] = 2.45
d[1] = 3.45
d[2] = 4.78
d[3] = 1.45
d[4] = 15.12
d[5] = 1.0
d[6] = 9.0
d[7] = 15.32
d[8] = 78.17
Artan Sırada Sıralanmış dizi
d[0] = 1.0
d[1] = 1.45
d[2] = 2.45
d[3] = 3.45
d[4] = 4.78
d[5] = 9.0
d[6] = 15.12
d[7] = 15.32
d[8] = 78.17
Azalan Sırada Sıralanmış dizi
d[0] = 78.17
d[1] = 15.32
d[2] = 15.12
d[3] = 9.0
d[4] = 4.78
d[5] = 3.45
d[6] = 2.45
d[7] = 1.45
d[8] = 1.0
```

32

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- **Dizilerin Kopyalanması**
- Çok Boyutlu Diziler

Dizilerin Kopyalanması

- Bir dizinin tüm elemanları başka bir diziye kopyalanabilir.
- Dizi elemanları teker teker atanabilir.
- **System.arraycopy()** fonksiyonu kullanılabilir.

```
public class DiziOrnekler {  
  
    public static void main(String args[]) {  
        int[] dizi1 = { 1, 2, 3, 4 }; // ilk dizi  
        int[] dizi2 = { 100, 90, 78, 45, 40, 30, 20, 10 }; // daha geniş olan 2. dizi  
  
        // kopyalama işlemi başlıyor  
        // 0. indisinden dizi1 uzunluğu kadar kopyalama yap  
        System.arraycopy(dizi1, 0, dizi2, 0, dizi1.length);  
        for (int i = 0; i < dizi2.length; i++) {  
            System.out.println("dizi2[" + i + "] = " + dizi2[i]);  
        }  
    }  
}
```

```
dizi2[0] = 1  
dizi2[1] = 2  
dizi2[2] = 3  
dizi2[3] = 4  
dizi2[4] = 40  
dizi2[5] = 30  
dizi2[6] = 20  
dizi2[7] = 10
```

Konular

- Diziler
- Dizi İşlemleri
- Nesne Dizileri
- Fonksiyonlara Dizi Gönderilmesi
- Fonksiyonlardan Dizi Döndürülmesi
- Dizilerde Sıralama
- Dizilerin Kopyalanması
- Çok Boyutlu Diziler

Çok Boyutlu Diziler

- Çok boyutlu dizilerde de, dizi elemanlarının aynı türde olması gereklidir.
- İki boyutlu diziler **matris** verilerinin **gösteriminde kullanılabilir**.

```
int[][] t1 = {  
    { 1, 2, 3, },  
    { 4, 5, 6, }  
};
```

```
int[][] t1 = new int[3][4];  
int[][] t1 = new int[][4]; // ! Hata !
```

Çok Boyutlu Diziler

```
public class DiziOrnekler {  
  
    public static void main(String args[]) {  
        int ikiboyutlu[][] = new int[3][4];  
        ikiboyutlu[0][0] = 45;  
        ikiboyutlu[0][1] = 83;  
        ikiboyutlu[0][2] = 11;  
        ikiboyutlu[0][3] = 18;  
        ikiboyutlu[1][0] = 17;  
        ikiboyutlu[1][1] = 56;  
        ikiboyutlu[1][2] = 26;  
        ikiboyutlu[1][3] = 79;  
        ikiboyutlu[2][0] = 3;  
        ikiboyutlu[2][1] = 93;  
        ikiboyutlu[2][2] = 43;  
        ikiboyutlu[2][3] = 12;  
        // ekrana yazdırıyoruz  
        for (int i = 0; i < ikiboyutlu.length; i++) {  
            for (int j = 0; j < ikiboyutlu[i].length; j++) {  
                System.out.println(" ikiboyutlu[" + i + "][" + j + "] = "  
                    + ikiboyutlu[i][j]);  
            }  
        }  
    }  
}
```

```
ikiboyutlu[0][0] = 45  
ikiboyutlu[0][1] = 83  
ikiboyutlu[0][2] = 11  
ikiboyutlu[0][3] = 18  
ikiboyutlu[1][0] = 17  
ikiboyutlu[1][1] = 56  
ikiboyutlu[1][2] = 26  
ikiboyutlu[1][3] = 79  
ikiboyutlu[2][0] = 3  
ikiboyutlu[2][1] = 93  
ikiboyutlu[2][2] = 43  
ikiboyutlu[2][3] = 12
```

```
45 83 11 18  
17 56 26 79  
3 93 43 12
```

37

Çok Boyutlu Diziler

- Dizilerdeki boyutlar aynı uzunlukta olmayabilir.

```
public class DiziOrnekler {  
    public static void main(String args[]) {  
        int ikiboy[][] = new int[3][];  
  
        ikiboy[0] = new int[2];  
        ikiboy[1] = new int[1];  
        ikiboy[2] = new int[3];  
  
        ikiboy[0][0] = 89;  
        ikiboy[0][1] = 32;  
        ikiboy[1][0] = 5;  
  
        // ikiboy[1][1] = 3 ; // ! Hata !  
        ikiboy[2][0] = 55;  
        ikiboy[2][1] = 78;  
        ikiboy[2][2] = 49;  
    }  
}
```

38

Çok Boyutlu Diziler

```
public class DiziOrnekler {
    String isimler[][][];

    public DiziOrnekler() {
        isimler = new String[2][2][3];
        veriAta();
    }

    public void veriAta() {
        isimler[0][0][0] = "aslan";
        isimler[0][0][1] = "boz AyI";
        isimler[0][0][2] = "ceylan";
        isimler[0][1][0] = "deniz Anası";
        isimler[0][1][1] = "essek";
        isimler[0][1][2] = "fare";
        isimler[1][0][0] = "geyik";
        isimler[1][0][1] = "hamsi";
        isimler[1][0][2] = "inek";
        isimler[1][1][0] = "japon baligi";
        isimler[1][1][1] = "kedi";
        isimler[1][1][2] = "lama";
        ekranaBas();
    }
}
```

39

Çok Boyutlu Diziler

```
public void ekranaBas() {
    for (int x = 0; x < isimler.length; x++) {
        for (int y = 0; y < isimler[x].length; y++) {
            for (int z = 0; z < isimler[x][y].length; z++) {
                System.out.println("isimler[" + x + "][" + y + "][" + z + "] =" +
                    isimler[x][y][z]);
            }
        }
    }
}

public static void main(String args[]) {
    DiziOrnekler hayvanlaralemi = new DiziOrnekler();
}
```

```
isimler[0][0][0] = aslan
isimler[0][0][1] = boz AyI
isimler[0][0][2] = ceylan
isimler[0][1][0] = deniz Anası
isimler[0][1][1] = essek
isimler[0][1][2] = fare
isimler[1][0][0] = geyik
isimler[1][0][1] = hamsi
isimler[1][0][2] = inek
isimler[1][1][0] = japon baligi
isimler[1][1][1] = kedi
isimler[1][1][2] = lama
```

40