

Java ile Nesneye Yönelik Programlama (Object Oriented Programming)

Giriş

Doç. Dr. Aybars UĞUR

Örnek 1

Bir Yolcu sınıfı, yolcu1 nesnesi oluşturulması ve kullanılması

```
class Yolcu {  
  
    String ad;  
    String soyad;  
    int yasi;  
  
    Yolcu() { };  
  
    Yolcu(String ad, String soyad)  
    {  
        this.ad = ad; this.soyad = soyad;  
    }  
  
    public void yazdir()  
    {  
        System.out.println("Ad   : "+ad);  
        System.out.println("Soyad : "+soyad);  
    }  
}
```

```
class Ornek_Class {  
  
    public static void main(String args[])  
    {  
        Yolcu yolcu1 = new Yolcu("Ali","Yilmaz");  
        yolcu1.yazdir();  
    }  
}
```

TEMEL BİLGİ ve TERMİNOLOJİ

I

Sınıf (Class) : Soyut bir veri tipinin hem verilen tiplerdeki veriler kümesini, hem de bu değerler üzerinde yapılabilecek işlemler kümesini bir araya getirir.

Örnek : "Yolcu" sınıfı.

Nesne (Object) : Sınıf tipindeki değişkenlere nesne adı verilir.

Örnek : "yolcu1" nesnesi.

Metot (Method) : Bir eylemi veya işlemi gerçekleştiren sınıf üyesidir.

"Yolcu()" yapıcı metotları ve "yazdir()" metodu "Yolcu" sınıfının metotlarıdır.

Sınıf Üyeleri (Class Members) : Sınıfın elemanlarına üye adı verilir.

Değişkenler, metotlar ...

Örnekler : "ad", "soyad", "yasi" değişkenleri; "Yolcu()" yapıcı metotları ve "yazdir()" metodu "Yolcu" sınıfının üyeleridir.

TEMEL BİLGİ ve TERMİNOLOJİ

II

Yapılandırıcı (Yapıcı) metot (Constructor) : Sınıftan yeni bir nesne yaratıldığı anda çağrılan metoda yapıcı metot adı verilir. Yapıcı metot ismi, sınıf ismi ile aynıdır.

Yolcu yolcu1 = new Yolcu("Ali","Yilmaz");

"yolcu1" nesnesi "new" deyimi ile oluşturulurken "Yolcu" sınıfının iki tane "String" parametre alan yapıcı metodu devreye girer.

// Yapıcı metot

Yolcu(String ad, String soyad)

```
{  
    this.ad = ad; this.soyad = soyad;  
}
```

Örnek 2

Vector Uygulaması

Bir vektör oluşturarak sırayla "Ali", "Cemil", "Kemal" isimlerini ekleyiniz. Vektörü iki şekilde dolaşarak (for, enum) isimleri ekrana listeleyiniz.

```
import java.util.*;

class Vektor1
{
    public static void main(String args[])
    {
        final Vector v = new Vector(1);
        v.addElement("Ali");
        v.addElement("Cemil");
        v.addElement("Kemal");

        for(int i=0; i<v.size(); ++i)
            System.out.println(v.elementAt(i));

        Enumeration enum = v.elements();
        while(enum.hasMoreElements())
            System.out.println(enum.nextElement());
    }
}
```

Örnek 3

Bir Yolcu (ad, yas) sınıfı oluşturunuz. Yolcu sınıfından yararlanarak oluşturduğunuz iki yolcuyu bir vektöre yerleştiriniz. "for" döngüsü ile tersten dolaşınız.

```
import java.util.*;
```

```
class Yolcu
```

```
{  
    String ad;  
    int yas;  
    public Yolcu(String ad, int yas)  
    {  
        this.ad = ad;  
        this.yas = yas;  
    }  
}
```

```
class Vektor2
```

```
{  
  
    public static void main(String args[])  
    {  
        final Vector v = new Vector(1);  
  
        Yolcu y1 = new Yolcu("Ali",25);  
        v.addElement(y1);  
  
        Yolcu y2 = new Yolcu("Zekiye",15);  
        v.addElement(y2);  
  
        for(int i=0; i<v.size(); ++i)  
        {  
            Yolcu y = (Yolcu)v.elementAt(i);  
            System.out.println(y.ad+" "+y.yas);  
        }  
    }  
}
```

ALIŞTIRMALAR

1. Örnek 2'yi, vektöre isme göre sıralı olarak eleman ekleyecek şekilde değiştiriniz.
2. Örnek 2'yi, vektöre yasa göre sıralı olarak eleman ekleyecek şekilde değiştiriniz.
3. Örnek 2'de, ismi verilen bir yolcuyu silen metodu ekleyiniz.
4. Örnek 2'de, yaşı verilen bir yolcuyu silen metodu ekleyiniz.
5. Örnek 2'de bir yolcunun bilgisini günleyen metodu ekleyiniz.

Örnek 4 - 1

```
import java.util.*;
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class vektor extends JFrame
{
    int yer = -1;

    public vektor()
    {
        super("Vektor Ornek");
        Container c = getContentPane();
        c.setLayout(new FlowLayout());

        final Vector v = new Vector(1);

        final JTextField tf = new JTextField(10);
        c.add(tf);
```

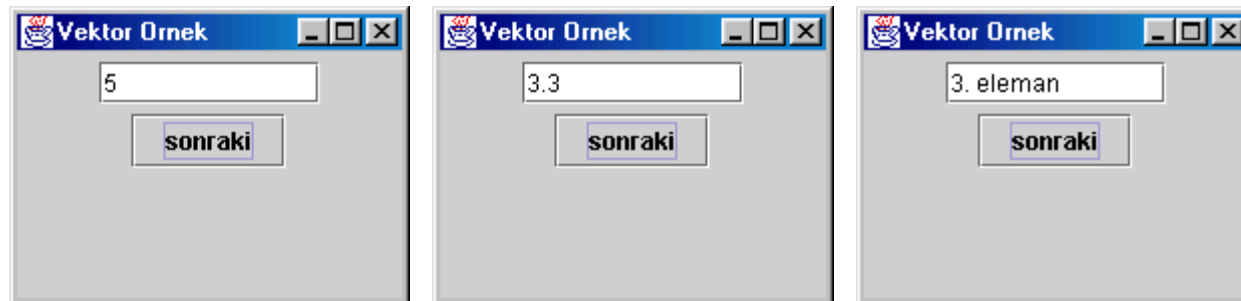
```
        final JButton sonraki = new JButton("sonraki");
        sonraki.addActionListener
        (
            new ActionListener()
            {
                public void actionPerformed(ActionEvent e)
                {
                    //tf.setText((v.firstElement()).toString());
                    if (yer<(v.size()-1)) ++yer; else yer = 0;
                    tf.setText((v.elementAt(yer)).toString());
                }
            }
        );
        c.add(sonraki);

        Double d = new Double(3.3);
        Integer i = new Integer(5);
        v.addElement(i);
        v.addElement(d);
        v.addElement("3. eleman");
        setSize(200,150);
        show();
    }
}
```


Örnek 4 - 2

```
public static void main ( String args[] )
{
    vektor app = new vektor();

    app.addWindowListener
    (
        new WindowAdapter()
        {
            public void windowClosing(WindowEvent e)
            {
                System.exit(0);
            }
        }
    );
}
```



Örnek 5 – 1

Telefon – GUI Bileşenleri

```
import java.awt.*;
import java.awt.event.*;
import javax.swing.*;

public class telefon extends JFrame implements
    ActionListener
{
    public Container c;
    private String names[] = { "1", "2", "3", "4", "5",
        "6", "7", "8", "9" };
    private JButton b[];
    private JTextField tf1;

    public telefon()
    {
        super("telefon");

        c = getContentPane();
        c.setLayout(new BorderLayout());

        b = new JButton[names.length];
```

```
JPanel p1 = new JPanel();
    for(int i=0; i<names.length; ++i) {
        b[i] = new JButton(names[i]);
        b[i].addActionListener(this);
        p1.add(b[i]); }
    tf1 = new JTextField();

    c.add(p1,BorderLayout.CENTER);
    c.add(tf1,BorderLayout.NORTH);
    setSize(150,150);
    show();
}

public void actionPerformed(ActionEvent e)
{
    if(e.getSource()==b[0])
        tf1.setText(tf1.getText()+"1");
    else
    if(e.getSource()==b[1])
        tf1.setText(tf1.getText()+"2");
    else
    if(e.getSource()==b[2])
        tf1.setText(tf1.getText()+"3");
}
```

Örnek 5 – 2

Telefon – GUI Bileşenleri

```
public static void main(String args[])
{
    telefon app = new telefon();
    app.addWindowListener(
        new WindowAdapter() {
            public void windowClosing(WindowEvent e)
            {
                System.exit(0);
            }
        }
    );
}
```

