

Java GUI Frame Sınıfı

Bir kullanıcı arayüzü oluşturmak için ya bir tane **JFrame** nesnesi veya Applet oluşturmak gerekmektedir.

Frame Yaratmak

Frame yaratmak için **JFrame** sınıfı kullanılmaktadır.

Örnek Bir Uygulama

```
1 import javax.swing.JFrame;
2 import javax.swing.WindowConstants;
3 public class MyFrame {
4     public static void main(String[] args) {
5         JFrame frame = new JFrame("MyFrame"); //Frame nesnesi oluşturulur
6         frame.setSize(400, 300); // Frame'in boyutu belirlenir
7         // Frame'in ekranın tam ortasında açılmasını sağlar
8         frame.setLocationRelativeTo(null);
9         /*jdk 1.7 de , JFrame.EXIT_ON_CLOSE yerine WindowConstants.EXIT_ON_CLOSE kullanılması tavsiye edilir */
10        frame.setDefaultCloseOperation(WindowConstants.EXIT_ON_CLOSE);
11        frame.setVisible(true); // Frame'in gösterilmesini sağlar
12    }
13 }
```

Constructors

JFrame(): Başlıksız frame oluşturur.

JFrame(title: String): Başlığı girilen parametre değerine göre belir bir frame oluşturmayı sağlar. Örnek: **JFrame("Hello World")**; denildiğinde "Hello World" başlıklı bir frame oluşturulur.

Methods

setSize(width: int, height: int): void : Frame'in boyutunu belirlemeyi sağlar.(void: dönüş tipi)

setVisible(visible: boolean): void : Frame'in gösterilebilmesi için true değer alması gerekir.

setDefaultCloseOperation(mode: int): void : Frame kapatıldığında programın yapacağı işlemi belirtmeyi sağlar. Parametre olarak aldığı değerler şunlardır:

1. Jdk 1.7 den önceki versiyonlarda:

JFrame.EXIT_ON_CLOSE : Frame kapatıldığında programda kapanır

JFrame.DO_NOTHING_ON_CLOSE : Frame kapatılmasından program etkilenmez

JFrame.DISPOSE_ON_CLOSE : Frame kapatıldığında program başka işlem varsa çalışmaya devam eder, yoksa kapanır.

JFrame.HIDE_ON_CLOSE : Frame kapatıldığında memory'den silinmez, tekrar çağrılmayı bekler. Program kapanmaz.

2. Jdk 1.7 ve sonraki versiyonlarda:

WindowConstants.EXIT_ON_CLOSE : Frame kapatıldığında programda kapanır

WindowConstants.DO_NOTHING_ON_CLOSE : Frame kapatılmasından program etkilenmez

WindowConstants.DISPOSE_ON_CLOSE : Frame kapatıldığında program başka işlem varsa çalışmaya devam eder, yoksa kapanır.

WindowConstants.HIDE_ON_CLOSE : Frame kapatıldığında memory'den silinmez, tekrar çağrılmayı bekler. Program kapanmaz.

Not: Eğer **setDefaultCloseOperation()** metodunu kullanmazsak, frame kapatıldığında varsayılan değer **HIDE_ON_CLOSE** olduğu için program kapanmaz.

setLocationRelativeTo(c: Component): void : Parametre olarak girilen componente göre ekranda nasıl gösterileceğini belirler. Eğer parametre olarak null girilirse, ekranın ortasında gösterilir.

Not: **JButton**, **JLabel**, **JTextField**, **JPanel**, vs sınıflara component denir.

pack(): void : Frame'de bulunan componentlerin büyüklüğüne göre tüm componentleri barındırmak için frame'in boyutunu otomatik olarak belirlenmesini sağlar.

Not: Eğer **setSize** metodundan sonra, **pack** metodunu kullanırsak, **setSize** metodunda girilen boyuta göre değil **pack** metodunu göre, frame'in boyutu belirlenir.

Not: **setSize** metodunu **setLocationRelativeTo(c: Component)** metodundan önce yazmazsak **setLocationRelativeTo(c:Component)** metodu istenen sonucu vermez.

Frame'e Component Ekleme

Her **JFrame** bir content pane (içerik bölümü) içerir. Bu content pane, java.awt.**Container** sınıfından türetilmiş bir nesnedir. JFrame'e eklenen **JButton** gibi componentler bu content pane yerleştirilir. JDK'nın önceki versiyonlarında, **JFrame**'e bir component eklemek istediğimizde **getContentPane** metodunu kullanarak content pane nesnesini döndürür, daha sonra bu nesnenin **add** metodunu kullanarak component eklerdik:

```
1 JFrame frame= new JFrame();
2 JButton jbtOK= new JButton("OK");
3 java.awt.Container container = frame.getContentPane();
4 container.add(jbtOK);
```

Java 5. versiyon ile yukarıdaki kodların yaptığı işlem şu şekilde kısalmıştır:

```
1 JFrame frame= new JFrame();  
2 JButton jbtOK= new JButton("OK");  
3 frame.add(jbtOK);
```

Not: Sonu olarak, bir frame nesnesine component eklendiğinde aslında bu component frame nesnesine deęil, frame nesnesinin barındırdığı content pane nesnesine eklenmektedir.

Frame nesnesinden bir component ıkarmak istediğimizde remove(c:[Component](#)) metodu kullanılır.