



**ERCIYES ÜNİVERSİTESİ MÜHENDİSLİK FAKÜLTESİ
BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ BİLGİSAYAR
PROGRAMLAMA DERSİ LABORATUVAR FÖYÜ**



DENEY FÖYÜ 13

DENEY ADI

JAVA ile GRAFİK UYGULAMASI

DERSİN ÖĞRETİM ÜYESİ

Prof. Dr. Bahriye AKAY

DENEYİ YAPTIRAN ÖĞRETİM ELEMANI

Arş. Gör. Abdalbaki KARTAL

Öğrenci Numarası:

Öğrenci Ad Soyadı:

Deney Tarihi:

**2026
KAYSERİ**

1. JAVA PROGRAMLAMA DİLİ VE GRAFİK KÜTÜPHANELERİNE HAZIRLIK

1.1. GİRİŞ

Bu deney föyü, derste yapılacak Java tabanlı grafik uygulamasına hazırlık amacıyla hazırlanmıştır. Föy kapsamında uygulamanın konusu açıklanmayacak; öğrenciler derse Java dilinin temel yapısı, değişken tanımlama, koşullu ifadeler, metod kullanımı ve grafik programlamada ihtiyaç duyulacak temel kütüphaneler hakkında ön bilgi edinerek gelecektir.

Java, nesne yönelimli programlama yaklaşımını destekleyen, platform bağımsız çalışabilen ve geniş standart kütüphane desteğine sahip bir programlama dilidir. Bir Java programı genellikle sınıflar, metodlar ve nesnelere etrafında organize edilir. Bu laboratuvar çalışmasında amaç, öğrencilerin Java sözdizimini temel düzeyde tanıması ve derste kullanılacak grafik kütüphanelerinin ne işe yaradığını kavramasıdır.

1.2. JAVA PROGRAMLAMA DİLİNİN GENEL YAPISI

Java programlarında kodlar sınıf yapısı içinde yazılır. Çalıştırılabilir bir Java programında main metodu bulunur. Program çalışmaya main metodundan başlar. Dosya adı ile public class adının aynı olması gerekir.

```
public class GrafikUygulaması {
    public static void main(String[] args) {
        System.out.println("Merhaba Java");
    }
}
```

Yukarıdaki örnekte GrafikUygulaması sınıfı adıdır. main metodu programın başlangıç noktasıdır. System.out.println() komutu ise konsola çıktı yazdırmak için kullanılır.

- public class: Programdaki sınıfın tanımlandığı bölümdür.
- main metodu: Programın çalışmaya başladığı ana metottur.

- System.out.println(): Konsola yazı veya değişken değeri yazdırır.
- Süslü parantezler { }: Kod bloklarının sınırlarını belirtir.

1.3. DEĞİŞKEN TANIMLAMA VE VERİ TİPLERİ

Değişkenler, program içerisinde veri saklamak için kullanılır. Java güçlü tip denetimine sahip bir dildir; bu nedenle her değişken tanımlanırken veri tipi belirtilmelidir.

Veri Tipi	Açıklama	Örnek
int	Tam sayı değerleri tutar.	int sayı = 10;
double	Ondalıklı sayıları tutar.	double oran = 3.14;
boolean	true veya false değerini tutar.	boolean aktif = true;
char	Tek karakter tutar.	char harf = 'A';
String	Metinsel ifadeleri tutar.	String ad = "Ali";

```
int x = 100;
int y = 200;
String mesaj = "Koordinat bilgisi";
boolean durum = true;
```

Grafik tabanlı Java uygulamalarında değişkenler; koordinat bilgisi, genişlik, yükseklik, renk seçimi, durum kontrolü ve kullanıcı etkileşimi gibi işlemlerde kullanılabilir.

1.4. KOŞULLU İFADELER: IF - ELSE YAPISI

Koşullu ifadeler, programın belirli durumlara göre farklı işlemler yapmasını sağlar. Java'da en temel karar yapısı if - else if - else yapısıdır.

```
int puan = 75;

if (puan >= 60) {
    System.out.println("Geçti");
} else {
    System.out.println("Kaldı");
}
```

Grafik uygulamalarında if - else yapıları; hangi çizimin yapılacağına, hangi rengin kullanılacağına veya kullanıcıdan gelen bir seçime göre hangi işlemin çalıştırılacağına karar vermek için kullanılabilir.

```
String secim = "cizgi";

if (secim.equals("cizgi")) {
    System.out.println("Çizgi çizilecek");
} else if (secim.equals("daire")) {
    System.out.println("Daire çizilecek");
} else {
    System.out.println("Geçersiz seçim");
}
```

1.5. FONKSİYONLAR / METOTLAR

Java'da fonksiyonlar genellikle metot olarak adlandırılır. Metotlar, belirli bir işlemi tekrar kullanılabilir hâle getirmek için kullanılır. Bir metot parametre alabilir, işlem yapabilir ve sonuç döndürebilir.

```
public static int topla(int a, int b) {
    return a + b; }
```

Metotlar kod tekrarını azaltır, programı daha okunabilir yapar ve büyük problemleri küçük parçalara ayırmayı sağlar. Grafik programlamada da farklı çizim adımlarını ayrı metotlar hâline getirmek mümkündür.

```
public static void bilgiYazdir(String mesaj) {
    System.out.println(mesaj); }
```

- void: Metodun geriye değer döndürmediğini belirtir.
- return: Metodun sonucunu çağrıldığı yere döndürür.
- Parametre: Metoda dışarıdan gönderilen değerdir.

1.6. KÜTÜPHANE ENTEGRASYONU VE IMPORT KAVRAMI

Java'da hazır sınıfları ve paketleri kullanabilmek için import ifadesinden yararlanılır. Kütüphaneler, programcıya önceden hazırlanmış sınıflar ve metotlar sunar. Böylece pencere oluşturma, buton ekleme, renk tanımlama, çizim yapma ve olay yönetimi gibi işlemler daha kolay gerçekleştirilebilir.

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

import javax.swing.* ifadesi Swing kütüphanesindeki sınıfları, import java.awt.* ifadesi ise temel grafik ve arayüz bileşenleriyle ilgili sınıfları programa dâhil eder. Yıldız (*) işareti, ilgili paketteki birden fazla sınıfın kullanılabilmesini belirtir.

1.7. DERSTE KULLANILACAK TEMEL JAVA KÜTÜPHANELERİ

Bu laboratuvar çalışmasında özellikle Java'nın standart grafik ve arayüz kütüphaneleri üzerinde durulacaktır. Öğrencilerin derse gelmeden önce aşağıdaki sınıfların ne işe yaradığını genel hatlarıyla incelemesi beklenmektedir.

Kütüphane / Sınıf	Görevi	Kullanım Amacı
javax.swing.JFrame	Pencere oluşturma için kullanılır.	Grafik uygulamasının ana penceresini oluşturma
javax.swing.JPanel	Pencere içinde özel alan oluşturur.	Çizim yapılacak alanı tanımlama
java.awt.Graphics	Temel çizim işlemlerini sağlar.	Çizgi, şekil ve yazı çizme
java.awt.Color	Renk tanımlama için kullanılır.	Şekillere ve yazılara renk verme
java.awt.Font	Yazı tipi ve boyutu ayarlama için kullanılır.	Ekran yazarken biçimlendirme
java.awt.event	Olay yönetimi sınıflarını içerir.	Buton, fare veya klavye olaylarını işleme

Derste kullanılacak bazı temel çizim komutları aşağıda verilmiştir. Bu komutların hangi parametrelerle çalıştığını incelemek,

uygulama sırasında öğrencilerin daha hızlı ilerlemesini sağlayacaktır.

Komut	Görevi
drawLine(x1, y1, x2, y2)	İki nokta arasında çizgi çizer.
drawRect(x, y, width, height)	Boş dikdörtgen veya kare çizer.
fillRect(x, y, width, height)	İçi dolu dikdörtgen veya kare çizer.
drawOval(x, y, width, height)	Boş oval veya daire çizer.
fillOval(x, y, width, height)	İçi dolu oval veya daire çizer.
drawPolygon(xPoints, yPoints, nPoints)	Çokgen çizer.
drawString(text, x, y)	Ekrana metin yazar.
setColor(Color.RED)	Sonraki çizimlerin rengini belirler.

1.8. VS CODE ÜZERİNDE JAVA ÇALIŞTIRMA

Java dosyası .java uzantısı ile kaydedilmelidir.

Terminal üzerinden bir Java dosyasını derlemek ve çalıştırmak için aşağıdaki komutlar kullanılabilir:

```
javac DosyaAdi.java  
java DosyaAdi
```

Örneğin public class adı GrafikUygulaması ise dosya adı GrafikUygulaması.java olmalıdır. Derleme işleminden sonra aynı klasörde .class uzantılı dosya oluşur. Bu dosya Java'nın derlenmiş çıktısıdır ve düzenlenmesi gerekmez.

1.9. DERS ÖNCESİ HAZIRLIK GÖREVLERİ

Derse gelmeden önce aşağıdaki hazırlıkların yapılması beklenmektedir:

- public class, main metodu ve System.out.println() komutlarının görevini inceleyiniz.
- int, double, boolean, char ve String veri tiplerini tekrar ediniz.
- if - else ve else if yapısının kullanımını örneklerle inceleyiniz.
- Java'da metod tanımlama, parametre gönderme ve return kullanımını tekrar ediniz.
- javax.swing, java.awt ve java.awt.event paketlerinin genel görevlerini araştırınız.
- Graphics sınıfındaki temel çizim komutlarının parametrelerini inceleyiniz.

1.10. DENEYİN AMACI

Bu deneyin amacı, öğrencilerin Java programlama dilinin temel sözdizimini ve grafik tabanlı uygulamalarda kullanılacak temel kütüphaneleri tanımasını sağlamaktır. Bu kapsamda öğrencilerin değişken tanımlama, koşullu ifade kullanma, metod oluşturma, hazır kütüphaneleri programa dâhil etme ve temel grafik çizim komutlarını anlama becerilerinin geliştirilmesi hedeflenmektedir.

1.11. KISA ÖZET

Bu föyde Java programlama dilinin temel yapısı, değişken tanımlama, if - else karar yapısı, metod kullanımı, import kavramı ve grafik uygulamalarında kullanılabilecek temel Java kütüphaneleri ele alınmıştır. Özellikle javax.swing, java.awt ve java.awt.event paketleri, grafik tabanlı uygulamalar için gerekli temel bileşenleri sağlamaktadır.

Öğrencilerin derse hazırlıklı gelebilmeleri için basit bir Java programını derleyip çalıştırabilmeleri ve Graphics sınıfının temel çizim komutlarını incelemeleri beklenmektedir. Derste yapılacak uygulamanın konusu ders sırasında açıklanacak ve öğrenciler bu föyde yer alan bilgilerden yararlanarak uygulamayı geliştirecektir.