



**T.C.**  
**SAKARYA ÜNİVERSİTESİ**

**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**  
PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU

**BELİRTİLEN SINIFI**  
**ANALİZ EDEN JAVA PROGRAMI**

**Grup Elemanları:**

**B161210037 - Burak YILDIRIM**

**B161210043 - İbrahim TURAN**

**SAKARYA**

**Mart, 2018**

Programlama Dillerinin Prensipleri Dersi

## Belirtilen Sınıfı Analiz Eden Java Programı

Burak Yıldırım, M.İbrahim Turan

B161210037

B161210043

---

### Özet

Ödevimizin amacı dışarıdan verilen bir sınıfı analiz ederek sınıf hakkında bilgileri çıktı olarak veren java programını geliştirmektir. Ödevin yapım aşamasında çeşitli araştırmalar sonucu Reflection yönteminin problemimiz için en uygun çözüm yöntemi olduğunu tespit ettik. Ödevi yaparken yaşadığımız tek ve en büyük problem metodların parametre isimlerini almak oldu. Kod çalışmasına rağmen parametre isimleri arg0,arg1 şeklinde dönüyordu. Bu problemi bir günümüzü ayırmamıza rağmen çözemedik. Araştırmalarımız sonucu sorunun IDE den kaynaklı olabileceğini öğrendik.

---

### 1. GELİŞTİRİLEN YAZILIM

Yazılımı geliştirmeye başlamadan önce ödev dökümanını inceleyip bizden isteneni iyice kavramaya çalıştık. Java dilinde sınıfın nasıl tanımlanacağı , değişkenlerin nasıl kullanıldığı vb. ile ilgili araştırmalar yaptık.Sonrasında ödevi yaparken nasıl bir yol izleyeceğimizi belirledik. Ödev çok fazla matematiksel işlem gerektirmediği için algoritma tasarlamamız uzun sürmedi. İlk başta dosya işlemleri ile verilen Java kaynak dosyasını satır satır okuyup analiz ederek çözüme ulaşabileceğimizi düşündük. Fakat çözüm yolları ile ilgili araştırmalar yaparken Reflection yöntemi ile karşılaştık. Bu yöntemin ödevin yapılması için en uygun çözüm yöntemi olduğuna karar verdik çünkü hem nesne yönelimli programlama prensibine uyuyordu hem de çok kullanışlı metodlar içeriyordu. Sonra nasıl kullanıldığına dair araştırma yapıp bilgi edindik ve kod yazmaya başladık.

Kodu yazarken öncelikle projemizi test etmek için birden fazla java kaynak dosyası yazdık.Sonra kullanıcının kaynak dosyayı programa nasıl entegre edeceği hakkında düşündük. Bunu Class türünde bir nesne tanımlayıp kullanıcıdan forName metodu ile dosya ismini alarak çözdük. Dosya isminin yanlış girilmesi ihtimaline karşın kodlarımızı try – catch bloğunun içine yazdık. Reflection metodlarını kullanarak istenen sınıfa ait tüm bilgileri çekip istenen formatta çıktı verecek şekilde düzenledik. Karşılaştığımız bir diğer sorunda eğer veri tipi String olursa ekrana class.java.lang String çıktısı geliyordu ve göze batıyordu. Bunu isimDuzelt adında bir metod yazarak çözdük.Karşılaştığımız son ve bizce en önemli sorun parametreden isim almak oldu. Kod çalışıyordu fakat parametre isimleri “present” olmadığı için ekrana arg0,arg1...argN şeklinde çıktılar veriyordu. Bu sorunu çözemedik. Araştırmalarımız sonucu sorunun IDE den kaynaklı bir sorun olduğu kanısına vardık. Çünkü Eclipse IDE sinde denediğimizde küçük ayarlamalar sonucu bir sıkıntı gözlemlemedik fakat NetBeans için çözüm bulamadık.

### 2.ÇIKTILAR

```

run:
Sinif ismini giriniz(uzanti olmadan): Sayi
Sinif Adi : Sayi
Alt Elemanlar :2 adet
Veri ismi : deger Tipi : int
Veri ismi : eksimi Tipi : boolean
Uye fonksiyonlar : 3 adet
Metod ismi: toString
Dönüs Tipi : String
Parametre Sayisi : 0
-----
Metod ismi: Eksimi
Dönüs Tipi : boolean
Parametre Sayisi : 0
-----
Metod ismi: setDeger
Dönüs Tipi : void
Parametre Sayisi : 2
arg0 int
arg1 int
-----
BUILD SUCCESSFUL (total time: 6 seconds)

```

**Burada programın , verilen iki farklı girdiye ait çıktıları görülmektedir.**

---

```

run:
Sinif ismini giriniz(uzanti olmadan): kedi
Sinif Bulunamadi Lütfen Tekrar Deneyiniz : NOT: Analiz etmek istediginiz .java uzantili dosyayi proje icerisindeki test dosyasinin icine atmayi unutmus olabilirsiniz
Veya .java dosyasına , package test eklememis olabilirsiniz
sakarya
Sinif Bulunamadi Lütfen Tekrar Deneyiniz : NOT: Analiz etmek istediginiz .java uzantili dosyayi proje icerisindeki test dosyasinin icine atmayi unutmus olabilirsiniz
Veya .java dosyasına , package test eklememis olabilirsiniz

```

**Eğer girilen girdi hata verirse, sistem resimdeki gibi hata fırlatmaktadır.**

### **3.SONUÇ**

Bu projeyi yaparken,Java dili daha yakından tanındı. Bir sınıfı analiz etmenin kodda çıkan hataları tespit etmeyi daha da kolaylaştırıldığı öğrenildi.Nesne yönelimli programlama prensibi daha iyi anlaşıldı.