

**T.C.**  
**SAKARYA ÜNİVERSİTESİ**  
**BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ**  
**BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ**

**Ders** : **Programlama Dillerinin Prensipleri**  
**Grup** : **1. Öğretim A**  
**Dersi Veren** : **Dr.Öğr.Üyesi Muhammed Fatih Adak**  
**Dönem** : **2023-2024 Bahar Dönemi**  
**Tarih** : **07.04.24**  
**Konu** : **1. Ödev**  
**Hazırlayan** : **Elif Özhan**  
**Öğrenci No** : **B221210001**  
**Öğrenci Mail** : [\*\*elif.ozhan1@ogr.sakarya.edu.tr\*\*](mailto:elif.ozhan1@ogr.sakarya.edu.tr)

Ödev Java programlama dilinde yazılmış olup Eclipse ortamında hazırlanmıştır. Bu proje “Program” isimli tek bir sınıf içerisinde Java sınıflarının değerlendirmeye alınmasını temel alır. Program içerisinde dosyaIsimleriniAl(String) : String[], javaKodAnalizi(String) : void, klonlaGitHubProjesi(String) : void, kopyalaJavaDosyaları(String) : void ve main(String[]) : void olarak beş adet fonksiyon bulunur. Detaylı olarak fonksiyonların ne yaptığını incelemek gerekirse:

### 1. klonlaGitHubProjesi(String) : void

Bu fonksiyon, program ilk çalıştığında kullanıcıdan alınmış olan GitHub linkinde bulunan dosyaların alınmasını ve bu projeye ait klonun oluşturulmasını sağlar. Proje linki alınır, geçici bir dosya oluşturulur. Daha sonrasında indirme işlemi gerçekleşir ve hedef klasöre klonlama yapılır. Son olarak geçici dosya silinir.

### 2. kopyalaJavaDosyaları(String) : void

Bu fonksiyon, daha sonra program içerisinde okuma yapıldığında klasör isminin bir sorun oluşturmasını engellemek adına tüm Java dosyalarının “Java\_files” isimli bir klasöre kopyalanmasını sağlar.

### 3. dosyaIsimleriniAl(String) : String[]

Bu fonksiyon, Java kodları analize tabi tutulacağı zaman her bir Java dosyasının sırayla ve kendi ismiyle çağrılıp işlem görmesini sağlamak adına tüm Java dosyalarının isimlerini bir dizide tutar. Geri dönüş değeri olan “dosyaIsimleri” bu String dizidir.

### 4. javaKodAnalizi(String) : void

Bu fonksiyon, Java dosyalarında bulunan kodların analize tabi tutulmasını sağlar. İlk olarak okuma işlemini başlatır. Satır satır kod içerisindeki değerlendirmeleri istenilen şekilde yapar. Örnek olarak “Yorum satırı mıdır değil midir?” gibi. Kullanılan karar yapılarıyla javadoc yorum satır sayısı, yorum satır sayısı, kod satır sayısı, LOC (her şey dahil kod satır sayısı), fonksiyon sayısı ve yorum sapma yüzdesi değerlerini her dosya için while döngüsü kullanarak hesaplar. Ekrana yazdırma işlemi de bu while döngüsünün içinde gerçekleştirilir. Kullanılan karar yapısıyla sınıf içermeyen Java dosyalarının ekran çıktısında yer alması engellenir.

### 5. main(String[]) : void

Bu fonksiyon, program açıldığında ilk çalışan fonksiyondur. İlk olarak kullanıcıdan GitHub linki ister ve bu linki gerekli işlemleri görmesi için klonlaGitHubProjesi(); fonksiyonuna parametre olarak gönderir. İşlem gerçekleştiğinde ekrana başarı mesajı gönderilir. İkinci olarak kopyalaJavaDosyaları(); fonksiyonunu çalıştırır ve işlem tamamlandığında yine ekrana başarı mesajı gönderir. Bu fonksiyonun hedef klasörü Java dosyalarını içeren klasör olacağı için parametre “Java\_files” olarak verilmiştir. Daha sonrasında Java\_files klasöründeki dosyaların isimleri dosyaIsimleriniAl(); fonksiyonu kullanılarak String dosya\_isimleri dizisine atılır ve ekrana başarı mesajı gönderilir. Son olarak program her dosya için analiz işlemini gerçekleştirmek adına bir for döngüsünün içinde javaKodAnalizi(); fonksiyonunu çalıştırır ve parametre olarak (for her çalıştığında bir sonraki Java dosyasını çağırarak üzere) dosya\_ismi’ni fonksiyona verir. Interface’lerin değerlendirmeye alınmaması için döngü içinde bir adet karar yapısı kullanılmıştır.

Sonuç olarak, program istenilen çıktıyı vermektedir. Hesaplamalarda herhangi bir hata bulunmamaktadır. Yorum sapma sayısı hesaplanırken sıfıra bölme hatası durumu için bu değer sıfıra eşitlenmiştir. Ödev hazırlanırken örnek GitHub repository olarak <https://github.com/mfadak/Odev1Ornek> kullanılmıştır. Readme dosyasındaki çıktıyla birebir aynı çıktı elde edilmiştir.