



SAKARYA  
ÜNİVERSİTESİ

T.C.  
SAKARYA ÜNİVERSİTESİ

BİLGİSAYAR VE BİLİŞİM BİLİMLERİ FAKÜLTESİ

BİLGİSAYAR MÜHENDİSLİĞİ BÖLÜMÜ

PROGRAMLAMA DİLLERİNİN PRENSİPLERİ ÖDEV RAPORU

1.ÖDEV

B211210569 – Muhammed İyd

1.Öğretim B grup

---

## Özet

Bu ödev, Eclipse ortamında Java dilinde yazılacak bir konsol uygulaması geliştirilmesini istemektedir. Programın temel işlevi, kullanıcıdan bir GitHub deposunun URL'sini almak, bu depoyu klonlamak ve içinde bulunan .java uzantılı dosyaları analiz etmektir. Özellikle, sadece sınıfları içeren Java dosyaları analize tabi tutulacaktır. Analiz edilecek öğeler şunlardır:

- Javadoc olarak yazılan yorum satırlarının sayısı,
- Diğer yorum satırlarının sayısı,
- Boş ve yorum satırları hariç gerçek kod satırlarının sayısı,
- LOC (Line of Code) yani dosyadaki toplam satır sayısı,
- Fonksiyon sayısı (sınıfın içindeki toplam fonksiyon sayısı),
- Yorum Sapma Yüzdesi (yazılması gereken yorum satırlarının yüzdelik olarak ne kadar sapma gösterdiği).

Bu analizler sonucunda, her sınıf için ayrı ayrı belirtilen metrikler ekrana basılacaktır. Ayrıca, ödevde Javadoc yorumları, diğer yorumlar, kod satırları ve fonksiyon sayısına göre bir yorum sapma yüzdesi hesaplaması yapılması gerekmektedir. Hesaplama formülleri ödev dokümanında verilmiştir.

Analiz sonuçları her sınıf için ayrı ayrı ekrana basılacak.

## 1.ÖĞRENDİKLERİMİZ:

Benim için Java'da yazmak harika bir ilk deneyimdi.

Git Komutlarının Kullanımı: Java'dan dış komutları çalıştırarak git deposu klonlama.

Dosya Sistemi İşlemleri: Java ile dosya ve klasör işlemleri gerçekleştirme,

özellikle klasörleri rekürsif olarak tarayarak işlem yapma.

Düzenli İfadeler: Java'da düzenli ifadeler kullanarak metin işleme ve analiz etme.

---

## 2. ZORLANDIĞIMIZ KISIMLAR:

REGEX'i daha önce kullanmadım. Bunu ilk kez deneyimlemek eğlenceli olsa da başa çıkmak çok zordu.

Bazı Komutları, Dış sistem komutlarını Java üzerinden çalıştırırken karşılaşılan güvenlik ve izin sorunları.

Hata Yönetimi, Analiz sırasında karşılaşılan hataların yakalanması ve kullanıcıya uygun geri bildirimin sağlanması.

Projeyi gerçekleştirirken, Java'nın dosya işlemleri, dış sistem komutları çalıştırma yetenekleri ve düzenli ifadeler ile derinlemesine çalışma fırsatı buldum.

Javadoc ta /\* ile başladığı için çoklu yorum da / ile başladığı için ilk başta her bir javadoc'u bir çoklu satır yorumu sayıyordu bu sorunu çözmekte biraz zorlandım.

fonksiyon gövdesinin kapama süslü parantezini "}" kontrol etmek biraz zorlayıcıydı çünkü kontrol ifadeleri ve tekrar ifadeleri de } içerebilirdi onun için ilk başta fonksiyon gövdesinin bitişini yanlış alıyordu.

Ayrıca, gerçek dünya senaryolarında karşılaşılabilecek performans ve hata yönetimi gibi zorluklarla başa çıkma konusunda önemli deneyimler edindik. Bu projede elde edilen tecrübeler, yazılım geliştirme becerilerimizi geniş bir yelpazede geliştirmemize olanak tanıdı.

Son olarak ödev çok eğlenceli bir ödev olup özellikle de düzenli ifadeler konusunda çok şey kattığını belirtmek istiyorum. Bütün dersi veren hocalarıma teşekkür ederim...

---

### 3. Yaptıklarımız:

1. **Main** Sınıfı: bu sınıfı, kullanıcıdan bir GitHub deposunun URL'sini alır ve bu URL'yi Depolsleyicisi sınıfına gönderir. Depolsleyicisi sınıfı, verilen URL'deki depoyu klonlar. Klonlama işleminden sonra, JavaDosyaAnalizci sınıfı, klonlanan depo içindeki Java dosyalarını analiz eder. Bu analiz sırasında, dosyalar içindeki yorum satırları, kod satırları, toplam satır sayısı ve fonksiyon sayısı gibi bilgiler incelenir. Son olarak, analiz tamamlandıktan sonra Depolsleyicisi sınıfı ile klonlanan depo silinir.
  2. **JavaDosyaAnalizci** Sınıfı: bu sınıf, verilen bir klasördeki Java dosyalarını rekürsif olarak bulur ve analiz eder. Her bir Java dosyası için, javadoc yorum satırları, diğer yorum satırları, kod satırları, fonksiyon sayısı gibi bilgileri hesaplar. Ayrıca, her fonksiyona düşen ortalama yorum satırı sayısı ile kod satırı sayısının oranını hesaplayarak, kodun yorumlanma derecesine dair bir sapma yüzdesi sunar.
  3. **Depolsleyicisi** Sınıfı: bu sınıfı, kullanıcının girdiği GitHub deposu URL'sini alarak, bu depoyu yerel bir klasöre klonlar. Klonlama işlemi tamamlandıktan sonra, bu sınıf aynı zamanda klonlanan klasörü ve içindekileri silme işlevselliğine de sahiptir. Bu işlemler için Java'nın Runtime ve Files kütüphaneleri kullanılmaktadır.
-

