

Programlama Dillerinin Prensipleri Ödev Raporu

Öğrendiklerim: Ödevde önerildiği gibi regex kullanmaya karar verdim. Ancak daha önce regex ile herhangi bir deneyimim yoktu. Bu yüzden mantığını anlamak ve kavramak adına regex nedir, nasıl kullanılır üzerine çalıştım. Java dili ile pratik yapmış bulundum.

Ödevde Yaptıklarım: Programı ilk önce kağıt üzerinde planlayarak tasarladım. Tabii ki geliştirme aşamasında her şey tasarladığım gibi gitmedi ancak yaptığım tasarı yolumu büyük oranda belirlemişti. Tasarıyı oluştururken regex'e dair hiçbir fikrim yoktu bu yüzden o sırada regex nedir bunu araştırdım ve ufak örneklerle regex kullanmayı pekiştirdim. Daha sonra büyük dosyalarda detaylı deneyler şeklinde nasıl kullanılır çözmeye çalıştım. Tasarıyı tamamladıktan sonra nihayet proje dosyamı oluşturdum ve öncelikle regex komutlarını tutan bir sınıf yazdım. İçerisine dosyada aramam gereken anahtar karakter dizilerini bulabilen regex ifadelerini yazdım. Dosyadan okunacak sınıf ve fonksiyonları tutması için 2 ayrı sınıf oluşturdum. Classes sınıfı içerisinde fonksiyon nesnelerini tutacak bir arraylist'e sahip. Functions sınıfı ise içerisinde 3 ayrı yorum türünü(Tekli satır, Çoklu satır, Javadoc) tutacak 3 ayrı arraylist'e sahip. Ardından dosya işlemleri içinde bir sınıf oluşturdum. Dosyanın adını komut argümanı ile okuyup sisteme aktarmasını sağladım. Ardından dosya okuma işlemlerini 2 ayrı fonksiyon ile gerçekleştirmek için zemin hazırladım. İlk fonksiyon dosyada ki sınıf ve fonksiyonları bularak sisteme aktarıyor. İkinci fonksiyon ise dosyadaki yorumları bularak ait oldukları fonksiyonlara atanıyor. Yani örnek olarak bir fonksiyona ait javadoc, dosya da ait olduğu fonksiyonun sistemdeki referansının içerisindeki javadoc arraylist'ine atanıyor. Verilerin saklanması bağlı listeye oldukça benzer diyebilirim. Mantığını en iyi bu şekilde anlatabilirim. Yorumların hangi fonksiyona ait olduğunu süslü parantezleri filtreleyerek ve bir sayaç ile kontrol ederek buldum. Bir açılış parantezi "{" parantez sayacını bir artırıyor. Bir kapanış parantezi "}" ise parantez sayacı sıfır ise fonksiyon sayacını bir artırarak sonraki fonksiyona geçilmesini sağlıyor. Eğer ki sıfır değil ise demek ki fonksiyonun bitmesini sağlayan kapanış parantezi değil demek ve sayacı bir azaltıp es geçiyor. Okuma işlemlerinden sonra ekrana istenen şekilde verilerin yazdırılmasını sağlayan fonksiyonu yazdım. Ardından yorumları ait oldukları fonksiyon ve class bilgileri ile yorumun türüne göre dosyalara yazdıran fonksiyonu yazdım. Hataları düzelttim ve son dokunuşları yaptım. Ödev son haline gelmiş sayılırdı. Daha sonra ödevi istenen son şekline getirerek ödevi bitirdim.

Eksik Bıraktığım Yerler: Bulduğum yanlışlarımı ve hatalarımı düzelttim. Yaptığım her şeyi birçok kez gözden geçirdim, herhangi bir eksik ya da yanlış bulamadım.

Zorlandığım yerler: Regex gerçekten kafa karıştırıcı, çok kafa karıştırıcı bir yapı. Ancak bir o kadarda kullanışlı ve yararlı. Ödevde en zorlandığım kısım bu olabilir. En ufak hataya yer vermiyor. Program tamamen regex sayesinde çalışıyor ve bir hata ile her şey alt üst olabiliyor. En fazla vakti regex'e harcadım diyebilirim.

Umut Can Çağatay – G201210029

umutcan.cagatay@ogr.sakarya.edu.tr