

# Real-Time Grammar-Based Syntax Highlighter with GUI Proje Raporu

**Ders:** Programlama Dilleri

**Proje Adı:** Real-Time Grammar-Based Syntax Highlighter with GUI

**Hazırlayan:** Sude Naz Doğdu

## 1. Dil ve Gramer Seçimi

Projede Python programlama dili kullanılmıştır. Python, metin işleme konusunda sunduğu güçlü olanaklar sayesinde sözcüksel ve sözdizimsel analiz işlemleri için oldukça uygundur. Arayüz geliştirmek amacıyla, Python'un standart kütüphanesi olan tkinter tercih edilmiştir. Bu kütüphane, dışa bağımlı olmadan kolayca GUI uygulamaları geliştirmeye imkân tanır.

Ancak, projede sözdizimi vurgulama işlevleri için herhangi bir hazır kütüphane (örneğin Pygments) kullanılmamıştır. Tüm vurgulama işlemleri, tkinter.Text arayüzü üzerinden tamamen el ile geliştirilmiş, token pozisyonları esas alınarak yapılmıştır.

Tanımlanan mini dil grameri: program

```
→ statement_list statement →  
assignment | if_statement assignment  
→ ID = expr ; if_statement → if ( expr ) {  
statement_list } expr → term ((+|-)  
term)* term → ID | NUMBER
```

## 2. Sözdizimsel Analiz Süreci

Sözdizimsel analiz işlemi, gramer kurallarına göre yazılmış bir Parser sınıfı ile gerçekleştirilmiştir. Parser, **Recursive Descent Parsing** yöntemi kullanılarak geliştirilmiştir. Her bir gramer kuralı için ayrı Python fonksiyonları tanımlanmış ve analiz işlemi bu fonksiyonlar üzerinden yürütülmüştür.

Önce kullanıcıdan alınan kod tokenize() fonksiyonu yardımıyla sözcüklere ayrılır. Ardından bu token dizisi parser tarafından sırayla işlenerek yazılan kodun gramer açısından geçerli olup olmadığı denetlenir. Geçerli ifade örneği:

```
if (x == 5) {  
    y = x + 1;  
}
```

### 3. Sözcüksel Analiz

Sözcüksel analiz işlemi, Python'un re modülü (düzenli ifadeler) kullanılarak gerçekleştirilmiştir. Her token türü için ayrı bir regex (düzenli ifade) tanımı yapılmış ve bu tanımlar tokenize() fonksiyonu içerisinde birleştirilmiştir.

Tanımlanan token türleri şunlardır:

IF, ELSE → Anahtar kelimeler

ID → Tanımlayıcılar

NUMBER → Sayılar

EQ, ASSIGN, PLUS → Operatörler

LPAREN, RPAREN, LBRACE, RBRACE, SEMI → Semboller

### 4. Ayrıştırma Metodolojisi

Top-Down Parsing yaklaşımı benimsenmiştir. Her bir gramer kuralı, Python'da ayrı bir fonksiyon olarak tanımlanmıştır. Parser, bu fonksiyonlar aracılığıyla sırasıyla assignment, if\_statement ve expr gibi yapıları kontrol eder.

Eğer analiz edilen token dizisi belirtilen gramer kurallarına uymazsa kullanıcıya anında hata mesajı ile geri bildirim verilir. Parser ayrıca GUI'ye entegre edilmiştir. Kullanıcı, "Kodu Kontrol Et" butonuna bastığında sözdizimsel kontrol gerçekleştirilir ve sonuç arayüzde gösterilir.

### 5. Vurgulama Şeması

Her token tipi için farklı bir renk şeması uygulanmıştır. Bu renkler, tkinter.Text bileşeni üzerinden tag\_add() ve tag\_config() metodları ile tanımlanmıştır.

Kullanılan renk şeması:

- **IF, ELSE:** Mavi
- **ID:** Siyah
- **NUMBER:** Yeşil • **EQ, ASSIGN:** Turuncu
- **PLUS:** Kırmızı
- **LPAREN, RPAREN, LBRACE, RBRACE:** Gri • **SEMI:** Mor

Kullanıcı metin kutusuna her tuş bastığında highlight() fonksiyonu çağrılır. Bu fonksiyon, yazılan metni anlık olarak analiz eder ve vurgulamayı günceller. Bu sayede gerçek zamanlı sözdizimi vurgulaması sağlanır.

## 6. GUI Uygulaması

Projenin kullanıcı arayüzü tkinter ile geliştirilmiştir. Arayüzde kullanıcı dostu bir tasarım hedeflenmiştir. Temel bileşenler:

- Geniş bir **Text** kutusu: Kod yazımı için alan
- Gerçek zamanlı sözdizimi vurgulaması
- “**Kodu Kontrol Et**” butonu: Parser’ı tetikler
- **Durum etiketi (Label)**: Kodun geçerli olup olmadığını gösterir

Bu yapı sayesinde kullanıcı hem yazdığı kodun sözdizimini anında gözlemleyebilmekte hem de dilerse buton aracılığıyla geçerlilik kontrolü yapabilmektedir.

## 7. Sonuç ve Değerlendirme

Bu proje, küçük bir programlama diline ait kodların hem sözcüksel hem sözdizimsel analizini yapan, aynı zamanda gerçek zamanlı sözdizimi vurgulaması sunan bir GUI uygulamasıdır. Hazır kütüphane kullanmadan geliştirilen bu yapı, dil tanıma, analiz ve görselleştirme açısından önemli bir öğrenme deneyimi sunmuştur. Ayrıca, kullanıcının etkileşimli bir şekilde analiz sürecine dâhil olması sağlanmıştır.

## 8. Ekler

Video Linki: <https://youtu.be/wu9H81JOPqw>