

**Makine Öğrenmesi ve Mühendislik Uygulamaları Final Raporu**

**Hazırlayan : Kadir Yıldız -** **2430151002**

**Sorumlu Öğretim Elemanı :** **Dr. Öğr. Üyesi**Mehmet BİLEN

**Rapor Teslim Tarihi :**

### ****Projenin Amacı ve Kapsamı****

Bu proje, gerçek dünyadan veri toplama, düzenleme ve organize etme süreçlerini uygulamalı olarak göstermek amacıyla gerçekleştirilmiştir.

Trendyol üzerinde "erkek t-shirt" kategorisinde yer alan ürünlere ilişkin bilgileri içermektedir. Bunun yanı sıra Trendyol’un herhangi bir kategörideki ürünlerinin verisini toplayacak şekilde kod yazılmuştır. Veriler arasında ürünün isimleri, fiyatı, favoriye alınma sayısı , değerlendirme sayıları ve ürün derecelendirme puanları yer almaktadır.

### ****Veri Toplama Süreci****

Veri toplama işleminde, C++ programlama dili ve CURL kütüphanesi kullanılmıştır. Web scraping yapılarak, ilgili sayfalardan ürün verileri JSON formatında düzenlenmiş bir şekilde elde edilmiştir. Trendyol’un curl ile çekilen ham HTML formatında gerekli tagları belirleyerek gerekli veriler çekilmiştir ve json formatında yazılmıştır.

#### Kullanılan Araçlar

1. **C++ , CURL Kütüphanesi , G++ ve Vim:** Web sayfasından HTML içeriği çekmek için kullanılmıştır. C++ dilinde yazılan kod G++ ile derlenmiştir.

Derleme methodu : g++ -Wall -std=c++17 -Iinclude -c -o bin/main.o src/main.cpp

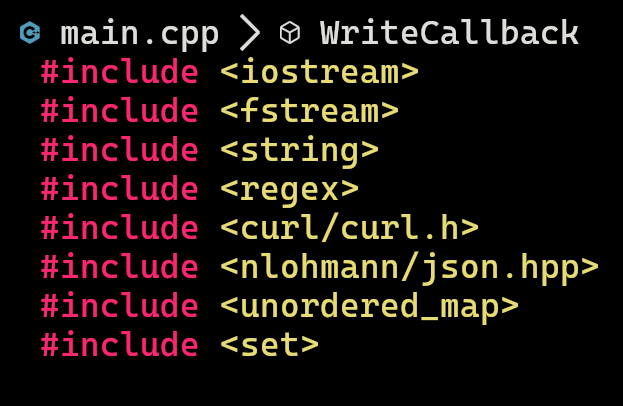
g++ -Wall -std=c++17 -o main bin/main.o -lcurl

Vim editörünü kullanarak yazılmıştır.

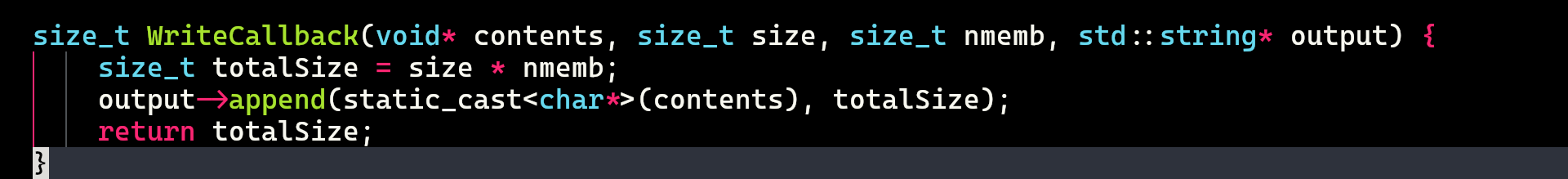
1. **Regex (Düzenli İfadeler):** HTML içerisindeki ilgili verileri çıkarmak için yazılmış düzenli ifadeler. Bu ifadeler aslında tek satırda’da yapılabilirdi. Ancak bazı verileri çekerken yanlış veya eksik veri çekmesinden dolayı farklı regexleri eklemeyi uygun buldum.
2. **JSON Formatı:** Tercihen Json formatında kayıt edilmiştir.

#### Veri Toplama Adımları

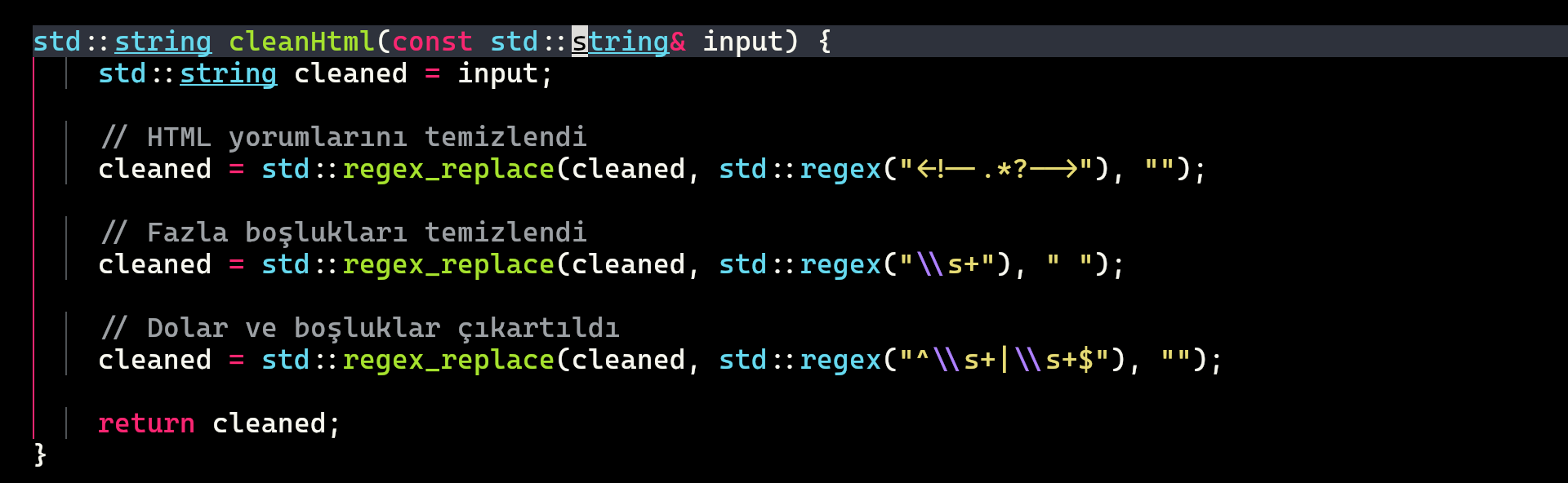
1. Gerekli kütüphaner eklendi. Regex gerekli taglerin çekilmesi, curl ham html’in çekilmesi , json.hpp ise json formatına düzenlemek için kolay bir kütüphane, key – value değerinin tutulması için unordered\_map ve json verilerinin tekrar etmemesi amacıyla set kullanılmıştır.



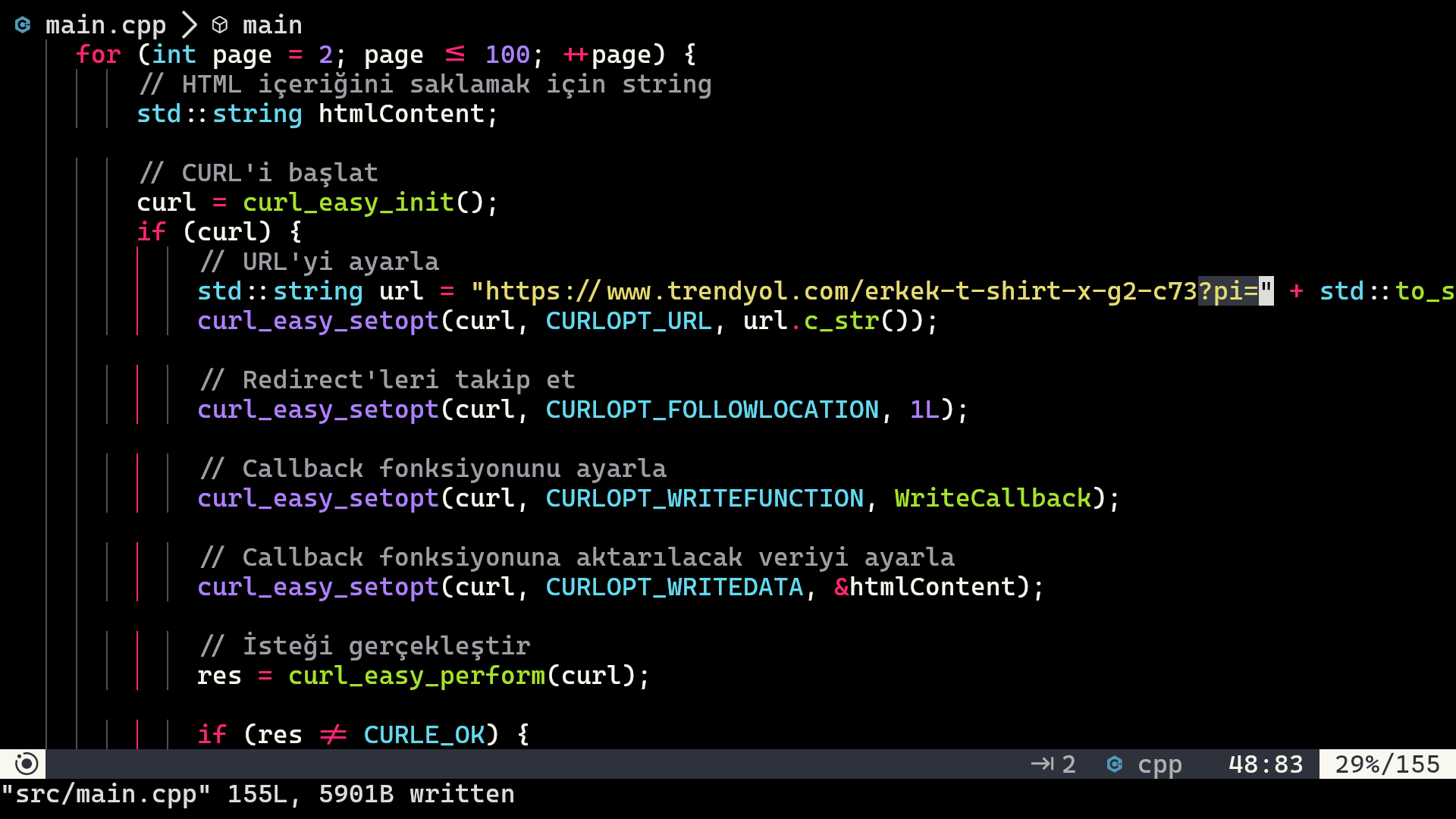
1. Curl ile çekilen verinin yazılması için fonksiyon eklendi.



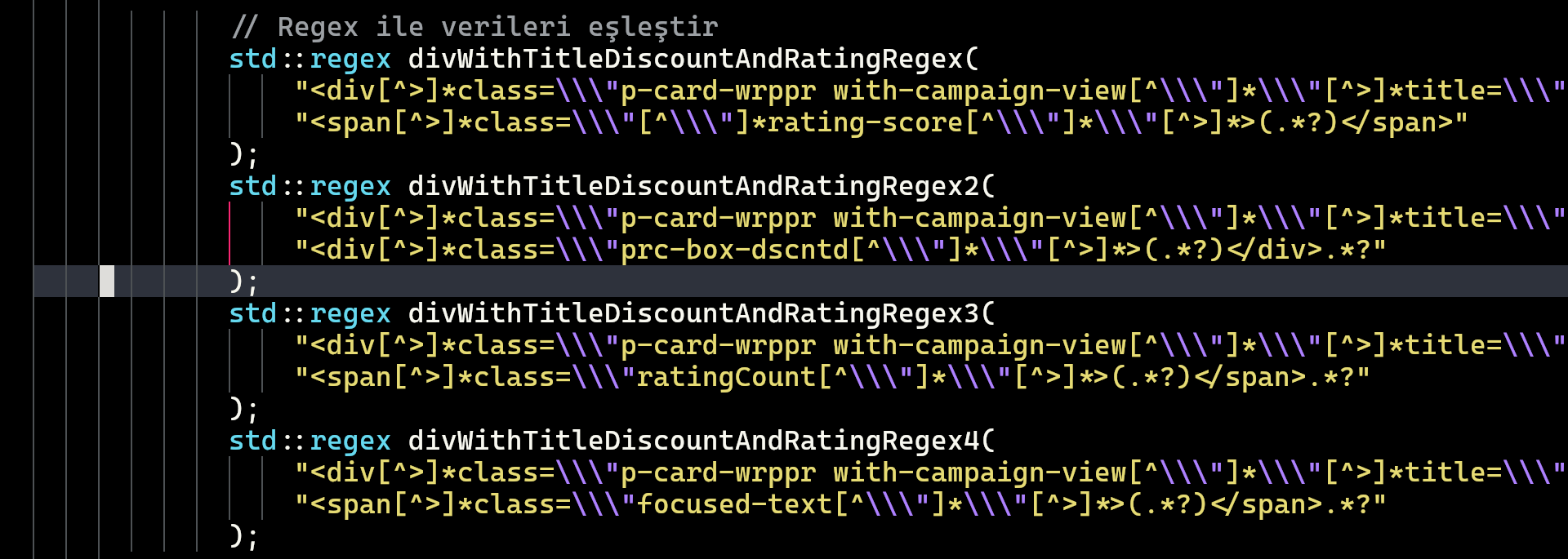
1. CleanHtml adında bir fonksiyon oluşturuldu. İşlevi verilerden gelen istemediğim karakteri silmektir. Dolar işareti ve gereksiz → gibi işaretler barındırmaktaydı.



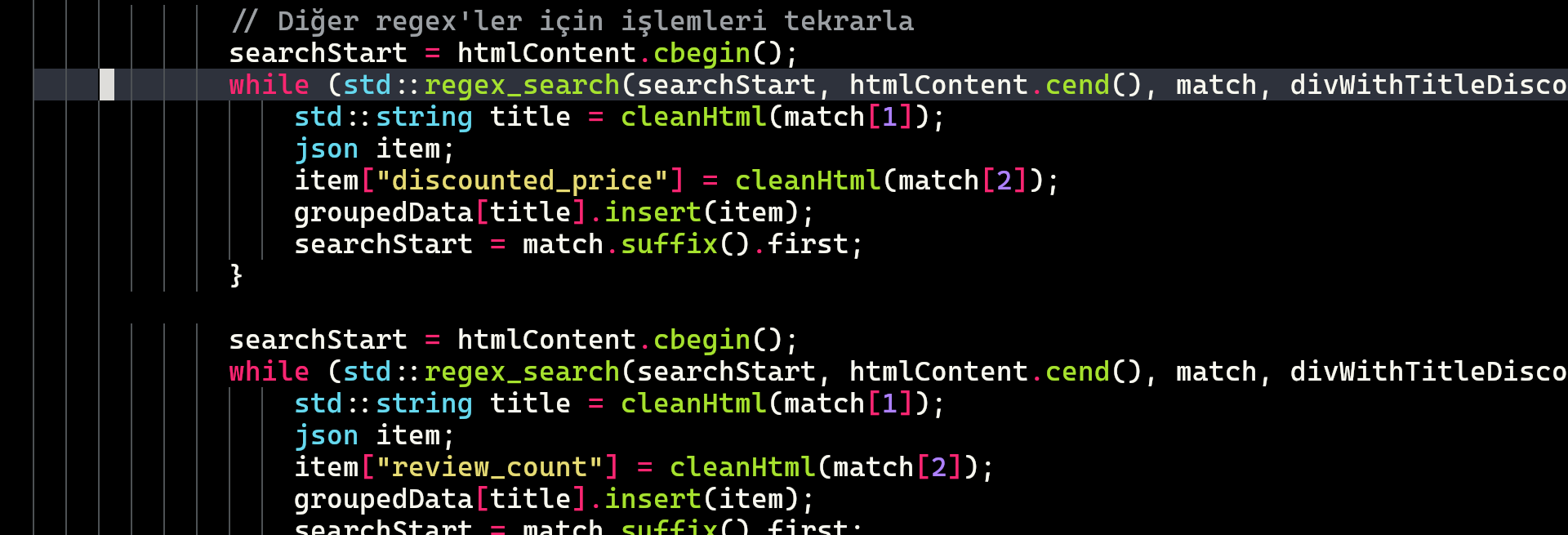
1. Trendyol sitesinde pi=2 ile anlatılmak istenen şey aslında 2. sayfaya indiğini ve farklı ürünler gösterdiğini anlatmaktadır. Bu sebepten ötürü başka sayfalardaki verileri çekmek amacıyla bir for oluşturdum ve page adındaki veriyi 2’den 100’e kadar artacak şekilde planladım. Böylece 100 sayfaya kadar tarayacaktır. 100 sayısı keyfi oluşturulmuştur.



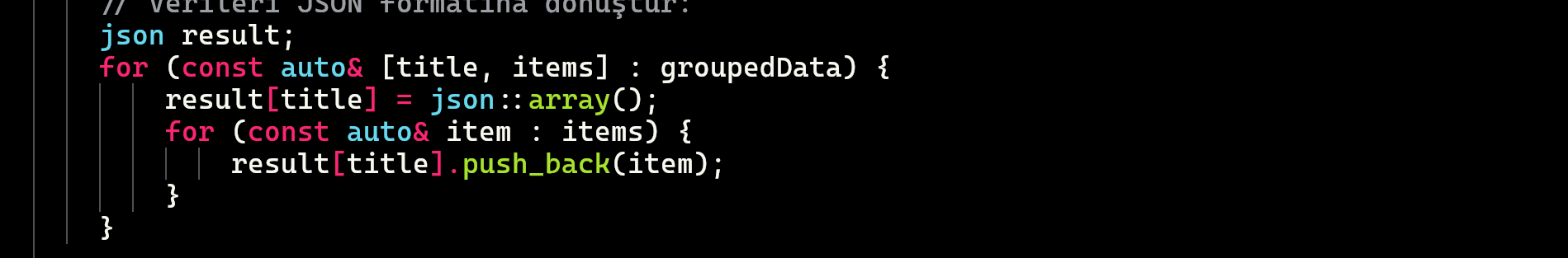
1. İstenen tagleri çekmek için Regex kullanıldı. Her bir Title için Fiyat, ne kadar kişinin favorilediği gibi veriler çekildi.



1. Çekilen veriler Json formatında düzenlendi ve Regex değerinin başlangıç değeri tekrar setlenmiş oldu.4 veri çekimi için tekrarlanmıştır. Json formatında kaydedilip Suffix() ile başlangıç değeri set edilmiştir. CleanHtml methodu ile verilerin istenmeyen karakterleri silinmiştir.



1. Veriler Json şeklinde birleştirilildi.



1. Veriler Output.json şeklinde yazdırıldı. 4 değeri daha okunabilir olması için boşluklandırma amacıyla kullanılmıştır.

### 

### ****Elde Edilen Veri Seti****

#### Verinin Genel Yapısı

Veri seti, JSON formatında organize edilmiştir ve aşağıdaki özellikleri içermektedir:

* **Title (Başlık):** Ürünün adı.
* **Discounted Price:** İndirimli fiyat.
* **Rating:** Ürünün puanı (5 üzerinden).
* **Review Count:** Ürün hakkında yapılan değerlendirme sayısı.
* **Favorites:** Ürünün favoriye alınma sayısı.

#### Örnek Veri

"Çiçek Baskılı Unisex Oversize Tshirt": [

{

"discounted\_price": "122,49 TL"

},

{

"favorites": "32.4B kişi"

},

{

"rating": "4.1"

},

{

"review\_count": "(138)"

}

]

### ****Sonuç****

Bu proje, gerçek dünyadan veri toplama ve temizleme sürecini uygulamalı olarak göstermektedir. Toplanan veri seti, çeşitli makine öğrenmesi uygulamaları için potansiyel olarak kullanılabilir.

Raporda belirtilen tüm adımlar, veri bilimi ve mühendislik uygulamaları açısından önemli bir deneyim sunmaktadır.

Projenin Github linki:

<https://github.com/kadiryildiz283/Makine-renmesi-ve-M-hendislik-Uygulamalar-Final-Raporu>

### ****Projenin Amacı ve Kapsamı****

Bu proje, gerçek dünyadan veri toplama, temizleme ve organize etme süreçlerini uygulamalı olarak göstermek amacıyla gerçekleştirilmiştir.

Seçilen veri seti, Trendyol üzerinde "erkek t-shirt" kategorisinde yer alan ürünlere ilişkin bilgileri içermektedir. Veriler arasında ürün başlıkları, indirimli fiyatlar, favori sayıları, değerlendirme sayıları ve ürün derecelendirme puanları yer almaktadır.

Amaç, bu verileri temizleyerek makine öğrenmesi projelerinde kullanılabilir hale getirmektir.

### ****Veri Toplama Süreci****

Veri toplama işleminde, C++ programlama dili ve CURL kütüphanesi kullanılmıştır. Web scraping yapılarak, ilgili sayfalardan ürün verileri JSON formatında düzenlenmiş bir şekilde elde edilmiştir.

#### Kullanılan Araçlar

1. **C++ CURL Kütüphanesi:** Web sayfasından HTML içeriği çekmek için kullanılmıştır.
2. **Regex (Düzenli İfadeler):** HTML içerisindeki ilgili verileri çıkarmak için yazılmış düzenli ifadeler.
3. **JSON Formatı:** Toplanan verilerin daha sonradan kolay işlenebilmesi için organize edilmesinde kullanılmıştır.

#### Veri Toplama Adımları

1. Trendyol üzerindeki ilgili URL'ye HTTP GET isteği gönderilerek HTML içeriği çekildi.
2. HTML içerisindeki ürün kartları Regex kullanılarak filtrelendi.
3. Toplanan verilerde tekrarlayan kayıtlar temizlendi.
4. Temizlenmiş veriler JSON formatına dönüştürüldü ve kaydedildi.

### ****Karşılaşılan Zorluklar ve Çözümöneri****

1. **Zorluk:** HTML yapısındaki karmaşıklıklar ve etiketlerin sabit olmaması Regex ifadelerinin çalışmasını zorlaştırdı.
   * **Çözüm:** Düzenli ifadeler, çoklu denemelerle daha esnek bir yapıya dönüştürüldü.
2. **Zorluk:** Aynı ürün bilgilerinin birden fazla kez toplanması.
   * **Çözüm:** Veri, ürün başlığı baz alınarak gruplanmış ve tekrarlar temizlenmiştir.

### ****Elde Edilen Veri Seti****

#### Verinin Genel Yapısı

Veri seti, JSON formatında organize edilmiştir ve aşağıdaki özellikleri içermektedir:

* **Title (Başlık):** Ürünün adı.
* **Discounted Price:** İndirimli fiyat.
* **Rating:** Ürünün puanı (5 üzerinden).
* **Review Count:** Ürün hakkında yapılan değerlendirme sayısı.
* **Favorites:** Ürünün favori sayısı.

#### Örnek Veri

[

{

"title": "Basic Siyah T-shirt",

"data": [

{"discounted\_price": "849 TL"},

{"rating": "4.4"},

{"review\_count": "780"},

{"favorites": "10.5B"}

]

},

{

"title": "Basic Beyaz T-shirt",

"data": [

{"discounted\_price": "799 TL"},

{"rating": "4.2"},

{"review\_count": "510"},

{"favorites": "8.2B"}

]

}

]

### ****Veri Seti Analizi ve Gözlemler****

* Ortalama indirimli fiyat: **824 TL**
* En yüksek puan alan ürün: **Basic Siyah T-shirt (4.4)**
* En fazla favori sayısı: **10.5B (Basic Siyah T-shirt)**

### ****Potansiyel Makine Öğrenmesi Uygulamaları****

1. **Tahmin Problemleri:**
   * Ürünün fiyatını tahmin etmek için bir regresyon modeli oluşturulabilir.
   * Favori sayısının ürün değerlendirme puanlarına etkisi analiz edilebilir.
2. **Öneri Sistemleri:**
   * Kullanıcının daha önce favorilediği ürünlere göre yeni ürün önerileri sunulabilir.
3. **Popülerlik Analizi:**
   * Ürünlerin hangi faktörlerle daha popüler hale geldiği çıkarılabilir (fiyat, değerlendirme sayısı vb.).

### ****Sonuç****

Bu proje, gerçek dünyadan veri toplama ve temizleme sürecini uygulamalı olarak göstermektedir. Toplanan veri seti, çeşitli makine öğrenmesi uygulamaları için potansiyel olarak kullanılabilir.

Raporda belirtilen tüm adımlar, veri bilimi ve mühendislik uygulamaları açısından önemli bir deneyim sunmaktadır.