

Grup 116 Yaz Lab. Projesi Raporu

Emircan Kural

Ahmet Muhammed Şahin

221307071

221307072

Dijital çağın gelişimiyle birlikte internet üzerindeki veri miktarı her geçen gün artmakta ve çeşitli sektörlerden kullanıcılar, internet üzerindeki bu zengin veri kaynaklarına kolayca erişebilme ihtiyacı duymaktadır. Özellikle yemek tarifleri gibi gündelik yaşamda sıklıkla başvuru bilgileri, kullanıcılar tarafından çeşitli web siteleri üzerinden elde edilmektedir. Ancak, bu bilgileri manuel olarak incelemek hem zaman alıcı hem de verimsiz olabilmektedir. Bu sorunu aşmak amacıyla, Python programlama dili kullanılarak otomatik veri çekme işlemlerini gerçekleştirebileceğimiz bir proje geliştirdik. Bu projede, yemek tarifleri içeren web sitelerinden verileri otomatik olarak çekmek amacıyla **Selenium** ve **Beautiful Soup** kütüphanelerini kullanarak web kazıma (web scraping) tekniklerini uyguladık.

Selenium, özellikle dinamik içeriklere sahip web sitelerinde kullanıcı simülasyonu yaparak etkileşime girme imkânı sağlarken; BeautifulSoup ise HTML ve XML yapılarının daha verimli bir şekilde işlenmesine olanak tanır. Bu iki güçlü kütüphaneyi kullanarak, hedef sitelerin yapısına göre özel olarak hazırlanan kod parçaları sayesinde, ilgili sayfalardan yemek tariflerine ait başlık, içerik, malzemeler, hazırlık aşamaları ve diğer bilgileri başarılı bir şekilde çekebildik. Çekilen verilerin düzenlenmesi, analizi ve saklanması ise proje kapsamındaki diğer adımları oluşturmuştur. Bu girişim, yemek tarifi gibi sıkça aranan bilgilere hızlı ve sistematik bir şekilde erişmeyi sağlayarak, kullanıcıların bilgiye ulaşma süreçlerini daha verimli bir hale getirmeyi amaçlamaktadır.

Veri Kaynakları ve Kod Yapısı

Bu projede yemek tariflerini kazımak için "Arda'nın Mutfağı" ve "Nefis Yemek Tarifleri" adlı popüler yemek tarifleri sitelerinden faydalandık. Her iki siteyi kullanırken sitelerin HTML yapılarındaki farklılıklardan dolayı iki ayrı kod geliştirdik. Bu sayede her sitenin özgün yapısına uygun, verimli ve doğru veri çekme işlemleri gerçekleştirdik.

İlk olarak, **Arda'nın Mutfağı** sitesinin yapısını analiz ederek bu site için özel bir kod yapısı oluşturduk. Bu kod yapısında, sayfanın HTML yapısı detaylı bir şekilde incelendi ve tarif başlıkları, içerikler, malzemeler, hazırlık adımları gibi öğelerin bulunduğu etiketler tespit edilerek, veri çekme işlemleri bu etiketlere göre yapılandırıldı. BeautifulSoup ve Selenium kütüphaneleri yardımıyla, sayfa üzerinde bulunan dinamik içeriklere erişim sağlanarak kullanıcı etkileşimi simüle edildi ve gerekli bilgiler otomatik olarak elde edildi. Bu işlem, Arda'nın Mutfağı sitesinde yer alan tarifleri düzenli ve sistematik bir şekilde toplamak için özelleştirilmiş bir veri çekme çözümü sunmaktadır.

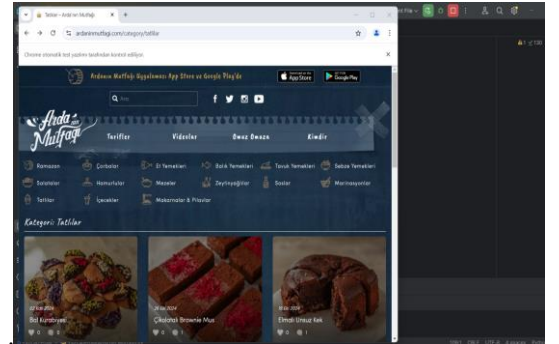
Kodun Amacı

Bu kod, bir yemek tarifleri sitesinden içecek tariflerini çekmek, her bir tarifi adını ve malzemelerini almak, ve bu bilgileri XML formatında bir dosyaya kaydetmek amacıyla geliştirilmiştir. Kullanıcı, kaç sayfa tarayacağını belirleyerek tariflerin otomatik olarak toplanmasını sağlayabilir.

Kullanılan Kütüphaneler ve Modüller

Kod, dört ana modül kullanılmaktadır:

- **Selenium:** Web tarayıcısını kontrol etmek ve web sayfalarını dinamik olarak yüklemek için kullanılır. webdriver modülü ile Google Chrome tarayıcısı üzerinden veri çekimi gerçekleştirilir.
- **BeautifulSoup (bs4):** Web sayfasının HTML içeriğini ayrıştırmak için kullanılır. Bu sayede, belirli HTML elementleri seçilebilir ve içeriği alınabilir.
- **time:** Web sayfalarını yüklerken bekleme süreleri eklemek için kullanılır. Bekleme süreleri, sayfanın tamamen yüklenmesini sağlamak için gereklidir.
- **xml.etree.ElementTree:** XML dosyası oluşturmak ve yazmak için kullanılan bir modüldür. Tarifler, XML yapısında organize edilir

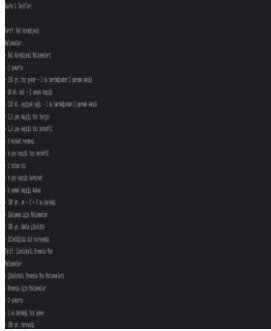


Sonuç ve Öneriler

Bu kod, web üzerindeki tariflerin otomatik olarak toplanmasını sağlayarak XML formatında düzenli bir veri yapısı oluşturur. Web scraping işlemi başarılı şekilde yapılmış olsa da, bu tür işlemler için sitenin kullanım şartları ve yasalara uygunluk göz önünde bulundurulmalıdır.

Geliştirme Önerileri

1. **Dinamik Bekleme Süresi:** `WebDriverWait` kullanılarak her sayfa yüklemesi için sabit bir süre yerine dinamik bir bekleme yapılabilir.
2. **Hata Yönetimi:** Sayfa yüklenmediğinde veya bağlantı kopmalarında kodun devam edebilmesi için hata yönetimi eklenebilir.
3. **Veritabanı Entegrasyonu:** XML yerine SQL veya NoSQL bir veritabanı kullanılarak verilerin daha hızlı erişilebilir ve sorgulanabilir olması sağlanabilir.



Yemek.com Kodunun Raporu

popüler yemek tarifleri sitesi “Yemek.com” üzerinden tarif bilgilerini çekerek XML formatında kaydetmek için Python programlama dilinde **Selenium** ve **Beautiful Soup** kütüphanelerini kullandık. Aşağıda, kodun işlevleri ve her bir adımda gerçekleştirilen işlemler detaylandırılmıştır.

Kullanılan Kütüphaneler ve Fonksiyonlar

Kodda, tarayıcı otomasyonunu sağlamak için **Selenium**, HTML verilerini ayrıştırmak için **Beautiful Soup**, ve veriyi XML formatında saklamak için **xml.etree.ElementTree** kütüphaneleri kullanılmıştır. Kodun temel fonksiyonları aşağıdaki gibidir:

1. **normalize_text:** Türkçe karakterleri, İngilizce eşdeğerlerine dönüştürerek bir normalizasyon işlemi gerçekleştirir. Bu işlem, tarif başlıklarının URL yapısına uygun hale getirilmesi için gereklidir.
2. **bosluk_ekle:** Sayılar ve metinler arasında boşluk ekleyerek malzeme listelerini daha okunabilir hale getirir.

Kodun İşleyişi

1. Tarayıcıyı Başlatma ve Sayfaya Erişim

İlk olarak, Selenium ile bir Chrome WebDriver nesnesi oluşturulur ve hedef site olan “Yemek.com” üzerindeki tarifler sayfasına erişilir. Dinamik içeriklerin yüklenmesi için belirli bir süre (5 saniye) beklenir.

2. Sayfa Kaydırma

Sayfanın tamamının yüklenmesini sağlamak için Selenium kullanılarak, tarayıcı "END" tuşuna basılarak sayfa aşağı kaydırılır. Bu işlem, belirli bir süre aralıklarla 400 kez tekrar edilerek tüm tariflerin yüklenmesini sağlar.

3. Tarif Başlıklarını Çekme

Sayfanın HTML içeriği BeautifulSoup ile ayrıştırılarak, tarif başlıklarının bulunduğu etiketler bulunur. Her bir tarif başlığı temizlenir, Türkçe karakterler normalize edilir ve uygun bir URL yapısına dönüştürülerek, tarifin detay linki oluşturulur.

4. XML Yapısı Oluşturma

Çekilen her tarif için XML formatında bir yapı oluşturulur. Tarifin başlığı ve detay linki, XML ağacında ilgili alanlara eklenir.

5. Detay Sayfasına Erişim ve Malzeme Listesi Çekme

Her bir tarifin detay sayfasına gidilir ve ilgili sayfanın HTML içeriği alınarak BeautifulSoup ile analiz edilir. Bu sayfadaki malzemeler listesi, sınıf adı üzerinden tespit edilerek XML yapısına eklenir. Ayrıca, “sosu için” gibi ibareler kontrol edilerek, sos malzemeleri farklı bir XML alanına eklenir.

6. Veri Kaydetme ve Tarayıcıyı Kapatma

Kodun sonunda, oluşturulan XML yapısı belirlenen dosya yoluna kaydedilir ve tarayıcı kapatılır. İşlem sırasında hata oluşursa kullanıcıya bilgi verilir, kod manuel olarak durdurulursa toplanan veriler kaydedilir.

Kodun Sağladığı Avantajlar

Bu kod sayesinde, “Yemek.com” gibi sitelerden tarif bilgileri sistematik bir şekilde toplanmakta, XML formatında saklanarak veri analizleri ve farklı uygulamalar için kolayca kullanılabilir hale getirilmektedir.

Referans

- Mitchell, R. (2018). *Web Scraping with Python: Collecting Data from the Modern Web*. 2nd Edition. O'Reilly Media.
- Selenium Documentation. (2024). Selenium.dev
- BeautifulSoup Documentation. (2024). Crummy.com
- Python Documentation - xml.etree.ElementTree. (2024). [Docs.Python.org](https://docs.python.org)

