Documentação de Código

Scraping de Restaurantes no Ifood

Vítor Carvalho Marx Lima

Data Atualização: 15/02/2024

Contents

[Introdução 2](#_Toc158919393)

[Requisitos 2](#_Toc158919394)

[Descrição das Bibliotecas Utilizadas 2](#_Toc158919395)

[Estrutura do arquivo Excel de Configurações 3](#_Toc158919396)

[Fluxo de Código: 3](#_Toc158919397)

[Conclusão 9](#_Toc158919398)

# Introdução

Este documento descreve o código Python projetado para realizar scraping de dados de restaurantes listados no site iFood a partir de uma lista de endereços. O objetivo é coletar algumas informações sobre os estabelecimentos, como nome, avaliação, categoria, distância, taxa de entrega, e salvá-las em um arquivo Excel para análise posterior.

# Requisitos

1 - Para executar este script, é necessário ter Python instalado, junto com as seguintes bibliotecas não instaladas por padrão:

Playwright, bs4, pandas e datetime.

2 – Um arquivo de configurações iniciais denominado ‘enderecos.xlsx’, onde algumas configurações essenciais para o funcionamento do script serão fornecidas pelo usuário. Essas configurações serão detalhadas posteriormente.

# Descrição das Bibliotecas Utilizadas

Playwright: Automatiza o navegador para interações web.

BeautifulSoup (bs4): Analisa o HTML retornado e facilita a extração de dados.

pandas: Utilizado para criar e manipular DataFrames, facilitando a organização dos dados coletados.

datetime: Fornece funções para manipular datas e horários de maneira flexível.

time: Utilizado para pausas durante a navegação, evitando bloqueios por requisições rápidas demais e evitando possíveis erros de carregamento no script.

os: Auxilia na manipulação de caminhos e arquivos do sistema operacional.

# Estrutura do arquivo Excel de Configurações

O script de utiliza um arquivo Excel (enderecos.xlsx) para determinar os endereços dos quais os dados dos restaurantes serão coletados. Este arquivo contém quatro colunas essenciais, descritas abaixo:

**1 - Endereço:** Contém os endereços completos dos locais a serem pesquisados. Cada linha representa um local específico do qual os dados dos restaurantes serão coletados. Os endereços devem estar completos e formatados corretamente para garantir a precisão da busca no site.

**2 - Número:** Refere-se ao número do edifício ou estrutura dentro do endereço especificado. Se o endereço não possuir um número específico, deve-se utilizar "s/n" (sem número) nesta coluna. Este detalhe é crucial para a precisão na localização dos restaurantes próximos ao endereço indicado.

**3 - Número de Páginas:** Indica quantas páginas de resultados devem ser coletadas para cada endereço. Este valor pode ser um número inteiro (1, 2, 3, etc.) para especificar um limite de páginas ou a palavra "todas" para indicar que todas as páginas disponíveis devem ser coletadas. Essa flexibilidade permite um controle mais fino sobre a quantidade de dados coletados, otimizando o tempo de execução do script conforme necessário.

**4 - Rodar:** Esta coluna determina se o endereço especificado na linha correspondente será incluído ou excluído da coleta de dados. Ela aceita dois valores possíveis: "S" (Sim) e "N" (Não). "S" indica que o script deve coletar dados do endereço especificado, enquanto "N" indica que o endereço deve ser ignorado. Isso permite um controle seletivo sobre quais endereços devem ser processados, facilitando a coleta de dados de locais específicos sem a necessidade de deletar endereços do arquivo Excel.

# Fluxo de Código:

O script foi projetado para automatizar a coleta de informações sobre restaurantes a partir do site iFood, utilizando um arquivo Excel com endereços específicos como ponto de partida. O processo inicia com a leitura deste arquivo, onde o script identifica quais endereços devem ser coletados com base na coluna "Rodar", prosseguindo apenas com aqueles marcados com "S". Em seguida, o script acessa o site do iFood, preenche os campos de endereço e número e confirma a localização para navegar até a seção de restaurantes. Uma vez na página dos restaurantes, ele executa a ação de clicar em "Ver mais" conforme o número de vezes especificado na coluna "Número de Páginas" do arquivo Excel.

Durante a coleta, o script extrai o HTML dos restaurantes listados e destila os dados necessários para compor um DataFrame com as colunas detalhadas acima. Cada coluna armazena informações pertinentes:

**1 - data:** Registra a data da coleta dos dados no formato dia/mês/ano.

**2 - codigo\_estabelecimento:** A posição numérica do restaurante na sequência de coleta, indicando a ordem em que aparecem na listagem do site para o endereço especificado.

**3 - endereco\_universidade:** O endereço que foi usado para realizar a coleta de dados, conforme informado no arquivo Excel.

**4 - nome\_restaurante:** O nome comercial do restaurante conforme listado no iFood.

**5 - categoria:** A classificação do tipo de comida ou serviço que o restaurante oferece (por exemplo, Padaria, Salgados, Brasileira, Japonesa, etc.).

**6 - horario\_coleta:** O horário exato em que os dados do restaurante foram coletados pelo script.

**7 - status:** Indicação se o restaurante estava operando (Aberto) ou não (Fechado) no momento da coleta.

**8 - score\_estrela:** A avaliação média do restaurante em estrelas, onde 5 é a pontuação máxima.

**9 - foto:** Um indicador que mostra se o restaurante possui uma foto de perfil personalizada ou se usa uma imagem padrão do iFood.

**10 - distancia:** A distância em quilômetros do restaurante até o endereço usado para a busca.

**11 - taxa\_entrega:** O custo de entrega cobrado pelo restaurante para aquele endereço.

Para lidar com imprevistos, como problemas de conexão ou desempenho do computador, o script está equipado para realizar até três tentativas de coleta por endereço. Se ocorrerem erros durante a execução, o script os registra num arquivo de texto denominado 'enderecos\_com\_erros.txt'. Quando a coleta ocorre sem contratempos, o arquivo de texto é marcado com a mensagem "Nenhum erro durante a coleta!", servindo como uma confirmação de que o processo foi concluído com sucesso.

Os dados coletados com sucesso são exportados para um excel com o nome no formato **‘ifood\_data\_%d-%m-%Y--%H-%M.xlsx’**, onde ‘d’ é o dia, ‘m’ é o mês, ‘%Y’ é o ano, ‘%H’ a hora e ‘%M’ os minutos, por exemplo: **‘ifood\_data\_15-02-2024--13-23.xlsx’.**

As imagens a seguir mostram o fluxo de acesso e coleta ao site:A screenshot of a computer

Description automatically generated

Figura 1: Página inicial ‘ifood.com.br’, onde o script clicará na barra de pesquisa para inserir o endereço.

A screenshot of a computer

Description automatically generated

Figura 2: Inserção e seleção do endereço.

A screenshot of a computer

Description automatically generated

Figura 3: Inserção e confirmação do número.

A screenshot of a computer

Description automatically generated

Figura 4: Confirmação da localiação.

A screenshot of a computer

Description automatically generated

Figura 5: Salvando o endereço fornecido.

A screenshot of a computer

Description automatically generated

Figura 6: Acesso ao link dos restaurantes, onde Podemos expandir o número de restaurantes que aparecem na página ao clicar em ‘Ver mais’.

# Fluxograma

O fluxograma a seguir serve como uma ilustração do funcionamento do script na coleta dos dados.

Arquivo em Excel ‘enderecos.xlsx’ com as configurações iniciais para o programa:

Endereços, Número, Quantidade de Páginas, Rodar

Script:

Leitura do arquivo Excel de configurações iniciais

Acesso ao site ‘ifood.com.br’

Inserção do endereço na barra de busca

Inserção do número do endereço

Confirmação do endereço

Acesso à aba de restaurantes

Coleta do número de páginas de restaurante fornecido no arquivo de configuração

Tratamento dos dados coletados

Escrita dos dados em arquivo Excel e escrita dos endereços com erro em .txt

Arquivo Excel com os dados dos Restaurantes

Arquivo de Texto com os restaurantes que tiveram erro na coleta

# Conclusão

Este script oferece uma solução automatizada para coletar dados de restaurantes no iFood, útil para análises de mercado, avaliação de concorrência, entre outros fins analíticos. A documentação detalha seu funcionamento, requisitos, e estrutura de código, proporcionando um entendimento claro do processo de scraping implementado.