

Python による Web スクレイピング

—教材データを収集する—

伊藤康明

教養科目「生活とかがく」の授業では、家庭学習の課題提出に Glexa を使用している。課題には、画像データがあれば添付するように指示している。これらのデータは、次回の授業で学生と共有して評価するのだが、このデータのダウンロードと整理に結構手間がかかる。また、提示用の教材資料を Web から収集することがあるが、この作業を省力化したいと思ったのが、プログラム開発の動機である。研究の一部を報告する。

1. Web スクレイピング

一般的な Web スクレイピングの手順は以下の通りである。

- ・対象の Web サイトから、画面を構成する HTML データを取得する
- ・取得した HTML データの構造を解析し、必要な項目だけを抽出する
- ・抽出したデータを変換し、CSV などのファイルに出力する

上記を、複数のサイトに対して繰り返す。抽出したデータは CSV だけではなく、画像など様々なファイルの形として出力可能である。この方法で、Web サイトからのデータの取得を自動化すれば、手作業で行うよりも作業効率が格段に向上する。

Web スクレイピングを行う場合、注意すべき点として、サイトの利用規約を守ること、サーバに負担をかけないように、データ取得には時間間隔をあけること、データの利用に際しては法に基づいて取り扱うことなどがある。

2. Python 言語

Python はデータの処理・分析・解析を得意とし、Web スクレイピングに長けたプログラミング言語である。必要なライブラリが豊富に揃っているため、ソースコードがシンプルに記述でき、初心者にも学びやすい。世界的なプログラミング共有サイト「GitHub」では、個人を含めて多くの技術者が作成した多数のライブラリがオープンになっており、誰でも自由な利用が可能である。

Web スクレイピングでは、HTML のダウンロードに Request、データ抽出に BeautifulSoup、ブラウザ操作に Selenium が利用できる。これらのライブラリを自分のデバイスにインストールし、Python コードで処理の流れを記述することによって、目的のプログラムを作成する。

3. プログラム例

(1) 植物、動物の写真を収集する

「生活とかがく」「幼児の環境」の授業教材用として、Web 画像を検索する機会が多いが、検索結果の中から多数の画像を確認するのは大変である。Google 検索が最もポピュラーであり、ダウンロードのためのライブラリがいくつか作られている。しかし、そのほとんどがそのままのコードでは動かない。これは Web ページの仕様が度々変更されることに起因している。Chrome の拡張機能として Image-Downloader があり、一括ダウンロードが可能であるが、20 個程度ダウンロードすると終了してしまう。これは google 側で対策を講じているのかもしれない。

そこで、Web ブラウザを制御するライブラリである Selenium を使用して、通常の google 検索の過程を辿って表示されるサムネイル画像を一括ダウンロードするプログラムを作成した。取得した画像は元画像ではないため、画素数は劣るが、数百枚の画像がダウンロードされるのでやむを得ない。図 1 は「セリ 春の七草」で検索・一括ダウンロードしたファイル約 600 個の一部を示す。

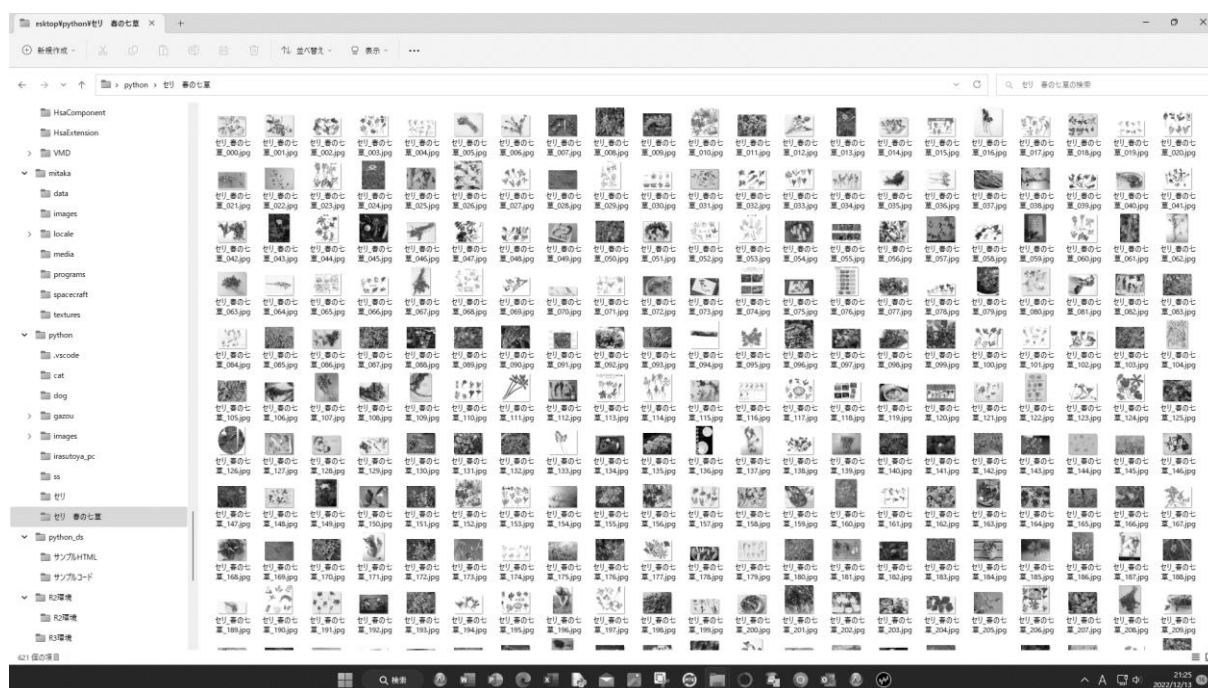


図1 フォルダに格納された画像ファイルの一部

(2) 気象データを取得する

「幼児の環境」の授業で、季節の変化について取り扱うとき、地域によって気候が異なることを、データで提示している。「Open Weather Map」のサイトは世界の都市の気象データを提供しており、API(Application Programming Interface) も用意されている。この仕組みを使うと、サーバにリクエストを送った場合、Web ページから必要なデータのみを抽出して返してくるので、プログラムでの処理が簡単になる。

このプログラムでは、都市名と取得したい気象情報を API に引数として渡してやると、結果が JSON 形式で返ってくる。

図 2 に処理結果を示す。

| | | |
|--|--|---|
| + 都市= Yokkaichi 天気= scattered clouds 最低気温= 7.3400000000 最高気温= 9.9399999999 湿度= 48 気圧= 1023 風速度= 5.66 | + 都市= London 天気= overcast clouds 最低気温= 10.90000000 最高気温= 13.41000000 湿度= 93 気圧= 1002 風速度= 6.69 | + 都市= Melbourne 天気= clear sky 最低気温= 24.56000000 最高気温= 29.02000000 湿度= 34 気圧= 1017 風速度= 5.66 |
|--|--|---|

図 2 気象データ (2022/12/20 15:00)

気象庁のサイトからも、アメダスデータや天気予報を取得することができる。

図 3 は四日市を指定して取得したデータの一部である。

```
21 時
現在の気温 : 0.0 度
現在の降水量(10分あたり) : 0.0 mm
翌日の天気: 晴れ 時々 くもり
天気概況 : 日本付近は強い冬型の気圧配置となっています。
三重県は、晴れまたは曇りとなっています。
18日は、強い冬型の気圧配置が続くためおおむね晴れますが、寒気の影響で雲の広がる所がある見込みです。
19日は、引き続き冬型の気圧配置となるためおおむね晴れますが、寒気の影響で雲が広がりやすいでしょう。
```

図 3 気象庁データ(2022/12/18 21:00)

(3) Glexa から、報告課題を取り出す

Glexa で提出された課題に添付された写真や動画を、効率的に取得して教材を作成する。

まず、Selenium を使って Glexa に自動ログインし、クラス画面から「提出レポート一覧画面」へ遷移する。Web ページを解析したところ、「ファイルの一括ダウンロード」機能があることが分かった。これはレポート全体をダウンロードするものと解釈していたのだが、実行してみるとレポートの添付ファイルのみを抽出してダウンロードしている。フォルダに保存された Zip ファイルを解凍すると、各ファイルが表示されるが、ファイル名に学生番号と氏名を含んでおり、そのまま授業で使用するのは好ましくない。1 個ずつファイル名から個人情報を削除するのは手間が大きい。そこで、OS モジュールを利用したプログラムにより一括変換を行って、ファイル名を通し番号に変えることにした。

当初は手作業で Glexa から添付ファイルを 1 個ずつダウンロードして整理していたが、提出日がばらばらであったり、授業直前の提出もあつたりして、結構授業前の作業に忙しかったが、一括ダウンロード機能により、助けられた。学生は自分の提出課題が授業で紹介されるので、提出が促され、動画で提出する学生も増えてきた。一方、これ

を止めて欲しいという声もあった。

図4は提出課題（シュリンクシート）の一覧画像である。

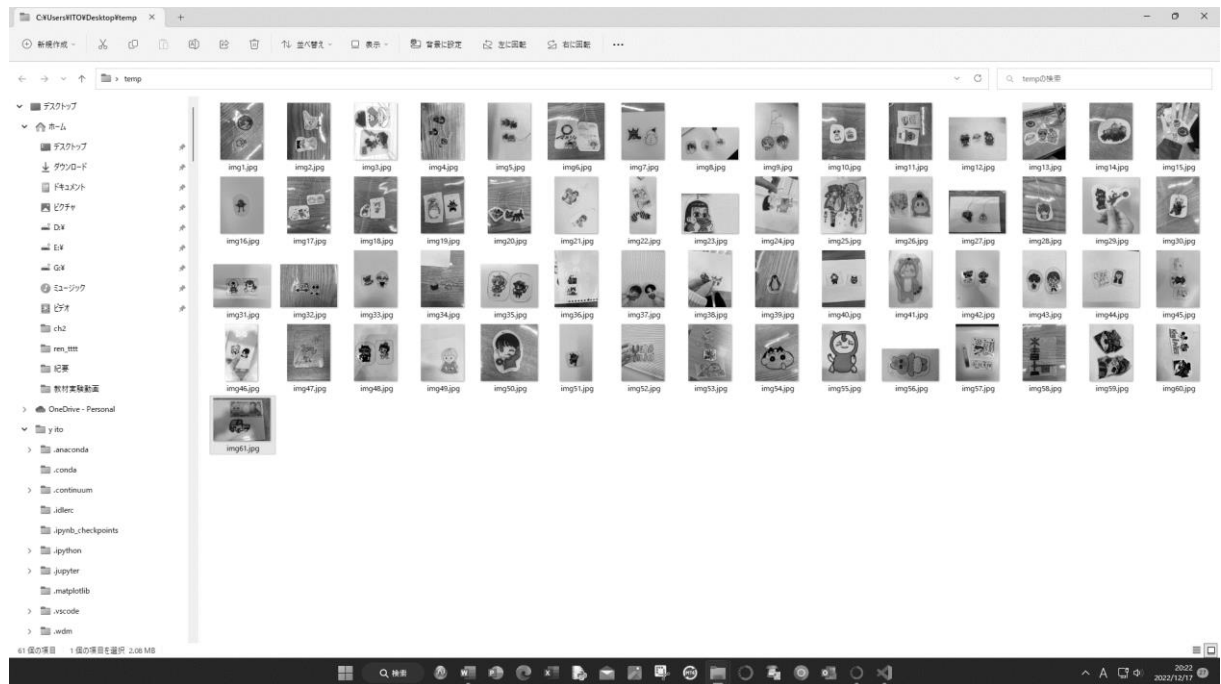


図4 提出課題の一括ダウンロード画像

4. おわりに

一般に Web スクレイピングのプログラムを半永久的に使用することは出来ない。これは、Web ページの仕様変更が度々行われ、HTML の記述が変更されるからである。この場合、プログラムの修正が必要となってくる。

官公庁や企業のサイトでは、API を用意して Web サービスの一部を公開している。利用は自由なので、効果的に活用してプログラムを作成すると良い。

参考文献

- 1) 鈴木たかのり他、「Python 実践レシピ」、技術評論社、2022/2/1
- 2) クジラ飛行機、「Python によるスクレイピング&機械学習」、技術評論社、2019/1/7
- 3) 清水義孝、「Python 最速データ収集術」、技術評論社、2022/2/8
- 4) Selenium クイックリファレンス」、<https://www.seleniumqref.com/>