

情報 I のための授業準備 プログラミング環境の構築

Python, JavaScript, Excel VBA, Scratch

東京学芸大学 特任教授 天良 和男

1. はじめに

2022年度から「情報 I」が必修となり、全員がプログラミングやデータ分析を学ぶことになる。

「情報 I」の教科書で扱われている主なプログラミング言語は、Python, JavaScript, Excel VBA, Scratchであることが明らかになった。

そこで本稿では、これらのプログラミング環境の構築にあたっての留意点を中心に取り上げる。

なお、紙面の都合上、インストール方法や使い方などの詳細情報については、右のQRコード、または下記のURLからリンクされている「開発環境の準備と使い方」に掲載されているので、これらもあわせて参照されたい。

開発環境の
準備と使い方



<https://www.jikkyo.co.jp/download/detail/61/9992660056>

2. 開発環境のインストールにあたっての留意点

不特定多数のユーザがアクセスする学校のPC環境には、個々のユーザによる勝手な設定変更やシステムファイルの削除、プログラムのインストールや実行などの安全をおびやかすリスクに対処するために、再起動するだけで自動的に必要な情報以外を破棄して元の状態に戻すための環境復元ソフトウェアが入っている。

プログラミング環境などの新規のソフトウェア

を学校のPCにインストールする際は、環境復元機能を一時的に無効（修復無効モード）にする、雛形となるディスク環境のイメージを端末のPCに一斉配信する、環境復元機能を有効（修復モード）に戻す、などの操作を適切に行う必要がある。

環境復元ソフトウェアが導入されている学校のPCに新規ソフトウェアをインストールする作業を行ったことがない場合は、PC教室の定期点検時などを利用して、保守点検業務を行う企業にインストール作業を依頼するほうがよい。

3. 構築や使用にあたっての留意点

プログラミング環境には、ファイルをダウンロード・インストールして利用するオフライン環境と、ダウンロード・インストールせずにWebブラウザを使って利用するオンライン環境がある。

3.1 Python

Pythonのプログラムを記述したファイルには、pyファイルとNotebookファイルがある。

pyファイルはPythonのプログラムだけを記述したテキスト形式のファイルであり、ファイルの種類を表す拡張子は「py」である。

これに対して、NotebookファイルはPythonのプログラムとともに、説明の文章やグラフ、実行結果などが含まれるJSONの書式で記述したテキスト形式のファイルであり、拡張子は「ipynb」である。Pythonのプログラムの編集、保存、実行、デバッグなどができる開発環境には、py形式のファイルを扱うものと、Notebook形式のファイルを扱うものがあるが、両方の形式のファイルに対応したものもある。

(1) py形式のファイルを扱う開発環境

・オフライン環境

このタイプには、Pythonに付属しているIDLE (Integrated DeveLopment Environment) やVSCoDe (Visual Studio Code) などがある。

IDLEはPythonをインストールすると利用でき、軽量かつシンプルで扱いやすいため、初心者向きの開発環境である。教科書に掲載されているプログラムを扱うにはIDLEで十分である。

VSCoDeはエディタであるため、Pythonもインストールする必要があるが、Jupyter拡張機能を追加することで、py形式だけでなく、Notebook形式のファイルを扱うこともできる。

・オンライン環境

Webブラウザ上でプログラムを記述し、それを実行できるオンライン環境にpaiza.IOなどがある。

paiza.IOはユーザ登録が不要であるが、登録すればプログラムを保存することもできる。

オンライン環境はインストールせずに利用できるが、対応していないモジュール (ライブラリやパッケージともいう) がある。paiza.IOでは、例えばグラフ描画のためのmatplotlibモジュールやGUIアプリの作成のためのTkinterモジュールなどには対応していないため、これらのモジュールを使用したプログラムは動作しない。

(2) ipynb形式のファイルを扱う開発環境

・オフライン環境

このタイプにはJupyter LabやJupyter Notebookがあり、ipynb形式だけでなくpy形式のファイルも扱うことができる。これらは、Pythonをインストールした後に単独でインストールすることもできるが、Anacondaをインストールしてもよい。

AnacondaはPython本体とJupyterLabや外部モジュールなどをまとめたPythonの配布形態 (ディストリビューション) の1つであり、これらを一括でインストールできるため、環境構築が容易であるが、不要なものまで導入されて動作が重くなることもある。このような場合は、Anocondaではなく個別にインストールしたほうがよい。

・オンライン環境

このタイプにはGoogle Colaboratoryがある。

利用するにはGoogleアカウントが必要であるが、操作方法はJupyter LabやJupyter Notebookと類似している。

なお、Google Colaboratoryはオンライン環境であるため対応していないモジュールがある。

例えばmatplotlibモジュールには対応しているが、Tkinterモジュールには対応していない。

また、90分経つと実行環境が初期化され、12時間経つとNotebook以外のファイルは削除される。

3.2 JavaScript

IEでは動かないコードがあるため、ブラウザはChromeやEdgeを利用することをお勧めする。

・オフライン環境

このタイプには、Windowsに標準搭載されているテキストエディタの「メモ帳」やVSCoDe (Visual Studio Code) などがある。これらはプログラムの作成・編集機能しかなく、実行環境はWebブラウザに搭載されている。

・オンライン環境

このタイプには、スマホアプリの開発ができるMonacaや登録不要のjsFiddleなどがある。なお、paiza.IOのJavaScriptはWebブラウザではなく、サーバ側のNode.jsで実行されるため、教科書に載っているプログラムはpaiza.IOでは動作しない。

3.3 Excel VBA

Excelがインストールされていれば利用できるが、開発環境の利用は、「開発」タブをリボンに表示し、「マクロの設定」を変更しておく必要がある。

3.4 Scratch

Scratchにはオフラインとオンライン環境がある。オンライン環境は、ChromeやEdgeなどのWebブラウザで利用できるが、IEでは動作しない。

4. おわりに

オンライン環境はインストール不要で便利であるが、一部のプログラムが動かない場合やネットワーク接続に問題が発生すると利用できない場合があるので、できるだけインストールすることをお勧めする。