

Python輪読会

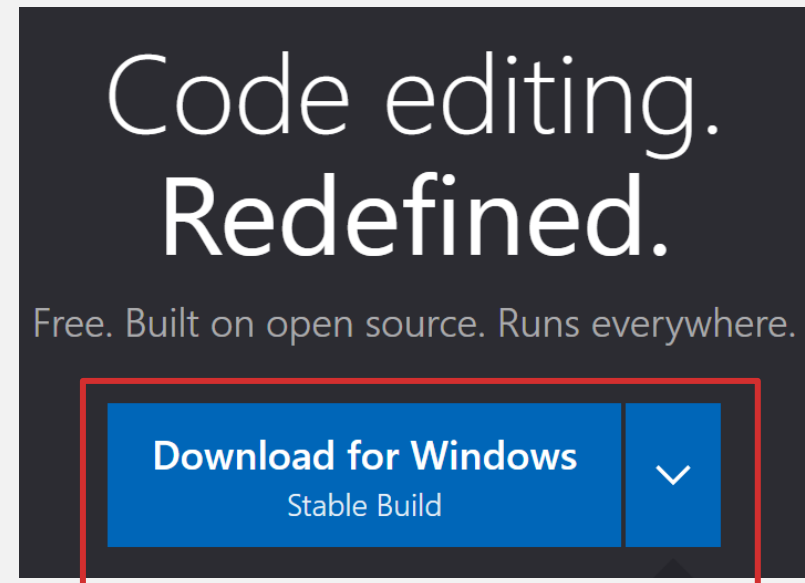
第1回

目次

- 環境構築
 - Visual Studio Code
- 0章 はじめてのプログラミング
- 次回以降の予定
 - 輪講の担当決定

Visual Studio Codeのインストール


- 次のリンクからインストーラーをダウンロード
https://code.visualstudio.com/?wt.mc_id=DX_841432
- サイト内の「Download for Windows」をクリックするとダウンロード開始
 - 右画像の赤枠部分
- あとは「次へ」を押すだけ
- インストール後は完了を押して起動

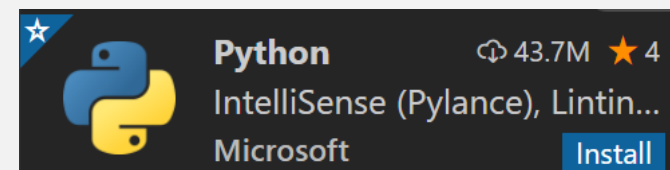


どうしてVisual Studio Codeなのか

- テキストエディターは世の中に多くの種類が存在
 - Atom: Githubが開発のエディター
 - SublimeText: 個人開発
 - 他にもVim、Emacsなどなど
- Visual Studio Codeはかなり負荷の小さいエディター
- 加えて開発がかなり盛ん
 - 常にメンテナンスがされている
- 多くの開発者が使用
 - howtoもたくさん

拡張機能のインストール

- 左のバーの  をクリックして拡張機能一覧を表示
- 検索欄に「Python」を入力すると一番上に右画像のような拡張機能が来るはず
- 「Install」をクリックしてインストール
- 使い方は画面共有を使って教えます



0章

はじめてのプログラミング

0.1 はじめてのPython

- Pythonはプログラミング言語の1つ
- AIやデータサイエンスなどのデータ分析の分野で注目されている
 - 他にもWeb開発やIoTなどにも利用され、汎用的
- 基本文法がシンプル
- 簡潔で読みやすいコード
- 便利な命令が豊富で直ぐに始めやすい
- **上記の特徴からプログラミングの学習によく用いられる**

0.2 はじめてのプログラミング

- JupyterLabを用いて、右図のコードを実行
- 「Hello world」という文字列を出力するプログラム
- 出力結果はコードの下に出力される

```
[1]: print('Hello, world')  
Hello, world
```


0.2 はじめてのプログラミング

- さらに「print()」の中身を変更
 - 「Hello, world」から「Good morning」に変更
- 中身を変えることで出力するものも変化
- 「print()」は**カッコ内に記述されたものを出力する命令**
- このような命令をPythonでは**関数**と呼ぶ

```
[2]: print('Good morning')
```

```
Good morning
```

0.2 はじめてのプログラミング

- 先と同じように文字列を出力するプログラム
- 右のコードを実行すると出力結果に「**Syntax Error**」というものが出力
 - 「Syntax Error」とはプログラムの文法が間違っているときの出力
 - 「'」が足りない
- エラーはプログラミングにおいて付きものなのでちゃんと**確認して解決することが大事**

```
[3]: print('Good morning)
```

```
File "<ipython-input-3-2a2469583343>", line 1  
    print('Good morning)  
          ^
```

```
SyntaxError: EOL while scanning string literal
```

0.2 はじめてのプログラミング

- 複数の文字列をまとめて表示するプログラム
- **上に書かれている文字列から順に出力**
- ちなみにコードが書かれている部分をJupyterLabでは**セル**と呼ぶ
- **別のセルの結果を利用することも可能**
- 複数のセルをまとめたものを**ノートブック**と言う

```
[4]: print('Good morning')  
      print('Hello, world')  
      print('This is a pen.')
```

```
Good morning  
Hello, world  
This is a pen.
```

0.3 Pythonプログラミングの基礎知識

- Pythonで書いたプログラムは以下の手順で実行

1. ソースコードの作成

- Pythonの文法に従って命令を記述
- 命令を記述したファイルをソースコードと呼ぶ
- 拡張子はPythonの場合、「.py」

2. 実行

- Pythonインタプリタと呼ばれるソフトウェアを使用してソースコードを機械語に変換・実行
- 文法上のエラーが出たり、命令の実行時にエラーが出た場合にはそこで中止

0.3 Pythonプログラミングの基礎知識

- Pythonプログラムの実行時に出てくるエラーは大まかに2通り
 1. Syntax Error
 - 機械語への変換前の文法チェック時に間違っていた場合に出てくるエラー
 2. Exception
 - 命令の実行時のエラー
 - エラー内容ごとに細かく種類が分かれる

0.3 Pythonプログラミングの基礎知識

- JupyterLabは統合開発環境の1種
- プログラムの開発を一つのソフトウェア上で完結可能
 - 普通にやる場合、テキストエディターでコードを書いてコマンドプロンプトを起動して実行し、エラーが出れば書き直す…ということをする
 - JupyterLabで作成したファイルは、Pythonインタプリタでは使用不可
- ちなみにJupyterLabはAnacondaというPythonにさらに色んなパッケージを追加したものに含まれている

0.3 Pythonプログラミングの基礎知識

- ソースコードを書くときには以下の点に注意
- 英数字は半角入力、**特に小文字と大文字の違い**に気を付ける
- **見た目が紛らわしい文字に注意**
 - l(小文字のエル)とI(大文字のアイ)など
- **括弧の種類を間違えない**
- **「'」と「"」を間違えない**
- **括弧や引用符(「'」と「"」のこと)は必ず同じもので閉じる**

0.3 Pythonプログラミングの基礎知識

- 読みやすいソースコードを書くにはコメントを書くことが肝要
- Pythonの場合、1行コメントと複数行のコメントは書き方が異なる
- 1行コメントは「#」から開始
- 複数行のコメントは「`"""`」で囲む

```
# これは一行コメント  
"""  
これは  
複数行の  
コメント  
"""
```


0.3 Pythonプログラミングの基礎知識

- プログラミングを学ぶ上で大事ななのはモチベーション
- やりたいことがあって、それを実現することを目標とするのもよし
- とりあえず勉強する、というのでも全然OK
- 楽しんで学ぶことが重要なので、エラーが出ても挫折せずに進めていきましょう

今後の予定

次回以降の予定

- 10/30: 1章
- 11/6: 2章
- 11/13: 3章
- 11/27: 4章
- 12/4: 5章
- 12/11~: 6章~
- 輪読終了後: 何かやります
 - グループワーク的なものを予定

輪講担当について

- 前期復習回(10/30~11/13)については3人体制で実施
 - 誰がどの範囲を担当するかは話し合って決めてください
- 以降は2人体制で
- 誰が次回を担当するかはその日に決めます