

自然言語処理 ——準備、正規表現——

<https://github.com/satoyoshiharu/nlp>

100本ノック第3章の位置づけ

- [100本ノック第3章の課題集](#)は、Pythonの正規表現です。
- 課題の内容は、自然言語処理の貴重なデータ源として巨大な言語テキストデータである Wikipedia からテキストを抜き出すために、正規表現を利用しています。Wikipedia のソースは、HTMLをはじめとするいろんなタグが複雑に混ざっているからです。
- ただ、BeatifulSoupというパッケージがあり、それを使うだけで、Wikipediaテキストを入手するという目的は達成できます。それよりも、正規表現は、実は、デバッグやネットワークサーバーのログから欲しい情報だけ抜き出すなど、様々な場面で日常的に活躍します。
- Pythonの正規表現は、基本的な部分だけ押さえて、あとは力業の世界です。必要になったときに、調べながら書くといいです。今は基本だけ押さえましょう。

正規表現資料

- Python reパッケージのマニュアル
 - <https://docs.python.org/ja/3/library/re.html>
- わかりやすい入門解説
 - <https://userweb.mnet.ne.jp/nakama/>
 - <https://takano.hatenablog.jp/entry/2019/03/22/053026>
 - https://dotinstall.com/lessons/basic_regex_v2

課題20～23,29

- 以下のスライドおよびリンクした資料などを参考に、[「100本ノック」の3章の課題20～23,29](#)を解いてみましょう。正規表現は、基本的な部分に慣れれば、あとは力ワザなので、基本的な課題20～23と、WikipediaのAPIに触れる29だけやりましょう。
- 「正規表現.ipynb」というノートをコピーし、冒頭の準備、基本事項をやった後、各課題のセクション下のコードセルに解答コードを書き、実行ログを残してください。
- ネットに解答集がいくつか見つけられます。講師の解答案も提供します。力をつけるため、なるべくそれらを見ないで自力でやってから、見てください。回答をコピーするのでは、力は少しもつかず、やった感を作るだけの、時間の無駄です。一方、考えながら、書くために必要なことを調べるのは成長につながる大切な時間となります。

確認クイズ

- 正規表現_確認クイズ.ipynb で力を試してください。穴埋めです。出力を指定しているので、それが出力できればOKです。