

2020年度Python会オンライン新歓動画 (8)

# プログラムの変更管理

～ バージョン管理システム **Git** を使ってみる ～

医学科4年 AtamaokaC

# 自己紹介など

- HN: AtamaokaC
- 医学科4年(学士編入)
- 研究は主に機械学習・自動診断
  
- 競プロやっています (AtCoder ID: [AtamaokaC](#))
- Python会Webサイト係 (<https://oumpy.github.io>)

今回は研究、、よりも**もっと基礎的な話**。

# もくじ

1. 「ファイル管理」の問題とは何か
2. バージョン管理システム **Git**  
(紹介と実演)
3. どんなことに使える？

(1/3)

「ファイル管理」の  
問題とは何か

# プログラミングとは

<新歓動画：ここまでの内容>

- 第1回 「神経生理学の視点で見るニューラルネットワーク入門」
- 第2回 「競プロのすすめ」
- 第3回 「研究とPython」
- 第4回 「深層学習を用いた画像処理」
- 第5回 「Python会でできること(bioinformatics, ikra編)」
- 第6回 「医学統計入門編 生存時間解析」
- 第7回 「プログラミングを使った脳波解析」

プログラミング = 「プログラムを書いたファイルを作る」作業

# プログラミング作業の実際

プログラムファイル

```
A = input()  
B = int(A)  
...
```

追加

```
A = input()  
B = int(A)  
...  
L = [B, C]  
...
```

修正

```
A = input()  
B = float(A)  
...
```



....

# プログラミング作業の実際

ほとんどの場合、、

- 一発で完成品が **できるわけがない**
- 原型を作ったら、**機能の追加や修正** を繰り返す
- よかれと思った **修正が失敗** になることもある
- 共同研究など: **同じファイルを複数人** で修正することも

→ ファイルは どうやって管理 する??

# 素朴な方法(1): 上書き保存

ひたすら上書きしていく!

- 修正がうまくいかなかったとき戻れない
- 修正中はちゃんと動くものがなくなる
- 共同作業で「どの部分はどれが最新か」不明  
→ 修正が消されてしまうことも

正直、論外。

## 素朴な方法(2): 過去版の保存

節目ごとにバージョンや日付をつけて**別名で保存**しておく

<よくなったところ>

- いざとなったら戻れる
- 修正中でもその前の版が使える

<やっぱりだめなところ>

- 共同作業でやはり修正が消える  
(結局**確認が面倒すぎる**ことが多い)
- 行ったり戻ったりすると**わけがわからなくなる**

微妙、、

# どんな仕組みがほしい？

- 過去の版にいつでも戻れる
- 行ったり戻ったりしても履歴が追える
- 変更点を把握できる
- 別々に加えた修正を統合できる

→ バージョン管理システム

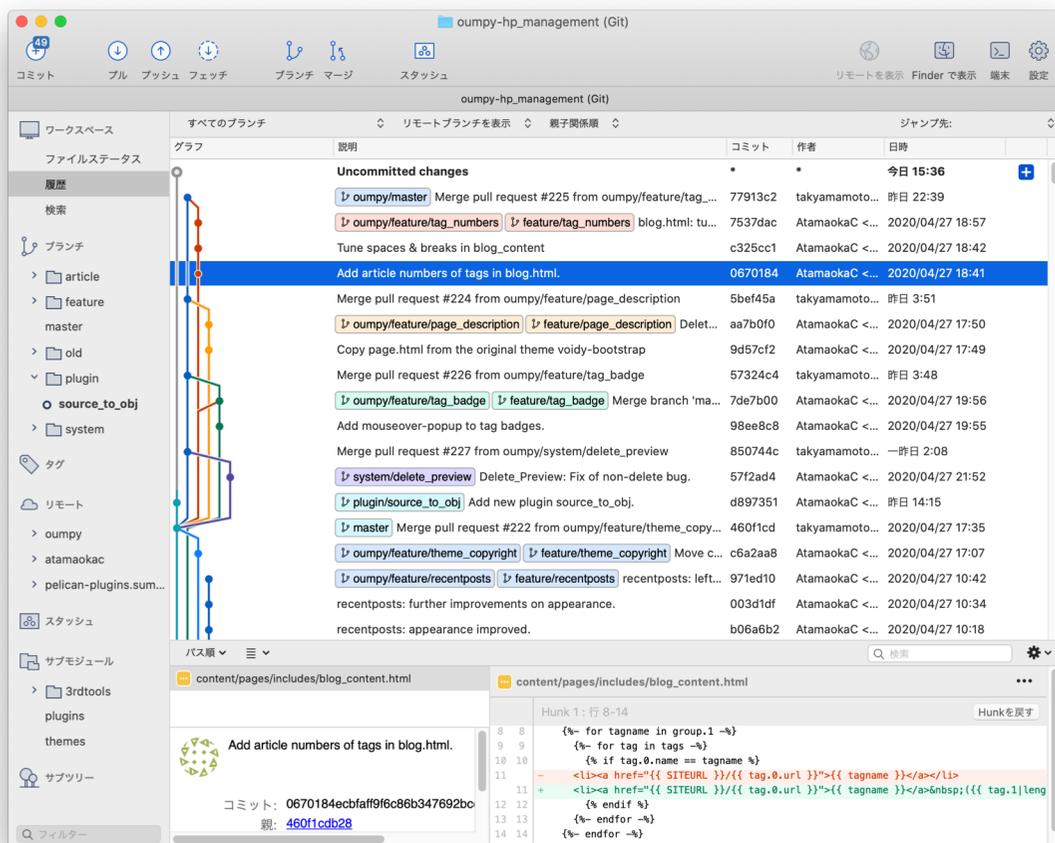
(特に **Git** が現在の主流)

(2/3)

バージョン管理システムGit

(紹介と実演)

# こんな感じ



修正の分岐・統合を  
ツリー状に記録



戻れる  
追える  
把握できる  
統合できる

# 以下、実演

## 注意

- Gitは裏で動く本体と、画面で操作するアプリ（クライアント）が分かれています。前者はMacなら最初から入っています。
- 後者は SourceTree というクライアントアプリを今回は使います。何を使っても基本的には同じです。 <https://www.sourcetreeapp.com>
- 簡単なPythonプログラムを書いていきますが、詳細は気にしなくて構いません。
- Atomというエディタを使いますが、これも何でもいいです。
- 今回はすべてGUIでやります。慣れたらコマンドでもどうぞ。

# 履歴データの共有

- コード共有サイトに履歴を全てアップする。  
(pushする、という)
- 最大手はGitHub <https://github.com>  
今回もこれを使います。  
(アカウントの作成・設定は今回割愛)
- アップしたコードは他人が共有可能。  
読み込み／書き込みできる人は設定可能。今回は公開

今回作ったレポジトリへは

<https://github.com/atamaokac/python-calc/>

でアクセスできます。

(3/3)

どんなことに使える？

# 用途(1) : プログラム開発全般

## Python会での利用例

- Ikra <https://github.com/yyoshiaki/ikra/>
- Python会HP [http://github.com/oumpy/hp\\_management/](http://github.com/oumpy/hp_management/)
- 競技プログラミング用ライブラリ  
<https://github.com/atamaokac/kyoprolib/>

etc, etc...

## 用途(2) : 文書類

- 論文原稿

LaTeXファイルをGit管理する

→ 共同編集が劇的に楽に！

- レポート、記事原稿など

MarkdownファイルなどをGit管理する

→ 一人の場合でも推敲が追いやすい！

## 用途(3) PCの設定ファイルとか、

- MacやLinuxはテキスト形式の設定ファイルが多い  
→ Git管理する人もいます。

# Git管理に（あまり）向かないもの

- バイナリファイル  
行単位で変更を管理するので、基本的に向きません
- 意図しなくても大幅に書き換えられるファイル  
Jupyter notebookなどが該当。  
まともにGit管理するには工夫が必要

逆に上記以外は基本何でもOK！！

# Gitを使いこなして 生産的な学生生活を

ついでにPython会にも入ろう