Git Branchの話

3

- ブランチのどこに自分がいるか表示させる

「git」branch」→ 最初は(*master)が出る

星は、現在地につく

masterは基本的に直接編集しない。のでご。

- masterからbranchを"切る"

※切る方法は複数あるが、そのひとつ。

「git」checkout」-b」名前を自分でつける」

(例) teian1 / teian2 / ⋮

切ったあと「git」branch」すると

master
*teian1

と出る。並びはアルファベット順。

- branchの間を移動する

「git」checkout」行きたいブランチ名」

ちなみに checkout」存在しないブランチ名」をするとエラーが出る。

- 切ったbranchを消す

主が切りたいbranch以外の場所に移動。どこでもよい。どの層にいても関係ない。

「git」branch」-D」消したいブランチ名」

(大文字) Delete

4

- ブランチの名前を変更する

「git」branch」-m」対象ブランチ名」変更ブランチ名」

(modify)

- ブランチで別々にcommitした内容を統合(merge)する

全マージのコマンド/手順

① マージ対象(例 master)に移動する

② 「git」merge」取りこみたいブランチ名」

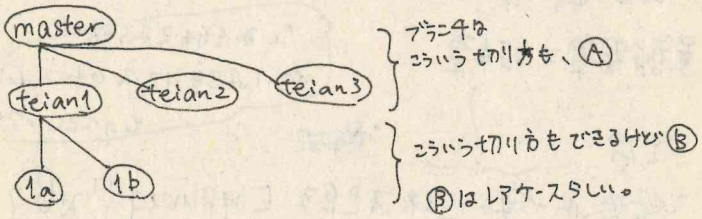
③ 「git」log」で上手くいったか確認

以上。

(マージ)

部分マージもできるし全体マージもできる。

全体こっちをするのは



ターミナル操作

2018.7.27

- ファイルを削除する

「rm」ファイル名」 =remove

- ディレクトリを削除する

「rm」-rf」ディレクトリ名」

消したい場所を直接指定してもいいし移動して書いてもいい。

recurcive(再帰する)

force ※「!」と同じ機能

- 今いる場所の配下のリストを表示する

基本は「ls」+ドットはターミナルだと隠しファイルを意味する。[ハイフンa]で出てくる。

「ltr」もオプション { long(+)time(+)reverce / 詳細 更新順 逆順 } のこと。

「ls」-altr]で合わせ技ができる。とても見せすぐる。

gitの初期管理の話とgitを止める方法

(.git)という隠しディレクトリが「git」init]すると(大切!)

gitを始めたディレクトリの中に生じる。この中に管理履歴が入ってく。

.gitを消すと全て消える=git終了となる。

- ファイル新規作成{兼}最終更新日の更新

「touch」ファイル名 またはファイルパス」。→場所指定のこと。

ターミナルの話

~Commit応用~

vimがめんどうなときは

「git」commit」-m」"vimに書くはずだったメッセージ"」

(message)

で省略できてしまう。

ダブルクオテーション忘れずいよう。

~コバナシ~

{ cd は current directory でもあるし Change directory でもある。

{ ターミナル画面は command line interface (CLI)
普段見てるやつは graphical user interface (GUI)

クセづけのために、git commit前のgit statusは

やった方が良い。最初に勝手にアレンジしないのは料理と一緒..