

■ GitHubアカウントの作成

STEP.1

GitHubのトップページへ移動 > <https://github.com/>

STEP.2

ユーザー名は以下のルールで設定してください。

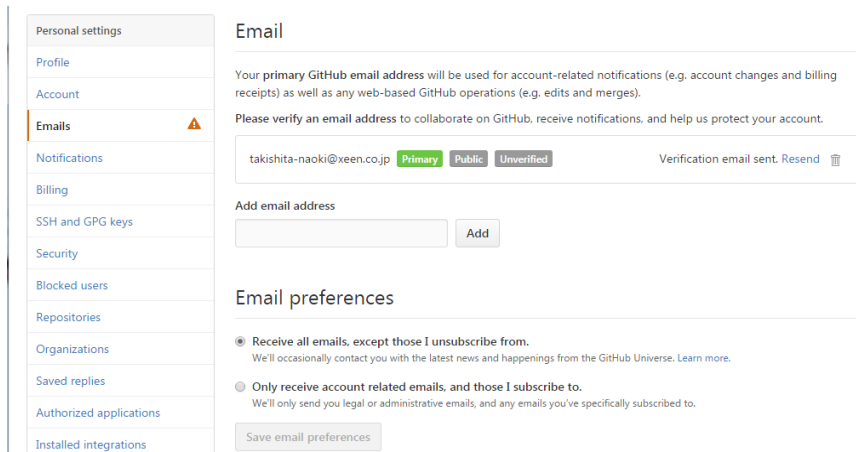
所属	ルール	例
徳島チーム	姓-名-Tokushima	Kiyohara-Kazuma-Tokushima
高松チーム	姓-名-Takamatsu	Aoki-Satoshi-Takamatsu
福山チーム	姓-名-Fukuyama	Takubo-Saki-Fukuyama
広島チーム	姓-名-Hiroshima	Terakawa-Terumi-Hiroshima
ジーンスタッフ	姓-名-xeen	Takishita-Naoki-xeen

パスワードは忘れないようにメモしてください。

STEP.3

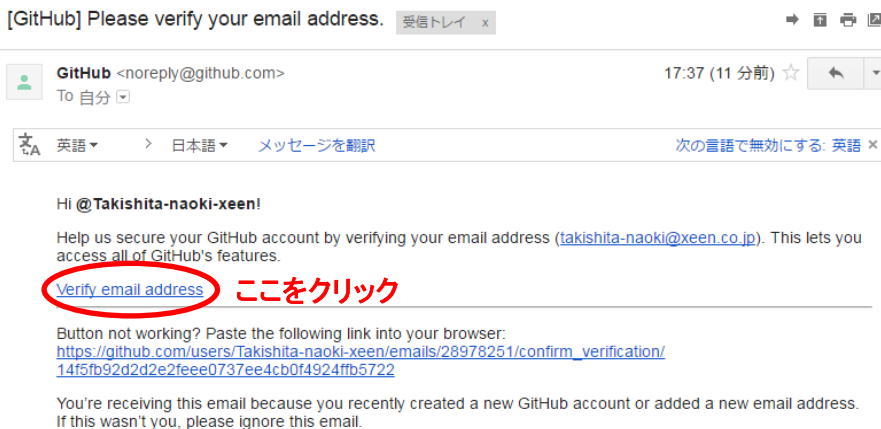
STEP.4

この画面になります。ここでは特に何もしません。



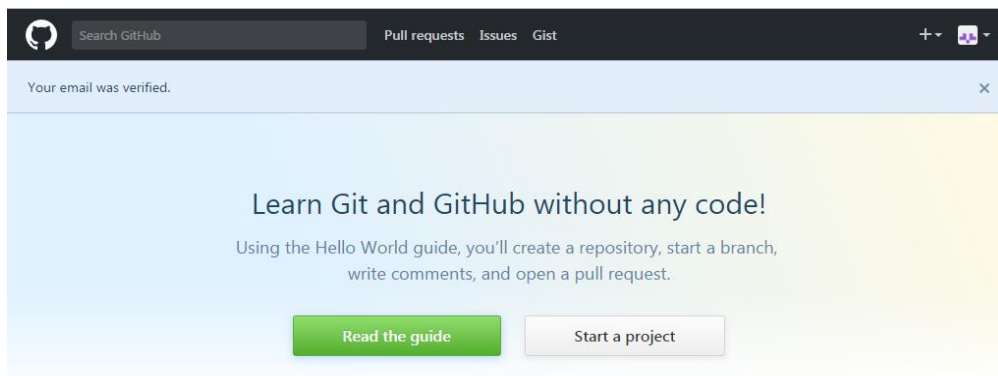
STEP.5

メールを確認してください。GitHubから以下のような英語のメールが届いているはずです。
[Verify email address](#) をクリックしてください。



STEP.6

メールアドレスが認証されて、GitHubのアカウント作成完了です。



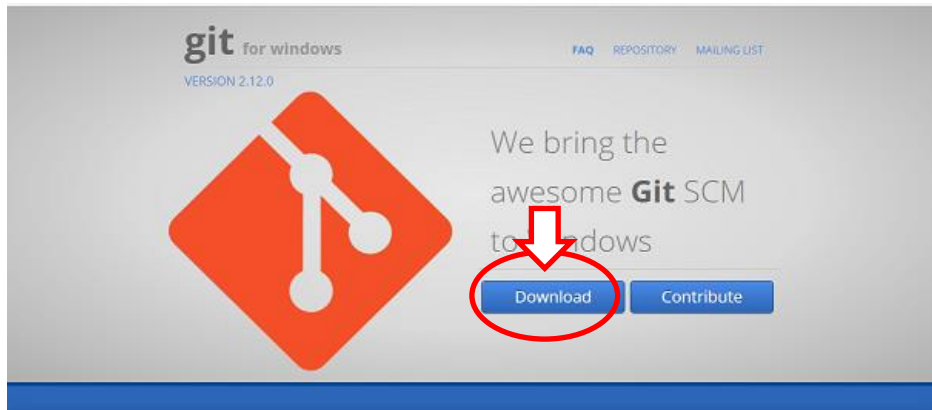
■ Gitを使うためのソフト(TortoiseGit)のインストール

《Windows用/プログラマー用》

①git for windows のインストール


STEP.1

Git for Windows を開く > <https://git-for-windows.github.io/>



STEP.2

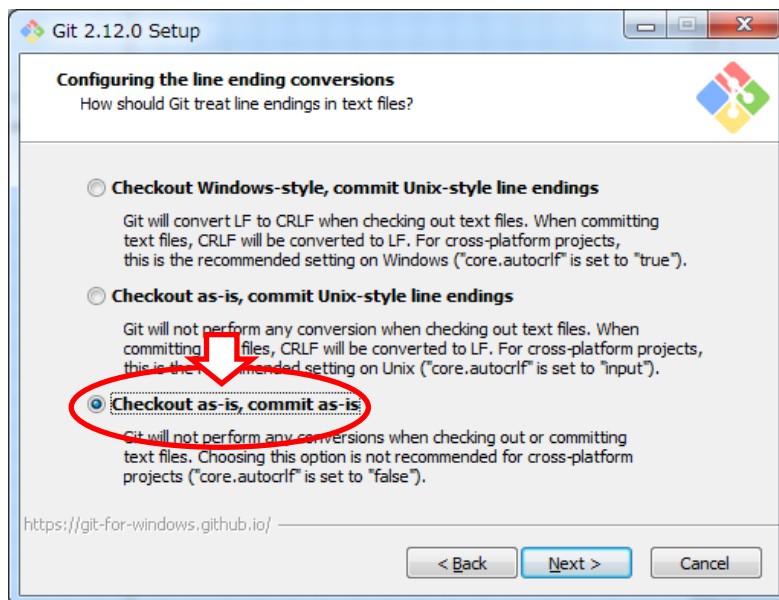
ダウンロードした exeファイルをダブルクリックで実行

 Git-2.16.2-64-bit.exe

その後は、ほぼ言いなりで Next を選択し続けるが、、、

STEP.3

ここで Checkout as-is, commit as-it を選択してからNext



STEP.4

ひきつづき、言いなりで Next を選び続けてインストールする。⇒ Finish

②Tortoise Git のインストール

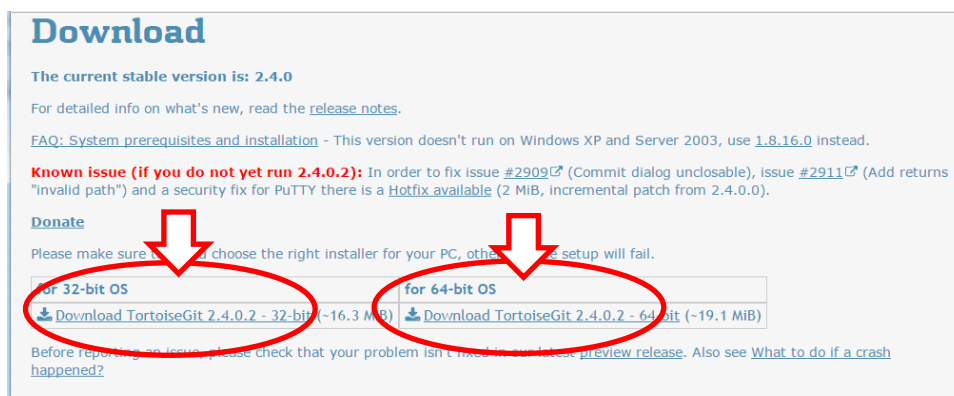
STEP.1

ダウンロードページを開く > <https://tortoisegit.org/>
Downloadを押す



STEP.2

32bitか64bitの自分のPCに合った方をダウンロードしてください。
どちらかわからない人は、先生に声をかけて自分のPCを見てもらおう！



STEP.3

さらにその下から、日本語の 32bit か 64bit もダウンロードしてください。

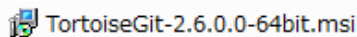
Language Packs

The language packs contain no standalone localized version of TortoiseGit, you need to download size of 2-6 MiB.

Language	Code	Completeness	32 Bit	64 Bit
Bulgarian	bg	55%	Setup	Setup
Catalan	ca	86%	Setup	Setup
Chinese, simplified	zh_CN	100%	Setup	Setup
Chinese, traditional	zh_TW	100%	Setup	Setup
Czech	cs	97%	Setup	Setup
Danish	da	50%	Setup	Setup
Dutch	nl	90%	Setup	Setup
Finnish	fi	49%	Setup	Setup
French	fr	97%	Setup	Setup
German	de	100%	Setup	Setup
Hungarian	hu	56%	Setup	Setup
Indonesian	id	51%	Setup	Setup
Italian	it	70%	Setup	Setup
Japanese	ja	96%	Setup	Setup
Korean	ko	90%	Setup	Setup

STEP.4

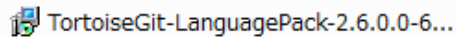
ダウンロードした TortoiseGit をダブルクリックしてインストール



言われるままに Next を押し続けられれば良いです。

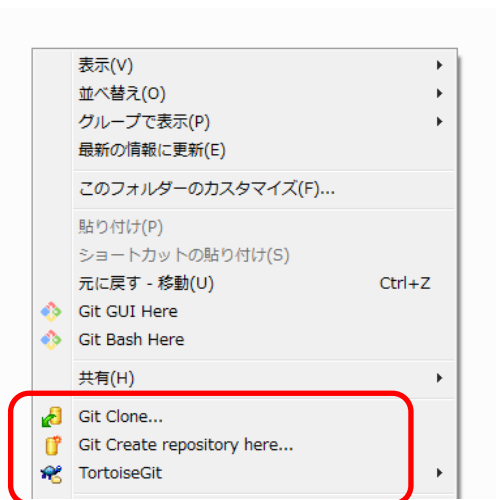
STEP.5

ダウンロードした日本語化パックをダブルクリックしてインストール



STEP.6

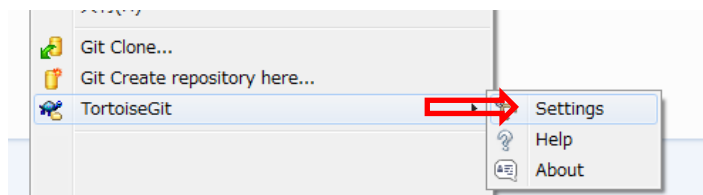
エクスプローラー上の適当な場所でマウスを右クリックした時にこういう項目がありますか？



無い場合はどこかで失敗しています！
先生に泣きつこう！！

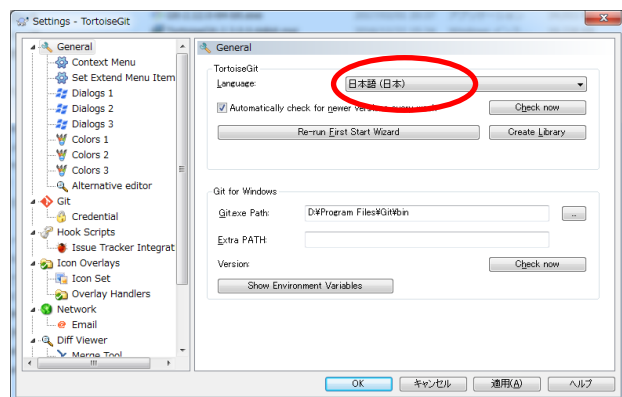
STEP.7

TortoiseGit から右側にマウスを進めて、Setup を選択

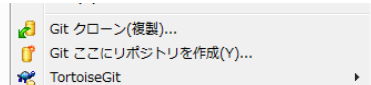


STEP.8

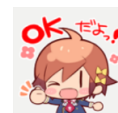
General から Language を English から 日本語 にへんこうして、OKする。



右クリックした時のGitメニューが
日本語になりましたか？



日本語にならなかった人は
先生に泣きつこう！



■ Gitを使うためのソフト (GitHub Desktop) のインストール

《Mac・Windows共通/デザイナー用》

① GitHub Desktop のインストール

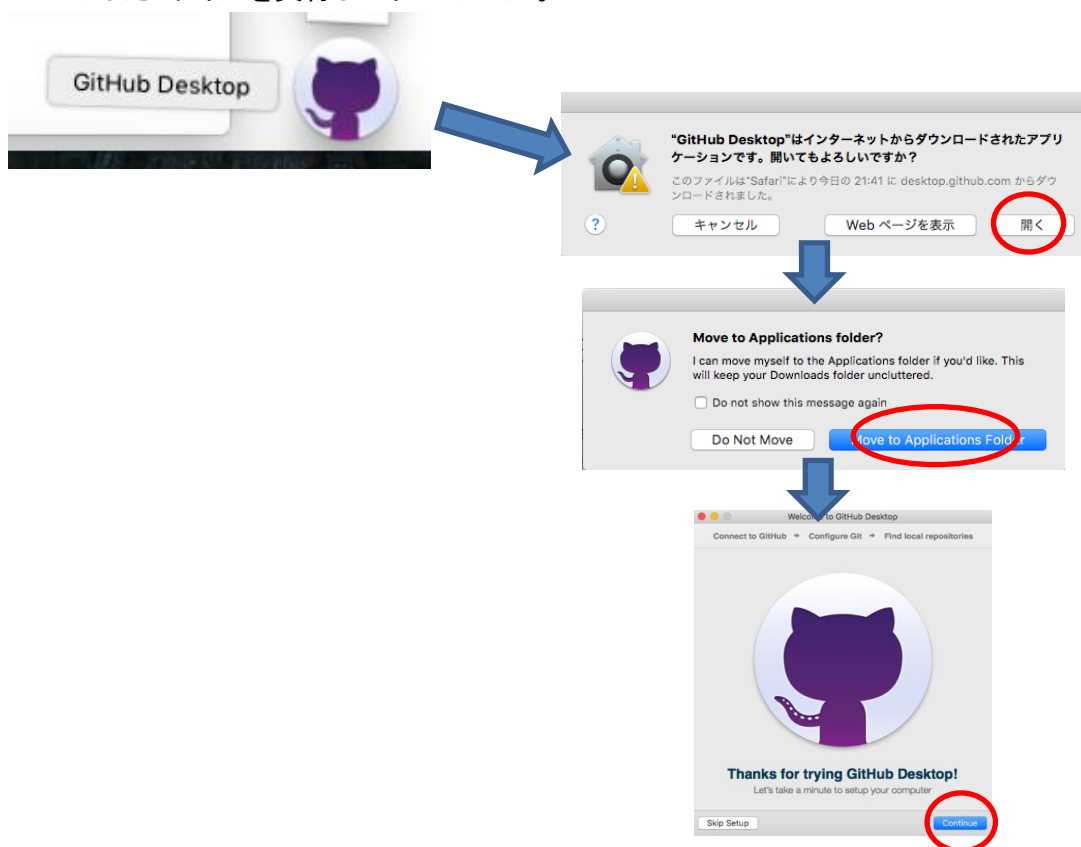
STEP.1

ダウンロードページを開く。 > <https://desktop.github.com/>
ダウンロードボタンを押す。



STEP.2

ダウンロードしたファイルを実行してインストール。



STEP.3

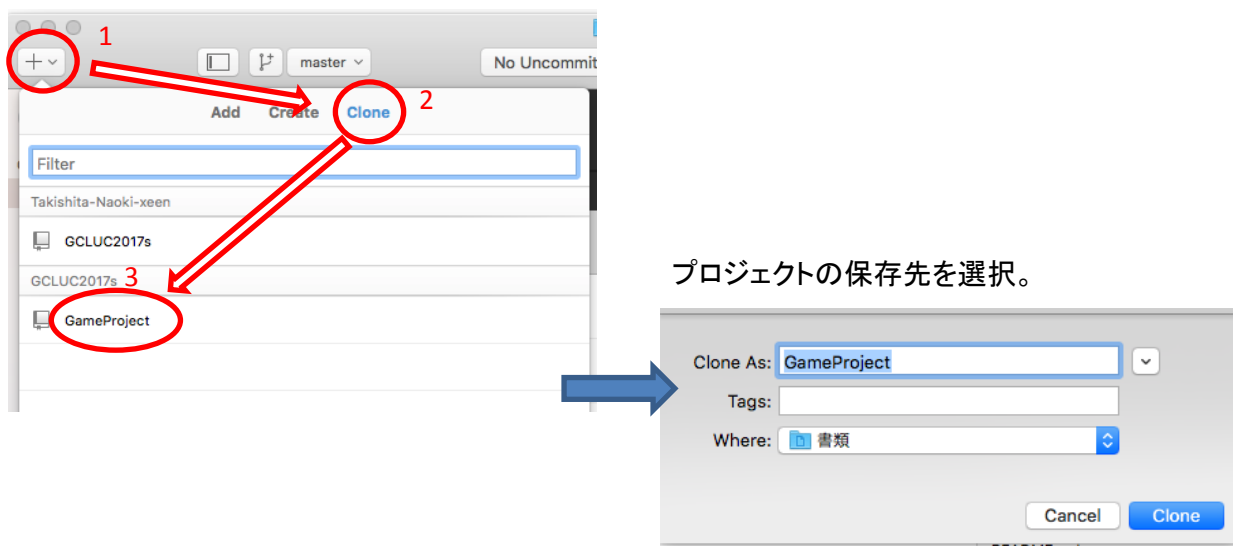
GitHub に登録したメールアドレスとパスワードを入力



STEP.4

GitHub Desktop が起動します。

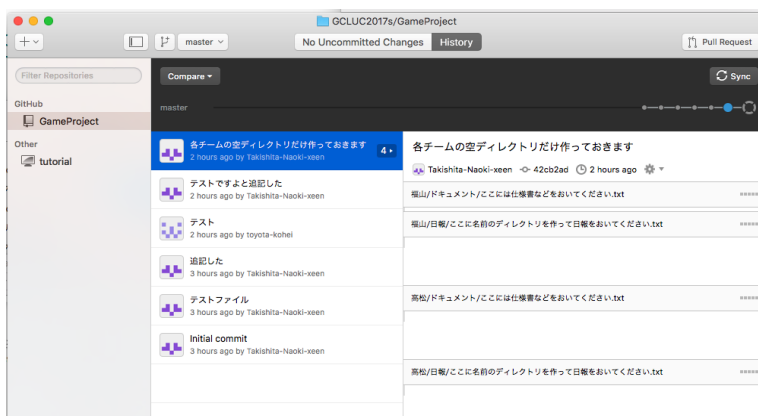
左上の + から Clone を選択して、GCLUC2018s の GameProject を選択してください。



プロジェクトの保存先を選択。

ここで指定した場所が、以降、GitHubにて管理されます。

準備完了。



■プログラマー用 TortoiseGit の使い方

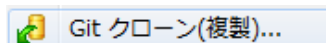
●サーバーからリポジトリをダウンロード (svnのチェックアウト相当)

STEP.1

自分のPCに Git で管理する作業ディレクトリを作る。

STEP.2

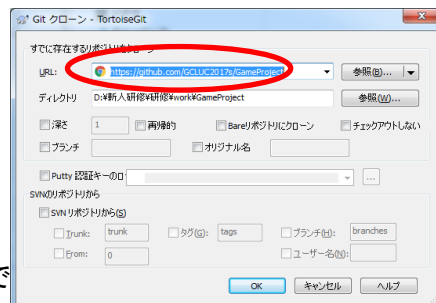
作った作業ディレクトリを右クリックして、Gitクローン を選択



STEP.3

URLに今回の研修用GitHubリポジトリのURLを入力

> https://github.com/GCLUC2018s/GameProject.git

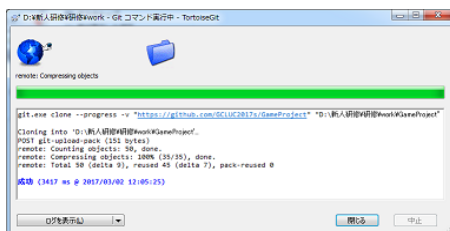


STEP.4

ダウンロードされます。

ウインドウに青い文字で 成功 と出るのを確認してください。

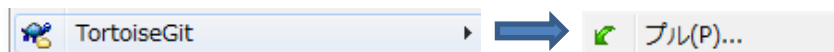
以降、TortoiseGit の操作では赤い文字が出るとエラーなので注意してください。



●サーバーから最新のデータを取得するとき (svnの更新 相当)

STEP.1

Git で管理しているディレクトリを右クリックして TortoiseGit → プル(P)

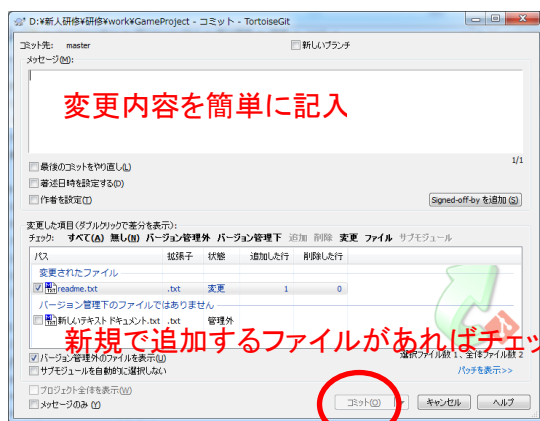
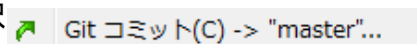


「OK」→「閉じる」

●ローカルの変更をサーバーにアップロード (svnのコミット 相当)

STEP.1

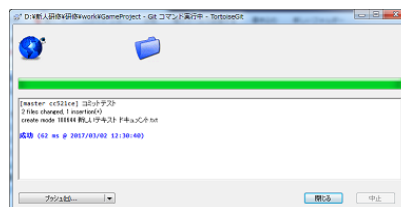
管理ディレクトリにて右クリックし、Gitコミット を選択



STEP.2

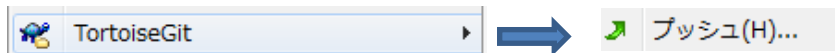
コミットログを記入(変更内容をわかりやすく)

追加するファイルがある場合はチェックを入れる
コミットボタンを押す

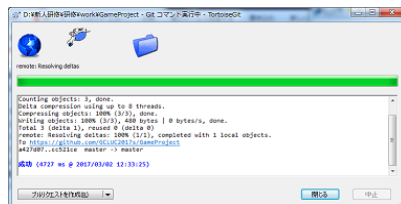


STEP.3

続けて、管理ディレクトリにて右クリック TortoiseGit → プッシュ



完了



●無視設定をするとき (svn 無視リストに追加 相当)

STEP.1

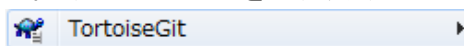
VisualStudio で共同作業をするような場合は、以下のディレクトリやファイルは同期管理から外さないとけません。

build ディレクトリ
*.suo
*.user

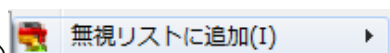
プロジェクト構成によっては異なる場合もあります。

無視設定にしたいディレクトやファイルを右クリック

TortoiseGit



→ 無視リストに追加
(または 削除して無視リストに追加)



→ 無視したいファイル名を選択、
または拡張子で指定する場合はそちらを選ぶ。

GreatGame.suo
*.suo

※なお、この利用方法はなるべく svn と近い感覚で操作できるように運用する場合ですので、Git本来のベストな利用方法とは異なるということを心に留めておいてください。

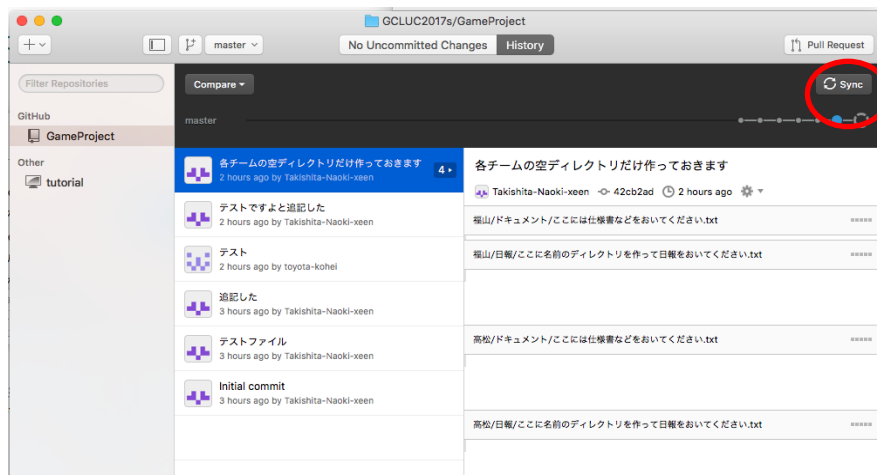


■デザイナー用 GitHub Desktop の使い方

●サーバーと同期させるとき(サーバーの最新データと同じ状態にする場合)

STEP.1

Sync ボタンを押す

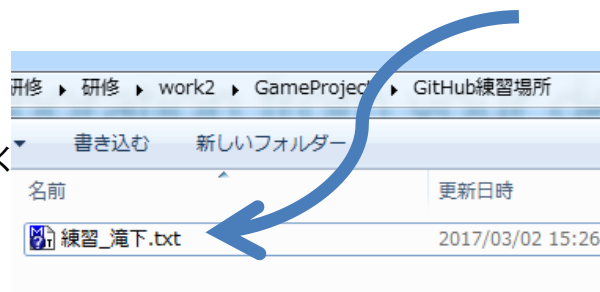


これでOK!

●GitHub管理のファイルを追加したい場合

STEP.1

GitHub で管理しているディレクトリの下にファイルを置く



STEP.2

GitHub Desktop を見てみよう。
Changes に印が付いていますね。

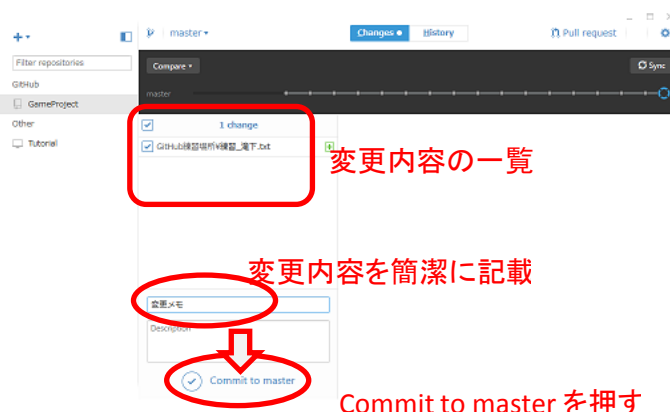
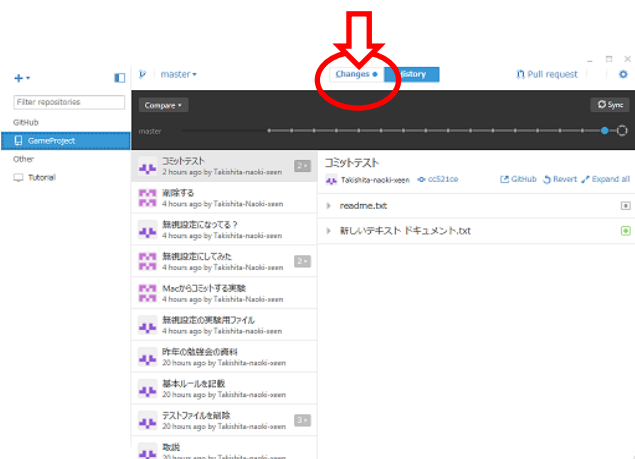
それでは Changes をクリックしてみてください。

STEP.3

変更があった内容がリストアップされています。

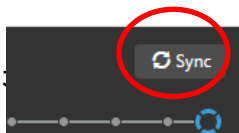
問題なければ、どういった変更なのか
コメントを記入してください。

その後、Commit to master を押す。



STEP.4

Syncボタンを押すとサーバーに
アップされて完了。

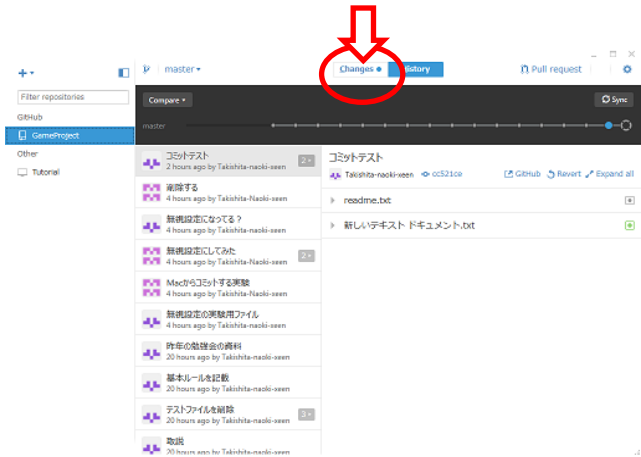


●ファイルの変更をサーバーにアップしたい場合
(画像を更新したりする時です。)

STEP.1

GitHub Desktop を見てみよう。
Changes に印が付いていますね。

それでは Changes をクリックしてみてください。



STEP.2

変更があった内容がリストアップされています。

問題なければ、どういった変更なのか
コメントを記入してください。

その後、Commit to master を押す。



STEP.3

Syncボタンを押すとサーバーにアップされて完了。

