

本日のテーマ： Git と GitHub とインターフェースとパッケージ

本日の目標：演習用のリポジトリをもとに自分の公開リポジトリをつくり、その一部を変更して公開する。

1. 時間になったら、授業の **meet** に接続せよ。
2. ビデオ視聴システムの説明および演習全体の説明を受けよ。
3. ビデオ(**git**, **GitHub**)を視聴せよ。ビデオのサイト、ユーザ名、演習中に知らせる。
4. 適宜、**git** の機能について、動かして確認せよ。**GitHub** に関わるものは、後で行う。
5. この指示書の用語の整理を参照し、について、どこから、どこへ情報を移すかを記入せよ。
(a) Add () から () へ
(b) Commit () から () へ
(c) Pull () から () へ
(d) Push () から () へ
(e) Fork () から () へ
6. **Git**・**GitHub** の補足資料の **pdf** を **classroom** より入手してどこに何か記載しているか確認せよ。(詳しく読む必要はない。)
7. ビデオに関する全体議論
8. **GitHub** に各自ユーザ登録せよ。すでに登録済みならば、それを使っても良い。
GitHub のユーザ登録について
新規の登録は、<https://github.com/>のページから行う。
ユーザ名は、使われていない限り **tut+学績番号** (例 **tut123456**)
使われている場合は、適宜、その後にアルファベットを追加
9. 下記の演習用の **GitHub** リポジトリを自分の **GitHub** アカウントで **Fork** せよ。自分のリポジトリ名は原則、演習用のリポジトリ名と同じとすること。これは、**GitHub** で行う。

演習用のリポジトリは

<https://github.com/TUT3Software4Project/2020informationQuantity>

10. 下記の **Google** スプレッドシートを開き、**Fork** した **GitHub** のリポジトリ名を記入せよ。
補足説明:**GitHub** の標準的な使用法では、演習用のリポジトリを「共有リポジトリ」として使い参加者に編集権限をあたえる。しかし、この演習では、各自がリポジトリを **fork** して公開するような使用法を採用する。このようにする理由は、不適切な変更がなされることによるトラブルを回避するためである。この方法では、公開されているリポジトリは、だれでも手元に取り寄せることができる。これを利用して教員は、各自の公開リポジトリにアクセスして、学籍番号のディレクトリの下のファイルを取り寄せて、評価を行う。このために情報の記入が必要となる。正しく記入しないと採点ができない。

GitHub の (ユーザ名を含む) リポジトリ名の報告シート：

<https://docs.google.com/spreadsheets/d/1CkHiWKqc9vuiDFhvk-21aIGtQbQ7J4KAqkNQLvQIT68/edit?usp=sharing>

<<<<<< ここまでは、必ず、演習時間中に行え。

-- これ以降は、演習中に終わらなかったら、各自時間をとって行え。

11. git のユーザ名とメールアドレスの登録をする。名前は、各自のものに取り替えること。セクターのマシン、xdev.edu.tut.ac.jp では、すでに登録済みのはずであるが、確認せよ。

登録の確認

```
git var -l
```

登録

```
git config user.name "Nanashi Gonbe"
```

```
git config user.email foo@example.com
```

12. Fork した自分の GitHub のリポジトリを自分のローカルリポジトリに clone せよ。

注意:ここでは、演習用のリポジトリを使っては*いけない*。報告したリポジトリを使え。 コマンド、資料 25 ページ「既存リポジトリの clone」を参照せよ。

13. README.md のある場所に移動し、git log をタイプすることによって、履歴を確認せよ。これ以降、git のコマンドの実行は、README.md のある場所で行うものとする。

14. 作業ツリーの自分の学籍番号のディレクトリの Hello.java の出力メッセージを変更せよ

実行方法は次のページを参照せよ。

15. Hello.java の変更を Commit したあと、Push せよ。Commit については、資料 26 ページを 参照せよ。Push については、資料 52 ページを参照せよ。

16. GitHub にログインし、変更がリポジトリに反映されていることをブラウザで確認せよ。

17 Java の interface と package の機能について、復習せよ。具体的には、Frequencer.java を参照し、そこで使われている機能が理解できることが必要である。

18 自分の好みの環境に git をインストールして使えるようにせよ。

19 自分のリポジトリを clone して、自分の好みの環境でソースコードを変更しない状態（パッケージをそのまま使う状態）で、Hello.java が実行できるように環境を整えよ。

サンプルコードの Hello.java 実行方法

自分の番号のディレクトリに移動して、make; make run で実行できる。

注意 1: ローカルリポジトリの変更は、各自の学籍番号のディレクトリの下のものだけにすること。

注意 2: プログラムでは先頭に宣言してある学籍番号のパッケージを必ずそのまま使え。パッケージ

名を変更したプログラムは、プログラムの仕様を満たしていないものとして評価される。パッケージを使わないプログラムは受け付けられない。

Git とGithubに関する用語の整理

