

Chapter 1

Git and GitHub

intro

Explain what git and GitHub are and their purposes

1.1 Config

Cette section devrait déjà être faite, vous devriez déjà avoir un compte github, avoir téléchargé git et avoir configuré votre email. Si jamais ce n'est pas fait, voici comment le faire.

Commencez par ouvrir votre terminal et taper "git"

Download : <https://git-scm.com/> xcode-select --install

Configuring git :

```
git config --global user.email "votre_courriel@ulaval.ca"
git config --global user.name "votre_nom"
```

token d'identification (Mac) (<https://docs.github.com/fr/authentication/keeping-your-account-and-data-secure/managing-your-personal-access-tokens#creating-a-personal-access-token-classic>)

1.2 Première étape avec Git

Cloner le repo de ce tutoriel aller chercher le lien du repo

cloner un repo

```
git clone https://github.com/cia-ulaval/tutoriels-h2024.git
```

Definition 1 *La commande `git clone <url-du-repo>` permet de créer une copie locale du projet.*

```

Last login: Sat Jan  4 11:15:58 on tty800
Welcome to fish, the friendly interactive shell
Type help for instructions on how to use fish
mathieubazinet@MBP-de-fsg ~$ git
usage: git [-v | --version] [-h] [--help] [-C <path>] [-c <name>=<value>]
          [--exec-path<= <path>]] [--html-path] [--man-path] [--info-path]
          [-p | --paginate | -P | --no-pager] [--no-replace-objects] [--bare]
          [--git-dir<= <path>] [--work-tree<= <path>] [--namespace<= <name>]
          [--super-prefix<= <path>] [--config<= <name>=<value>]
          <command> [<args>]

These are common Git commands used in various situations:

start a working area (see also: git help tutorial)
  clone Clone a repository into a new directory
  init Create an empty Git repository or reinitialize an existing one

work on the current change (see also: git help everyday)
  add Add file contents to the index
  mv Move or rename a file, a directory, or a symlink
  restore Restore working tree files
  rm Remove files from the working tree and from the index

examine the history and state (see also: git help revisions)
  bisect Use binary search to find the commit that introduced a bug
  diff Show changes between commits, commit and working tree, etc
  grep Print lines matching a pattern
  log Show commit logs
  show Show various types of objects
  status Show the working tree status

grow, mark and tweak your common history
  branch List, create, or delete branches
  commit Record changes to the repository
  merge Join two or more development histories together
  rebase Reapply commits on top of another base tip
  reset Reset current HEAD to the specified state
  switch Switch branches
  tag Create, list, delete or verify a tag object signed with GPG

collaborate (see also: git help workflows)
  fetch Download objects and refs from another repository
  pull Fetch from and integrate with another repository or a local branch
  push Update remote refs along with associated objects

'git help -a' and 'git help -g' list available subcommands and some
concept guides. See 'git help <command>' or 'git help <concept>'
to read about a specific subcommand or concept.
See 'git help git' for an overview of the system.
mathieubazinet@MBP-de-fsg ~$

```

Figure 1.1: Sortie attendue dans le terminal lorsque la commande “git” est tapée dans le terminal.



Figure 1.2: À remplir