

## SUIVI ATELEIR PRATIQUE VERSIONING

---

### 1) \$ git config --global --list

- affiche la configuration Git *globale* de l'utilisateur (fichier ~/.gitconfig).
- **Quand l'utiliser** : pour vérifier ton nom, ton email, l'éditeur, etc.

### 2) \$ git config --global user.name "ksamb81"

- enregistre le **nom d'auteur** qui sera inscrit dans les commits de *tous* tes dépôts (portée globale).
- **Remarque** : peut-être surchargé au niveau dépôt avec git config user.name "Autre Nom".

### 3) \$ git config --global user.email "ksamb@groupeisi.com"

- enregistre l'**adresse email** utilisée dans les commits (portée globale).
- l'email doit correspondre à celui déclaré sur GitHub si tu veux lier tes commits à ton compte.

### 4) \$ git init

- initialise un **nouveau dépôt Git** dans le dossier courant (crée .git/).
- **Remarque** : depuis Git récents, la branche par défaut peut être main (selon ta config). Tu peux forcer :  
git init -b master *ou* git init -b main, selon ta convention.

### 5) \$ git commit -m "debut validation"

- crée un **commit** avec le message fourni.
- **Prérequis** : il faut **avoir des fichiers indexés** (staged) via git add. Sinon Git répondra "nothing to commit".  
(On peut faire un commit vide avec --allow-empty si besoin.)

### 6) \$ git remote add origin https://github.com/ksamb81/Dev20252026.git

- ajoute un **remote** nommé origin pointant vers ton dépôt GitHub (URL HTTPS).
- **Remarque** : pour SSH : git remote add origin git@github.com:ksamb81/Dev20252026.git
- **Authentification** : en HTTPS, GitHub demande un **token personnel** (pas le mot de passe) au push.

### 7) \$ touch README

- crée un **fichier vide** nommé README (sans extension).
- **Remarque** : sur Windows PowerShell, touch n'existe pas par défaut ; utiliser ni README (New-Item) ou echo. > README.  
Souvent on utilise README.md en Markdown.

### 8) \$ git add README

- place README dans l'**index** (staging area) pour le prochain commit.

#### 9) \$ git add .

- ajoute **tous les fichiers modifiés/nouveaux** du répertoire courant (sauf ceux ignorés).
- **Remarque** : redondant juste après git add README, mais acceptable.  
Pour éviter d'ajouter des fichiers indésirables, pense à .gitignore.

#### 10) \$ git push origin master --force

- **envoie** la branche master locale vers origin en **forçant** la réécriture de l'historique distant.
- **DANGER** : --force peut **écraser** des commits déjà présents sur GitHub (perte pour d'autres).  
Préférer git push -u origin master (sans force) au premier push, et seulement utiliser --force-with-lease si tu dois réécrire l'historique en connaissance de cause.
- **Nom de branche** : si ton dépôt local s'appelle main, ce push sur master ne fera rien.  
Vérifie avec git branch.

#### 11) \$ git status

- affiche l'**état de l'arbre de travail** : fichiers modifiés, indexés, non suivis, branche active, divergence avec le remote.
- **Quand l'utiliser** : très souvent, avant un commit ou un push, pour savoir où tu en es.

#### 12) \$ git push origin master --force

- **Même effet que #10.**  
Évite de forcer à répétition. Si le but est de définir la branche amont au premier push : git push -u origin master (une seule fois).

#### 13) \$ git branch

- liste les **branches locales** et met en évidence la branche courante avec \*.
- git branch -vv montre aussi le tracking et le dernier commit.

#### 14) \$ git branch bkara

- **crée une branche locale** bkara (sans y basculer).
- **État** : tu restes sur la branche actuelle (probablement master/main) après cette commande.

#### 15) \$ git checkout bkara

- **bascule** sur la branche bkara.
- **Remarque** : avec Git récents, on préfère git switch bkara pour plus de clarté.

#### 16) \$ git checkout -b bkara1

- crée puis **bascule** immédiatement sur la branche bkara1, à **partir de la branche courante** (bkara ici).
- **Équivalent moderne** : git switch -c bkara1.

#### 17) \$ git commit -am "BrancheRevalidation2"

- **ajoute** (-a) **tous les fichiers déjà suivis** (tracked) *modifiés/supprimés* à l'index **et crée** un commit.
- -a **n'ajoute pas** les **nouveaux fichiers** non suivis. Pour eux, il faut un git add au préalable.

#### 18) \$ git push origin bkara1 --force

- pousse la **branche bkara1** vers le remote origin en **forçant** l'historique.
- **Bonnes pratiques** : premier push : git push -u origin bkara1 (sans force).  
Si tu dois réécrire l'historique : --force-with-lease plutôt que --force.

## Synthèse :

- **Éviter --force** sauf nécessité. Préférer --force-with-lease.
- **Nom de la branche par défaut** : assure-toi si tu utilises master ou main et sois cohérent côté GitHub (paramètre *Default branch*).
- **Premier push** : git push -u origin <branche> pour fixer la branche suivie (tracking), ensuite un simple git push suffit.
- **README.md** : préfère README.md pour une jolie présentation sur GitHub.
- **Nouveaux fichiers** : n'oublie pas git add (le -am n'inclut pas les fichiers non suivis).
- **HTTPS vs SSH** : en HTTPS, configure un **token** ; en SSH, ajoute ta **clé publique** à GitHub.

## Par la Pratique :

Clic droit sur un projet (dossier) / git Bash here

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

\$ git config --global --list

```
user.name=baye
user.mail=ksamb@groupeisi.com
user.email=baye@gmail.com
user.kara=karasamb
user.ksamb@groupeisi.com=ksamb@groupeisi.com
core.editor=code --wait
color.ui=auto
```

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

\$ git config --global user.name "ksamb81"

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git config --global user.email "ksamb@groupeisi.com"**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git config --global --list**

```
user.name=ksamb81
user.mail=ksamb@groupeisi.com
user.email=ksamb@groupeisi.com
user.kara=karasamb
user.ksamb@groupeisi.com=ksamb@groupeisi.com
core.editor=code --wait
color.ui=auto
```

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git init**

Initialized empty Git repository in C:/Users/HP/Desktop/Ecoles/.git/

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git commit -m "debut validation"**

On branch master

Initial commit

Untracked files:

(use "git add <file>..." to include in what will be committed)

```
"Exemple pour pr\303\251sentation.pbix"
Institut superieur.pbix
"Liste \303\251tudiants.xlsx"
administration-professeurs et facture.xlsx
"donn\303\251es d'enquete etudiants.xlsx"
"~$Liste \303\251tudiants.xlsx"
~$administration-professeurs et facture.xlsx
"~$donn\303\251es d'enquete etudiants.xlsx"
```

nothing added to commit but untracked files present (use "git add" to track)

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git remote add origin https://github.com/ksamb81/Dev20252026.git**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ touch README**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git add README**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git add .**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git add .**

```
[master (root-commit) cf5856a] Revalidation
9 files changed, 0 insertions(+), 0 deletions(-)
create mode 100644 "Exemplaire pour pr\303\251sentation.pbix"
create mode 100644 Institut superieur.pbix
create mode 100644 "Liste \303\251tudiants.xlsx"
create mode 100644 README
create mode 100644 administration-professeurs et facture.xlsx
create mode 100644 "donn\303\251es d'enquete etudiants.xlsx"
create mode 100644 "~$Liste \303\251tudiants.xlsx"
create mode 100644 ~$administration-professeurs et facture.xlsx
create mode 100644 "~$donn\303\251es d'enquete etudiants.xlsx"
```

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git push origin master --force**

```
Enumerating objects: 9, done.
Counting objects: 100% (9/9), done.
Delta compression using up to 4 threads
Compressing objects: 100% (8/8), done.
Writing objects: 100% (9/9), 505.66 KiB | 15.32 MiB/s, done.
Total 9 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)
To https://github.com/ksamb81/Dev20252026.git
* [new branch] master -> master
```

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

\$

**Nb : vous pouvez Faire des mises à jour sur le dossier en ajoutant des fichiers, mettre à jour les fichiers, etc., puis taper à nouveau les instructions :**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git status**

On branch master

Untracked files:

(use "git add <file>..." to include in what will be committed)

TD BD IAGE.pdf

nothing added to commit but untracked files present (use "git add" to track)

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git add .**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git commit -m "Revalidation2"**

[master a1f9131] Revalidation2

1 file changed, 0 insertions(+), 0 deletions(-)

create mode 100644 TD BD IAGE.pdf

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git remote add origin https://github.com/ksamb81/Dev20252026.git**

error: remote origin already exists.

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git push origin master --force**

Enumerating objects: 4, done.

Counting objects: 100% (4/4), done.

Delta compression using up to 4 threads

Compressing objects: 100% (3/3), done.

Writing objects: 100% (3/3), 266.61 KiB | 20.51 MiB/s, done.

Total 3 (delta 1), reused 0 (delta 0), pack-reused 0 (from 0)

remote: Resolving deltas: 100% (1/1), completed with 1 local object.

To https://github.com/ksamb81/Dev20252026.git

cf5856a..a1f9131 master -> master

Gestion des branches:

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git branch**

\* master

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git branch bkara**

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (master)

**\$ git checkout bkara**

Switched to branch 'bkara'

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (bkara)

**\$ git checkout -b bkara1**

Switched to a new branch 'bkara1'

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (bkara1)

**\$ git commit -am "BrancheRevalidation2"**

On branch bkara1  
nothing to commit, working tree clean

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (bkara1)

**\$ git remote add origin https://github.com/ksamb81/Dev20252026.git**

error: remote origin already exists.

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (bkara1)

**\$ git status**

On branch bkara1  
nothing to commit, working tree clean

HP@DESKTOP-Q3TE5TP MINGW64 ~/Desktop/Ecoles (bkara1)

**\$ git push origin bkara1 --force**

Total 0 (delta 0), reused 0 (delta 0), pack-reused 0 (from 0)  
remote:  
remote: Create a pull request for 'bkara1' on GitHub by visiting:  
remote: <https://github.com/ksamb81/Dev20252026/pull/new/bkara1>  
remote:  
To <https://github.com/ksamb81/Dev20252026.git>  
\* [new branch] bkara1 -> bkara1