



# GitHub Cheat Sheet

## Versionner son travail

### Versionner en local

<b>git init</b>	initialise le dépôt (se mettre sur le bon dossier), mieux à faire depuis Github.com
<b>git add .</b>	ajoute toutes les modifications (le . symbolise tout)
<b>git commit -m "explication"</b>	créer un nouveau commit. <b>git add</b> pousse les fichiers en zone d'index, <b>git commit</b> les sauvegarde réellement dans un nouveau commit

### Gérer les commits

<b>git log</b>	liste des commits
<b>git log -n2</b>	affiche les 2 derniers commits
<b>git show sha-1</b>	voir commit spécifique (cliquer molette souris pour coller)
<b>git checkout sha-1</b>	remettre la version du sha-1
<b>git checkout main</b>	remettre version la plus récente

### Versionner sur un dépôt distant

<b>git clone lien-github.com</b>	recupérer travail depuis dépôt distant
<b>git push -u origin main</b>	pousse les modifications vers serveur
<b>git push -f origin main</b>	pousse de force des modifications (à manipuler avec précaution)

### Naviguer dans Git Bash

<b>pwd</b>	savoir dans quel dossier je suis
<b>mkdir "dossier"</b>	créer un dossier (Make Directory)
<b>touch fichier.txt</b>	créer fichier
<b>ls</b>	liste le dossier courant
<b>ls -la</b>	liste tout plus précisément que ls
<b>cd dossier</b>	aller dans le dossier (Change Directory)
<b>cd ..</b>	Remonter d'un dossier

### Initialisation de Git

<b>git config --global user.name</b>	"Mon Nom"
<b>git config --global user.email</b>	mon@mail.com
<b>git config --global --list</b>	Affiche nom et mail

### Autres commandes

<b>git status</b>	état du fichier
<b>git diff</b>	affiche les modifs avant commit

### Commit son projet sur Github

**git add .**  
**git commit -m "message"**  
**git push -u origin main**





# GitHub Cheat Sheet

Travailler en équipe

## Travailler avec un dépôt distant

<b>git pull</b>	recupérer MAJ sur dépôt distant
<b>git pull --rebase</b>	fusionner les branches
<b>git merge</b> nom_branche	fusionner la branche

## Gérer les branches

<b>git branch</b> nom_branche	créer branche
<b>git branch</b>	afficher les branches
<b>git checkout</b> nom_branche	positionne sur la branche à modifier
<b>git branch -d</b> nom_branche	supprimer branche en local (ne pas se mettre sur la branche à supprimer)

## Gérer les branches sur un dépôt distant

<b>git push --set-upstream origin</b> nom_branche	créer la branche en dépôt distant (il faut d'abord faire un " <b>git branch</b> nom_branche" pour la créer en local)
<b>git branch -a</b>	voir liste des branches sur dépôt distant
<b>git push origin --delete</b> nom_branche	supprimer branche sur dépôt distant

## Fichiers indispensables

<b>.gitignore</b>	renseigner le chemin de chaque fichier que l'on ne veut pas commit.  ex : <b>dossier/fichier.txt</b> (ignore fichier.txt)  ex : <b>dossier/*</b> (ignore tous les fichiers à l'intérieur de dossier)
<b>README.md</b>	Renseigner toutes les informations utiles au projet

## Trouver qui a fait quoi

<b>git blame</b> index.html	affiche les mods du fichier <i>index.html</i> ligne par ligne
<b>git blame -L 10,20</b> index.html	affiche les mods de la ligne 10 à 20.  (appuyer sur " <b>q</b> " pour quitter l'affichage)

## Commit son projet sur Github

**git add .**  
**git commit -m "message"**  
**git push -u origin main**

