GitHub Cheat Sheet

|  |  |
| --- | --- |
| Versionner en local | |
| git init | La commande git init crée un nouveau dépôt Git. Elle permet de convertir un projet existant, sans version en un dépôt Git ou d'initialiser un nouveau dépôt vide. |
| git add . | Git add est la première commande d'une chaîne d'opérations qui indique à Git d'« enregistrer » un instantané de l'état actuel du projet dans l'historique des commits. Lorsqu'elle est utilisée seule, git add promeut les changements en attente du répertoire de travail vers la zone de staging. |
| git commit –m « *explication* » | Git commit ouvre l'éditeur de texte configuré en local et invite à saisir un message de commit. Lorsque vous transmettez l'option -m , vous renoncez à l'invite d'éditeur de texte au profit d'un message contextuel. |
| Versionner sur un dépôt distant | |
| git remote add origin *lien-github.com* | Ajouter un nouveau dépôt distant. |
| git push –u origin main | Faire un push d’une branche locale sur un repository distant et l’inspecter. C’est puor le mise à jour. |
| git clone *lien-github.com* | Cibler un dépôt existant et créer un clone ou une copie du dépôt cible. |
| Initialisation de Git | |
| Git congig –global init.defaultBranch main | Permet de changer le nom de dépôt par defaut en main. |
| git config –global user.name | Git vous permet de définir **un nom d’utilisateur** et une adresse e-mail global et par projet. Vous pouvez définir ou modifier votre identité git à l’aide de la commande git config. |
| git config –global user.email | Git vous permet de définir un nom d’utilisateur et **une adresse e-mail global** et par projet. Vous pouvez définir ou modifier votre identité git à l’aide de la commande git config. |
| git config –global --list | Si vous voulez vérifier vos paramètres de configuration, vous pouvez utiliser la commande git config --list pour lister tous les paramètres |
| Gérer les versions | |
| git log | Après avoir créé plusieurs commits ou si vous avez cloné un dépôt ayant un historique de commits, vous souhaitez probablement revoir le fil des évènements. Pour ce faire, la commande git log est l'outil le plus basique et le plus puissant. |
| git log –n 2 | -n <nombre>  Limite le nombre de commits dans la sortie. |