

- 1- **touch script.js** : permet de créer un fichier dont le nom est script et l'extension est js
- 2- **touch index.html style.css** : permet de créer deux fichiers qui sont index.html et style.css
- 3- **git config --global user.name 'Anis'** : permet de configurer le nom de l'utilisateur 'Anis'
- 4- **git config --global user.email 'mohamedrbh519@gmail.com'** : permet de configurer l'email de l'utilisateur
- 5- **git config -l** : affiche la configuration de git
- 6- **clear** : efface tout ce qu'on a écrit
- 7- **git init** : initialiser un repository
- 8- **ls**: permet d'afficher l'ensemble des fichiers et dossiers qui sont présents dans le répertoire
- 9- **ls -a** : permet d'afficher aussi les fichiers/dossiers cachés
- 10- **git status** : permet de consulter l'état du **working directory**
- 11- **git add .** : permet d'ajouter tous les fichiers et dossiers (qui sont dans le working directory) dans le **staging area**
- 12- **git commit -m "ajout code html"** : permet de faire une capture (snapshot ou sauvegarde) avec le message "ajout code html"
- 13- **git checkout 342c3f9c41e0735f8aa019955db82b060c6b0b2d** : aller vers le commit dont l'identifiant est 342c3f9c41e0735f8aa019955db82b060c6b0b2d
- 14- **git checkout master** : aller vers le commit le plus récent
- 15- **git log** : permet de consulter l'historique des commits
- 16- **git log --oneline** : version compacte de l'historique des commits
- 17- **git log --oneline -p index.html** : historique de toutes les modifications par commit liées au fichier index.html
- 18- **git add *** : permet d'ajouter tous les fichiers dans le snapshot de manière récursive
- 19- **git add --all** : équivalent de git add *
- 20- **git add .** : équivalent à git add *
- 21- **git add -A** : équivalent à git add *
- 22- **git add *.php** : tous les fichiers finissant par **php** de manière récursive
- 23- **git add httpdocs/** : permet d'ajouter tous les fichiers se trouvant dans le dossier **httpdocs** ainsi que les fichiers et dossiers créés ou modifiés dans les dossiers enfants de httpdocs
- 24- **git add *.html|php|css|js|xml** : permet d'ajouter tous les fichiers finissant avec ces 5 extensions récursivement

- 25- **git diff index.html** : permet de consulter la difference entre le fichier original et le fichier modifié
- 26- **git checkout {nom fichier}** : permet d'annuler les changements effectués dans le fichier {nom fichier}
- 27- **git branch {nom de la branche}** : créer la branche
- 28- **git branch** : permet d'afficher la liste des branches
- 29- **git checkout css** : aller vers la branche css
- 30- **git checkout master** : aller vers la branche master
- 31- **git merge css** : merger (fusionner) le code css avec le code master
- 32- **git branch -d {nom de la branche}** : permet de supprimer la branche
- 33- **git checkout -b {nom de la branche}** : permet de créer une branche et d'aller vers elle
git checkout -b {nom de la branche} : **git branch {nom de la branche}** + **git checkout {nom de la branche}**

-
- 34- **git reset --soft** : revient à l'ancien commit en conservant les fichiers à l'état unstaged
 - 35- **git reset --mixed** : revient à l'ancien commit en mettant les fichiers à l'état modified/untracked/deleted (mode par défaut)
 - 36- **git reset --hard** : revient à l'ancien commit sans rien conserver
 - 37- **git branch css** : créer la branche css
 - 38- **git checkout css** : basculer vers la branche css
 - 39- **git merge css** : mélange la branche css avec la branche courante
 - 40- **git branch** : affiche la liste des branches
 - 41- **git branch -d css** : supprime la branche css
 - 42- **git remote add origin**
<https://github.com/AnisLcool/testgithub4.git>
 - 43- **git branch -M "main"**
 - 44- **git push -u origin "main"**

- 1- modifier le code local
- 2- sauvegarder les changements
- 3- pousser les changements vers le repo distant