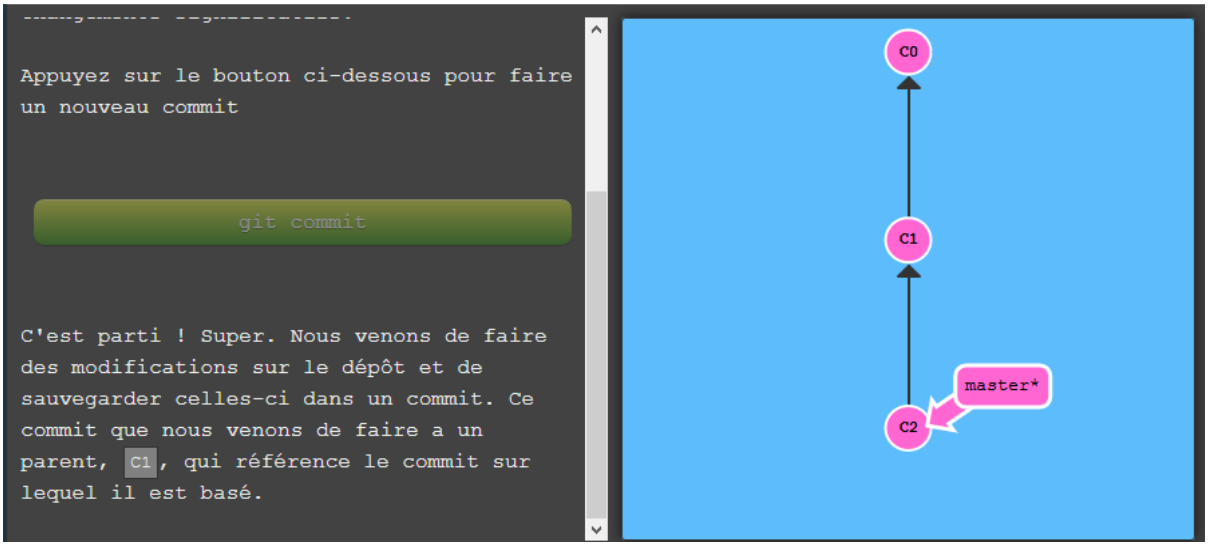
- Commandes GIT -

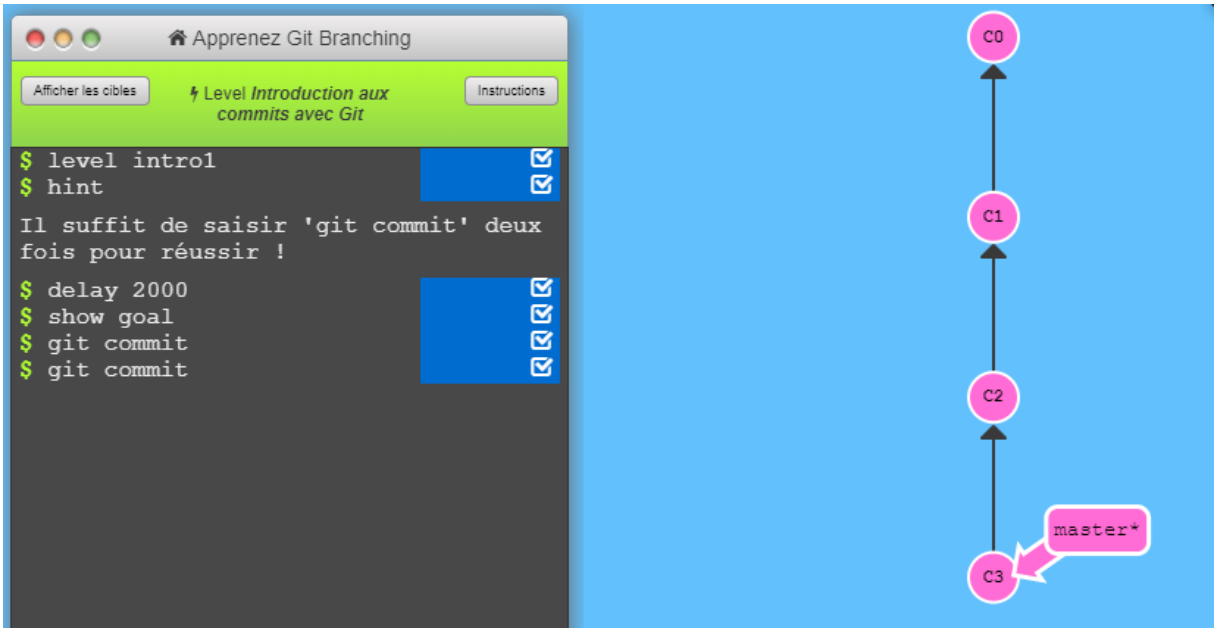
LO43 - Benjamin MONSERAND

Learn Git Branching

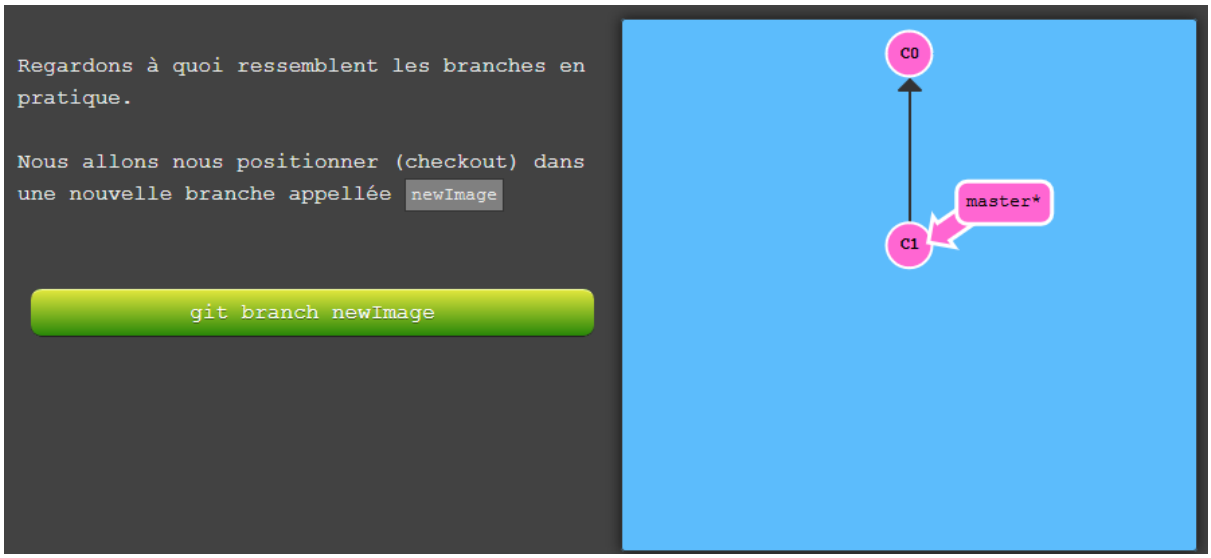
L'outil Learn Git Branching permet d'apprendre et visualiser l'effet des différentes commandes Git.

Git commit :



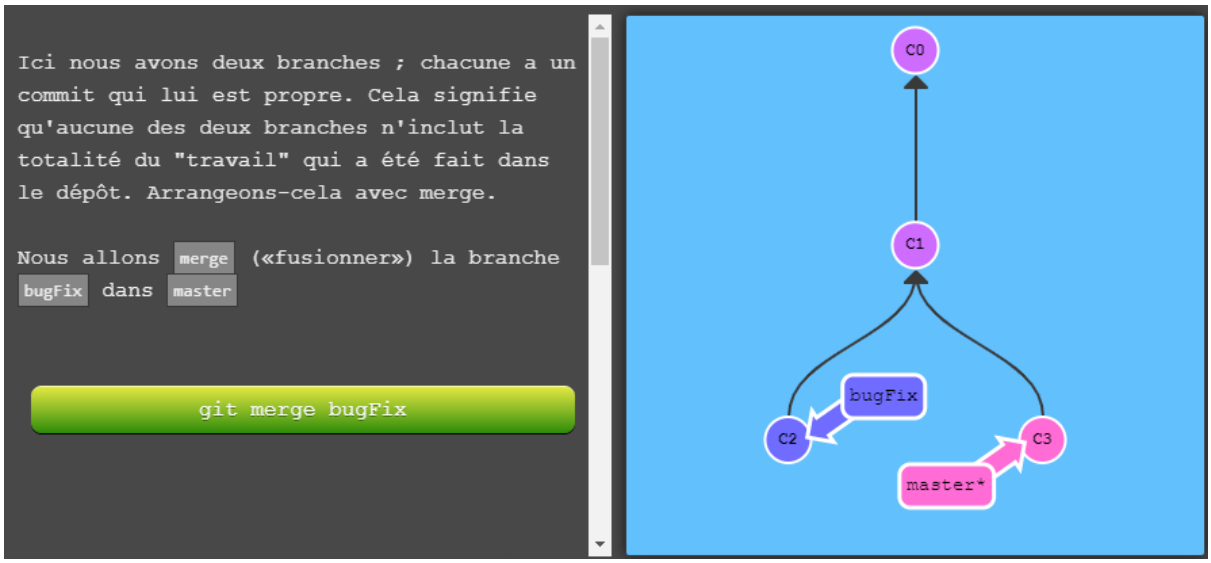


Git branch :

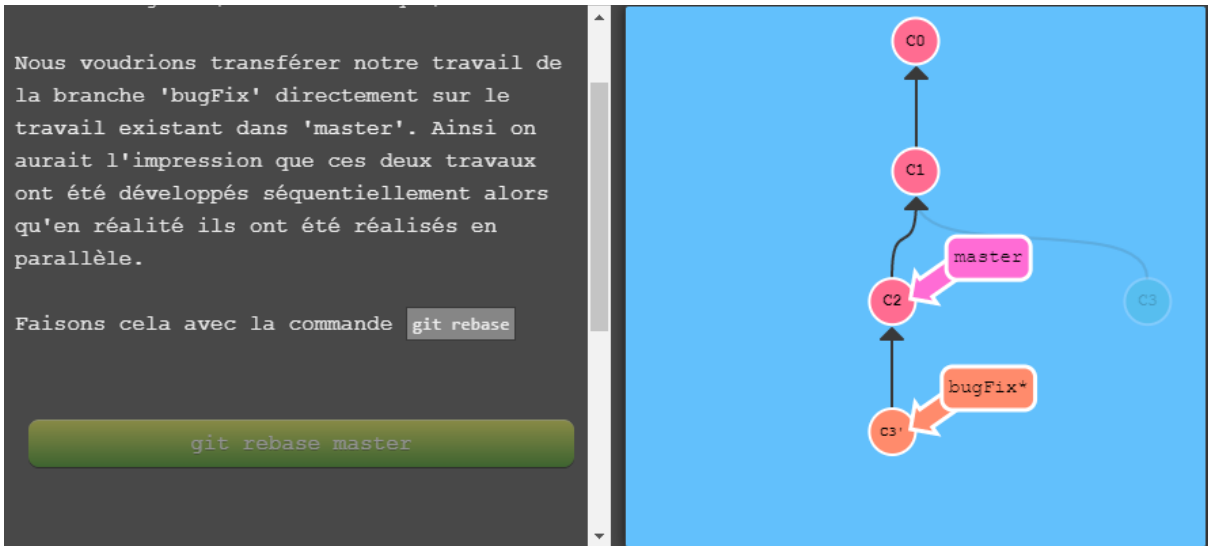

Regardons à quoi ressemblent les branches en pratique.
Nous allons nous positionner (checkout) dans une nouvelle branche appellée newImage
git branch newImage
C0
master*
C1


Indiquons à git que nous voulons nous positionner sur la branche avec
git checkout [nom]
Cela nous positionne sur la nouvelle branche avant de faire un commit avec nos modifications
git checkout newImage; git commit
C0
master*
newImage
C1

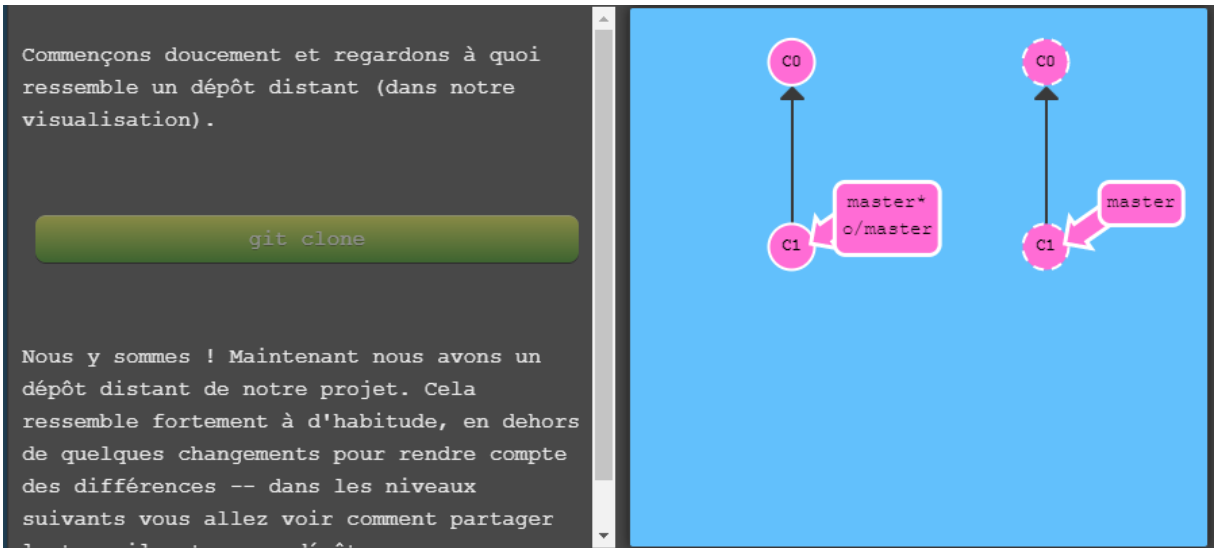
Git merge :

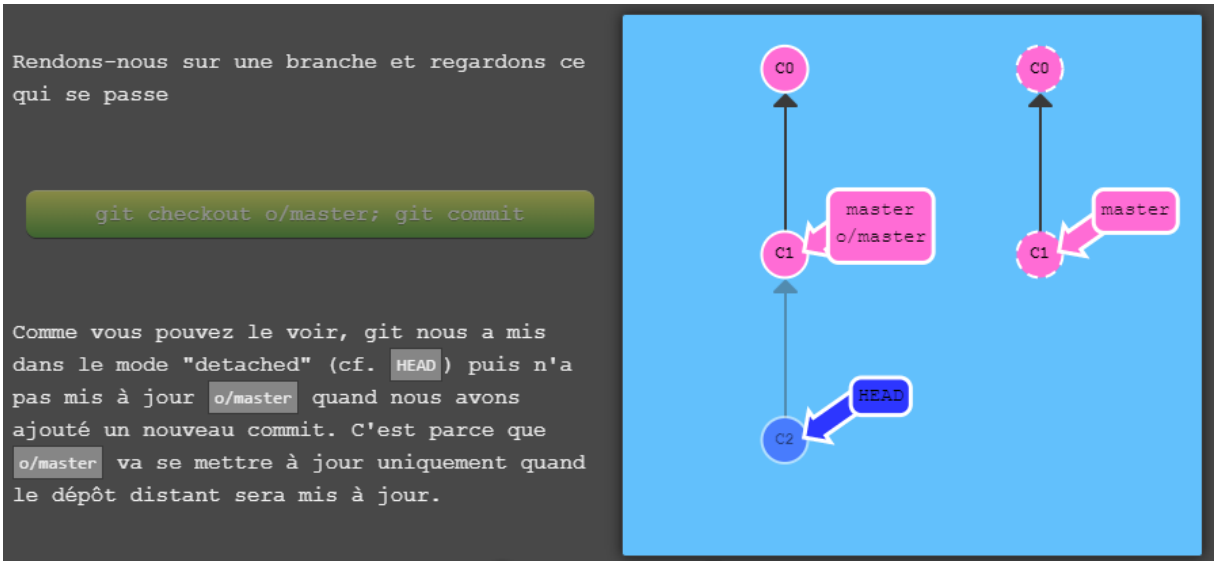


Git rebase :



Git clone :


Commençons doucement et regardons à quoi ressemble un dépôt distant (dans notre visualisation).
git clone
Nous y sommes ! Maintenant nous avons un dépôt distant de notre projet. Cela ressemble fortement à d'habitude, en dehors de quelques changements pour rendre compte des différences -- dans les niveaux suivants vous allez voir comment partager
C0
C1
master*
o/master
C0
C1
master


Rendons-nous sur une branche et regardons ce qui se passe
git checkout o/master; git commit
Comme vous pouvez le voir, git nous a mis dans le mode "detached" (cf. HEAD) puis n'a pas mis à jour o/master quand nous avons ajouté un nouveau commit. C'est parce que o/master va se mettre à jour uniquement quand le dépôt distant sera mis à jour.
C0
C1
master
o/master
C2
HEAD
C0
C1
master

Git pull :

Git Pull

Maintenant que vous avez vu comment rapatrier des données depuis un dépôt distant avec `git fetch`, mettons à jour notre copie de travail pour refléter ces changements !

Il existe en fait beaucoup de façons de faire cela -- une fois que vous avez de nouveaux commits disponibles localement, vous pouvez les incorporer dans votre branche de travail comme s'ils étaient des commits normaux d'autres branches. Cela signifie que pourriez simplement exécuter des commandes comme :

- `git cherry-pick o/master`
- `git rebase o/master`
- `git merge o/master`
- etc., etc.

En fait, le principe de *rapatrier* (fetch) les branches distantes puis les *fusionner* (merge) est si commun que git a en réalité une commande pour faire les deux à la fois ! Cette commande est `git pull`.

Git push :

Git Push

Ok, donc j'ai rapatrié les changements du dépôt distant et je les ai incorporés dans mon travail local. C'est super... mais comment je partage *mon* travail génial avec tous les autres ?

En fait, la manière d'envoyer du travail à partager fonctionne à l'opposé du téléchargement de travail partagé. Et quel est l'opposé de `git pull` (tire) ? `git push` (pousse) !

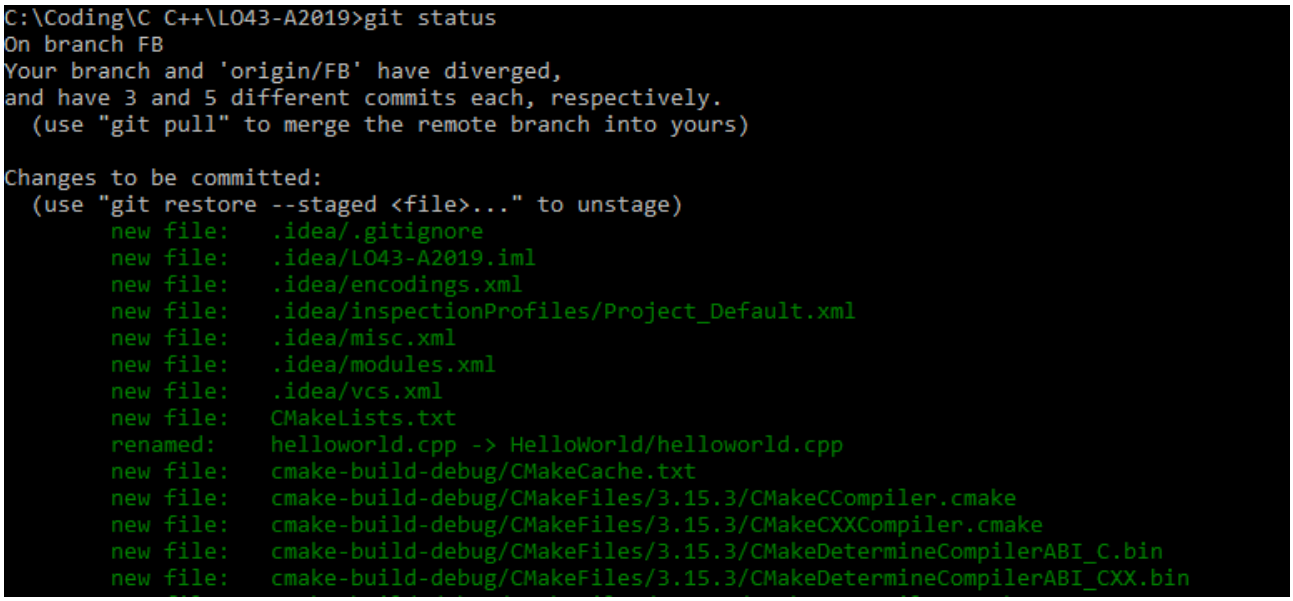
`git push` est responsable de l'envoi de *vos* changements vers un dépôt distant et de la mise à jour de ce dépôt pour incorporer vos commits. Une fois `git push` terminé, tous vos amis peuvent télécharger votre travail depuis le dépôt distant.

Descriptions commandes GIT

Git config :
Git config permet de modifier ou d'accéder aux différents paramètres de git. Plusieurs paramètres peuvent être modifiés comme le nom, l'e-mail, l'éditeur de texte par défaut, l'outil de merge ainsi que l'apparence de git.

Git init :
Git init permet de créer un dépôt git dans le dossier courant. Il créé un dossier .git à l'intérieur des dossiers vers toutes les références, les têtes et les accès à distance. Il crée aussi un fichier HEAD qui renvoie vers la tête de la branche principale master.

Git status :
Git status permet d'afficher le statut des fichier du dépôt courant. L'état est constitué de la liste des fichiers qui ont été modifiés depuis le dernier commit ou le dernier pull.



Git merge :
Git merge permet de combiner deux branches afin de fusionner le contenu des deux parents. Cette commande est essentielle pour intégrer de nouvelles fonctionnalités venant de deux branches parallèles.

Git diff :
Git diff permet de voir la différence de contenu entre la version locale et le dernier commit. On peut y voir les fichiers modifiés, les ajouts et suppressions, ainsi que le nombre total de modifications.