

EXPLICATIONS GIT

EXPLICATIONS GIT	1
1 - Qu'est-ce qu'un commit ?	1
2 - À quoi sert la commande git log ?	1
3 - Qu'est-ce qu'une branche ?	2
4 – Aller plus loin	2

1 - Qu'est-ce qu'un commit ?

Un commit c'est comme une photo de l'état de ton projet. Quand tu as fini de modifier une partie de ton projet (correction d'une fonction par exemple), tu fais une 'photo' (un commit) qui garde une image de l'état actuel du projet. N'oublie pas d'indiquer un message explicite qui correspond à tes modifications ("Correction de la fonction 'nom_de_la_fonction'" par exemple). C'est important pour t'y retrouver plus tard et/ou travailler à plusieurs sur le projet.

Ensuite, tu peux continuer à travailler sur ton projet et consulter à tout moment les différents commits ou revenir sur l'un d'eux. C'est très pratique pour organiser ton travail, revenir facilement en arrière si besoin et pour travailler à plusieurs. Fais autant de commits que nécessaire mais pas pour rien histoire de ne pas alourdir la liste de commits inutilement. J'insiste sur le fait de pouvoir s'y retrouver plus tard. C'est comme quand tu as appris à respecter des conventions d'écriture dans tes programmes et à les commenter clairement.

2 - À quoi sert la commande git log ?

La commande 'git log' sert simplement à afficher la liste de tous les commits effectués sur la branche active de ton projet (je reviens sur les branches après puisque c'est ta dernière question).

Voici un extrait de log ci-dessous :

```
commit 002f08ff9cf0a6e9e420e9b4b6d243daf533f29 (HEAD -> master, origin/master)
Author: Dheryo <33827012+Dheryo@users.noreply.github.com>
Date: Sat Dec 2 16:09:40 2017 +0100

    Améliorations mineures de la mise en page

commit e16c71e9ee0fae701f9af336928e70a08cd36feb
Author: Dheryo <33827012+Dheryo@users.noreply.github.com>
Date: Sat Dec 2 16:05:02 2017 +0100

    Ajout de README.txt

commit 81dd01134907abe2fd008b959de0f406d6cef64f
Merge: 1ba8c2c 865853b
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:37:29 2017 +0100

    Merge de la branche 'aller_plus_loin'
```

Les commits sont affichés du plus récent au plus vieux avec comme informations :

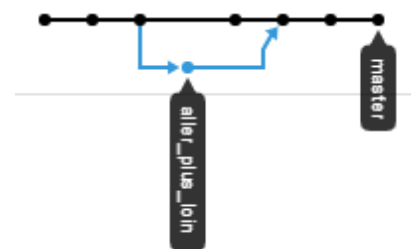
- le SHA (son identifiant unique qui te sert notamment à revenir sur ce commit),
- le nom de l'auteur,
- la date et l'heure du commit,
- le fameux message de description du commit qui doit être le plus clair possible,
- des informations entre parenthèses liées à la synchronisation du projet local et du repository, je ne détaillerai pas.

3 - Qu'est-ce qu'une branche ?

Il me semble qu'un exemple d'utilisation des branches sera plus parlant. Admettons que tu développes un jeu solo. En cours de développement, tu te dis qu'il faudrait essayer d'ajouter un mode multi. Plutôt que travailler sur la même version du projet, tu vas créer une nouvelle branche. A partir de là, les commits que tu vas faire sur la branche sobrement nommée "multi" et les commits que tu feras sur la branche par défaut qui s'appelle "master" seront indépendants.

Tu auras donc un projet avec deux versions parallèles comprenant les commits effectués avant la création de la branche "multi" suivis de commits indépendants. Ensuite, tu pourras 'merger' les deux branches (c'est-à-dire les fusionner) ou la supprimer si ton mode multi ne s'intègre pas dans ton jeu ou encore laisser la branche "multi" de côté en attendant de décider si elle devra être intégrée ou non.

Ci-contre une représentation de projet avec deux branches. Elle a été générée sur GitHub et reprend la branche principale « master » en noir et une seconde branche « aller_plus_loin » en bleu.



On peut ainsi corréler les deux images entre elles de la façon suivante :

```
commit 002f08ff9cf0a6e9e420e9b4b6d243dfaf533f29 (HEAD -> master, origin/master)
Author: Dheryo <33827012+Dheryo@users.noreply.github.com>
Date: Sat Dec 2 16:09:40 2017 +0100
    Améliorations mineures de la mise en page

commit e16c71e9ee0fae701f9af336928e70a08cd36feb
Author: Dheryo <33827012+Dheryo@users.noreply.github.com>
Date: Sat Dec 2 16:05:02 2017 +0100
    Ajout de README.txt

commit 81dd01134907abe2fd008b959de0f406d6cef64f
Merge: 1ba8c2c 865853b
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:37:29 2017 +0100
    Merge de la branche 'aller_plus_loin'

commit 1ba8c2cd6cafaca241eb4cef2a99eb17b28254
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:22:58 2017 +0100
    Ajout de la description des branches

commit 865853b3c9386b25d4aeb23688aeb94e3727629c (origin/aller_plus_loin)
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:20:10 2017 +0100
    Ajout d'informations de sources pour approfondir le sujet

commit d9993f64d2cfe6051b94d4aa69524a8f8d33f7f1
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:11:54 2017 +0100
    Ajout de la description de la commande 'git log'

commit d332e75c94202b975c6eb20ee72048a3a90b8ae2
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:10:36 2017 +0100
    Ajout de la description d'un commit

commit e465e8e055cf64bff314af3b0a5d6b39c16effc0
Author: Dheryo <djdesiderio@gmail.com>
Date: Sat Dec 2 15:05:15 2017 +0100
    Création du fichier explications.txt indiquant les questions à aborder.
```

On voit ainsi :

- que chaque point est un log,
- qu'après le 3^{ème} commit, a été créée une nouvelle branche nommée « aller_plus_loin »,
- que le 4^{ème} commit du log a été fait dans cette branche,
- que le commit suivant a été effectué sur la branche « master ».

A ce stade, chaque branche contient 4 commits, mais le dernier commit de chaque branche est différent,

- que le commit d'après consiste à ré-unifier les deux branches, ce qu'on appelle un « merge »

4 – Aller plus loin

Tu peux entre autre, consulter [la documentation officielle](#), le [cours Openclassrooms](#) ou aller sur [le repository GitHub](#) que j'ai créé pour l'occasion. En espérant avoir été suffisamment clair. Bonne continuation !