


Feuille de travaux pratiques n° 2

Découverte de `git`



Ce TP a pour objectif de vous faire découvrir le fonctionnement de base de `git` et de GitHub, ainsi que le fonctionnement des fichiers `markdown`. Le dépôt que vous allez créer va synthétiser des informations sur les objets célestes du système solaire, rangés par catégorie.

1 Préparatifs

Avant de vous lancer dans le TP à proprement parler, il vous faut

- un compte **GitHub** ,
- avoir installé `git` sur votre machine, déjà présent sur les ordinateurs en salle machine,
- avoir un éditeur de textes (par exemple VSCode),
- vous mettre d'accord pour travailler avec une autre personne à certains moments du TP,
- avoir créé une clé ssh.

2 Création

1. Configurez le nom d'utilisatrice ou d'utilisateur, l'email et l'éditeur pour `git`.
2. Ajoutez votre clé ssh publique à votre compte **GitHub** (dans **Settings**, **SSH and GPG keys**).
3.  Créez un dépôt sur **GitHub** avec votre compte, qui s'appelle par exemple `corps-celestes`. Mettez-vous d'accord avec votre binôme pour choisir des noms de dépôts différents.
4. Pour les réglages, mettez votre dépôt en public, choisissez la licence que vous préférez, ajoutez un fichier `.gitignore` automatique.
5. Clonez le dépôt sur votre machine (pour récupérer l'adresse,  cliquez sur **Code**, puis **SSH**).
6. Quels sont les fichiers présents ?
7. Affichez le contenu du fichier `.gitignore` dans votre éditeur. Modifiez-le pour que les fichiers pdf et html soient ignorés.

3 Premières modifications

8. Créez les répertoires `etoiles`, `planetes` et autres sur votre machine dans le répertoire `corps-celestes`.
9. Récupérez les fichiers `mercure.md` et `mercure.png` du dépôt GitHub <https://github.com/mpelleau/corps-celestes> et placez-les dans le bon répertoire.
10. Ajoutez ces fichiers à votre dépôt et poussez les modifications.
11. En vous inspirant de `mercure.md`, créez et placez correctement le fichier `soleil.md` et l'image correspondante.
12. Ajoutez ces fichiers à votre dépôt et poussez les modifications.

4 Les issues / tags

13. 🛠 Ajoutez une issue pour créer toutes les planètes du système solaire. On utilisera une liste de tâches (<https://git.goffinet.org/06-github.html#listes-de-t%C3%A2ches>)
14. 🛠 Ajoutez une issue pour créer tous les satellites de Saturne.
15. 🛠 Ajoutez une issue pour créer tous les satellites de la Terre.
16. 🛠 Créez un Milestone qui regroupe les issues précédentes.
17. Créez la Lune et le commit qui ferme l'issue associée.
18. Créez un tag, observez la release correspondante sur GitHub.
19. Créez `pluton.md` dans les planètes.
20. Ajoutez ces fichiers à votre dépôt et poussez les modifications.
21. 🛠 Ajoutez une issue pour signaler que Pluton n'est pas une planète.
22. Déplacez `pluton.md` dans `autres`, créez un commit qui ferme l'issue associée et poussez les modifications.
23. Créez l'objet **Voyager 1**. Voir https://fr.wikipedia.org/wiki/Voyager_1
24. **Voyager 1** est sorti du système solaire, supprimez le fichier du dépôt `git`, mais le conserver en local sur votre ordinateur (voir `git rm --help` pour trouver le bon paramètre).

5 Collaboration

25. 🛠 Nommez votre binôme comme collaboratrice ou collaborateur sur votre dépôt.
26. Choisissez un des deux dépôts pour la suite.
27. 🛠 Assignez-vous les issues qui restent dans l'interface des issues de GitHub.
28. L'un de vous ajoute **la température de surface** pour chacun des objets, l'autre **la période de rotation**.
29. La première ou le premier de vous qui a fini pousse ses modifications.
30. La seconde ou le second devra résoudre les conflits avant de pousser ses modifications.

6 Pull-request

31. 🛠 Allez voir un autre dépôt et repérez une erreur ou une fonctionnalité manquante.
32. 🛠 Forkez le projet
33. Implémentez la correction / amélioration.
34. 🛠 Faites une pull-request.

7 Pour aller plus loin : conversion en html et pdf

35. S'il n'est pas déjà installé, installez le paquet `pandoc`.
36. À l'aide de la commande `pandoc` avec les bons paramètres, créez les fichiers `mercure.pdf` et `mercure.html` à partir de `mercure.md`.
37. Quel est le résultat de `git add mercure.html` ? Donnez le bon paramètre à cette commande pour ajouter le fichier `mercure.html` malgré le `.gitignore`.
38. Modifiez votre fichier `.gitignore` pour que
 - les fichiers pdf soient ignorés en général,
 - mais le fichier `mercure.pdf` puisse être ajouté par `git add mercure.pdf`.
39. À l'aide de la commande `pandoc` avec les bons paramètres, générez le fichier `corps-celestes.pdf` à partir de tous les fichiers **markdown**.