

# GIT Begins

Push it to the limit

## Plongé dans l'univers de GIT

---

Git est un système de contrôle de version très utilisé dans le développement. Il offre un cadre structuré pour la gestion et le suivi des modifications apportées au code source, ce qui permet aux équipes de travailler de manière efficace et organisée. Git permet la fusion des fichiers en évitant les conflits autant que possible.

*Imaginez Git comme votre GPS du développement, vous guidant dans l'univers complexe du code. Il vous indique où vous vous trouvez dans l'historique des modifications, vous permet de revenir à des états antérieurs si nécessaire, et vous aide à corriger le cap si vous prenez une mauvaise direction.*

En somme, Git est un outil essentiel qui assure un contrôle rigoureux du code tout en facilitant une collaboration harmonieuse entre les membres de l'équipe.

## Contexte

---



Vous avez réalisé cette semaine le projet SHELL, il est maintenant temps de recevoir vos premiers points de compétences en système. yeahhh 🤙 !

Afin de vous évaluer, l'équipe pédagogique a besoin d'analyser votre travail et donc vous devez le partager. Pour cela, vous allez découvrir un outil très répandu dans le monde du développement informatique : **GIT**

## Job 01

---

Avant tout, il faut installer **GIT** sur votre machine. Ouvrir un terminal et écrire :

```
sudo apt install git-all
```

Une fois l'outil installé, créez un compte sur [github](https://github.com) avec **votre adresse la plateforme !**

Il vous faut choisir obligatoirement un **username** au format **prenom-nom** (votre prénom, un tiret et votre nom, le tout en minuscule !). Respecter le format est indispensable pour que l'équipe pédagogique puisse récupérer votre travail.

Avec votre adresse mail la plateforme, vous disposez aussi du compte premium **github éducation**.

Une fois connecté, demandez à avoir un upgrade vers [github pour education](https://github.com/education).

La prochaine étape est de communiquer via le chat à **Deepthought**, envoyez-lui un lien vers votre **compte GitHub**. Laissez-lui quelques instants, il va vous envoyer, au plus vite, une demande par mail vous permettant de rejoindre l'organisation GitHub de La

Plateforme (<https://github.laplateforme.io>). En attendant que Deepthought vous réponde, vous pouvez continuer votre travail.

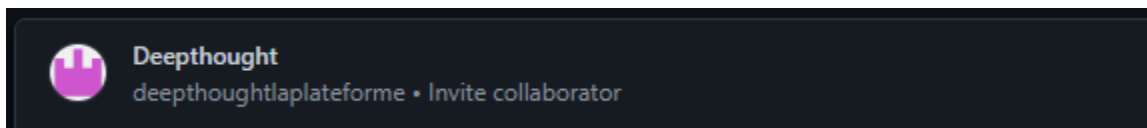
## Job 02

---

Dans ce job, vous allez rendre votre premier travail. Pour cela :

Sur GitHub :

- **Créer** un repository **public** nommé **shell**
- **Donner les droits** à deepthoughtlaplateforme (Attention de donner les droits au bon)



Sur votre terminal en utilisant les lignes de commandes :

- **Cloner** le repository **shell**
- **Ajouter** le travail du projet réalisé à l'intérieur du repository **shell** (la documentation du sujet shell + ghost in the shell)
- **Pusher** le nouveau contenu de votre repository. *(On prend les bonnes habitudes dès le début, il faut mettre un message de commit explicite, ex: "Shell project report")*

**Félicitations ! Vous venez de rendre votre premier projet avec GIT.**

## Job 03

---

De retour sur github **créer** un nouveau repository nommé **git-begins**, de la même manière **cloner** ce dernier via le terminal.

- Dans le terminal et dans le repository cloné, **créer** un fichier nommé **"commands-git"**, dans ce fichier, **écrire** les commandes git qui permettent de :



- **Cloner un repository**
- **Les 3 commandes de push**

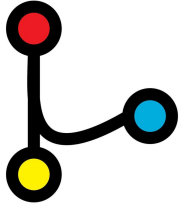
Réalisez un **push**.

Sur github, **modifier** le contenu du fichier **"commands-git"** en y ajoutant la commande qui permet de **pull**.

De retour sur votre terminal, réalisez la commande de **pull** sur votre repository **git-begins**. Vous devriez y voir dans le fichier **"commands-git"** les modifications effectuées sur github.

## Pour aller plus loin...

---



Dans le repository **git-begins**, sur le terminal et toujours en ligne de commande, **créer** une branche "**add-create-branch-command**".  
**Basculer** sur cette branche.

**Ajouter** dans le fichier "**commands-git**" la commande permettant de créer une branche ainsi que celle qui permet de **switcher** de branche.

Réalisez un **push**.

## Rendu

---

Le projet est à rendre sur <https://github.com/prenom-nom/git-begins>

Pensez à mettre votre repository en **public** !

## Compétences visées

---

- Git

## Base de connaissances

---

- [Site officiel de Github](#)
- [Wiki GIT](#)
- [Site de GIT](#)