- Initialisation d'un nouveau répertoire git

- Création d'une arborescence :

- src :

- core :

- Main.java

- Application.xml

- test :

- build.gradle (fichier)

- Sauvegarde des modifications

- Création d'un fichier de log sauvegarder à la racine dans .gitlog

- Ajout d'une classe User dans un package "entities" dans le package "core" :

- Un User possède un nom, un prénom, un login et un mot de passe

- On générera les getter et setter

- Sauvegarde des modifications

- Dans Application.xml vous renseignerez votre nom et votre prénom sous la forme "Copyright\_Lastname\_Firstname"

- Sauvegarde des modifications

- Afin de développer l'application votre chef de projet vous fait travailler en équipe

- Créer un nouveau répertoire github

- Synchroniser votre répertoire local avec le répertoire distant

- Créer une nouvelle branche "dev" issue de la branche "master"

- Créer une nouvelle branche avec la nomenclature "nomPrenom"

- Vous travaillerez sur cette dernière branche

- Créez deux constructeurs dans la classe "User" un vide et l'autre utilisant tout les champs

- Sauvegarde des modifications

- Mergez vos modifications avec la branche "dev"

- Votre nouveau collègue arrive dans l'équipe, il doit récupérer le projet et se créer une branche de développement avec la nomenclature "nomPrenom" issue de la branche "master"

- Votre collègue doit créer un constructeur vide dans la classe "User" qui initialise une "arraylist" de "Droits" et créer la dite classe dans "entities"

- En même temps vous créez un "point d'entré de l'application" dans le fichier "Main.java"

- Sauvegarde des modifications pour vous et votre collègue sur vos branches respectives

- Mergez vos branches sur la branche "dev"

- Votre chef de projet vous indique que le fichier .gitlog doit être enlevé, vous devez donc "revert" le commit associer à se fichier

- Mergez la branche "dev" sur la branche "master"

- Votre collègue doit maintenant créer des attributs avec des getter et des setter dans la classe "Droits" sur sa branche de développement un droit possède les attributs suivants :

- niveau

- sévérité

- nom

- société

- Vous vous appercevez d'un bug dans le fichier Application.xml, directement dans la branche "master" effectuez la modification suivante :

- remplacer la ligne "Copyright\_Lastname\_Firstname" par "Copyright\_Super\_Projet"

- rajouter en dessous de cette ligne les nom et prénom des membres du groupes

- Vous sauvegardez vos modifications et pusher la branche "master"

- Votre collègue doit maintenant sauvegarder ses modifications merger sa branche sur la branche "dev" puis merger la branche "dev" sur la branche "master"

- Vous devez maintenant générer des fichiers de log git ".gitlog", votre manager s'étend trompé :

- Créer pour chacun un fichier de log pour votre branche personnelle avec la nomenclature "branchName.gitlog"

- Vous devez tout deux merger vos fichiers de log pour qu'ils soient présent sur la branche "master"

- Votre manager souhaiterai maintenant que vous créiez un fichier de configuration pour le projet :

- Vous devez créer un fichier "conf.prog", contenant un flux JSON avec 1 jeu de clefs valeurs sous forme de chaine, attention les valeurs doivent être vide dans un premier temps, insérer les champs ("version":"","db":"","conf-version":"") vous devez sauvegarder vos modifications et merger votre fichier jusqu'à la branche "master"

- Votre collègue vas créer un fichier ".gitignore" sur ça branche de développement avec dans le fichier l'exclusion du fichier "conf.prog" votre collègue doit sauvegarder ses modifications et merger son fichier jusqu'à la branche "master"

- Vous et votre collègue devez récupérer les fichiers ".gitignore" et "conf.prog" sur vos branche de développement respectif

- Vous et votre collègue devez sauvegarder vos branches de développement sur le répertoire distant

- Assurez vous que toutes les branches soient envoyé sur le répertoire distant et envoyez un email à antoine.cronier@tactfactory.com avec l'adresse de votre répertoire