**Exercice sur GitHub** **et Git**

**Partie 3 :**

Réponse aux questions :

* Qu'est-ce que GitHub ? : GitHub est un service d'hébergement Web de dépôt Git qui offre toutes les fonctionnalités de Git ainsi que l'ajout de plusieurs de ses propres fonctionnalités.
* Quand a-t-elle été créée ? : GitHub a été lancé le 10 avril 2008 et le développement a commencé le 19 octobre 2007.
* Pourquoi ? : Parce qu’en plus d'offrir l'hébergement de projets avec Git, GitHub repose sur le principe du fork (embranchement) par défaut : toute personne « forkant » le projet devient publiquement de facto le leader de son projet portant le même nom que l’original.
* Par qui ? : Ce site est développé par Chris Wanstrath, PJ Hyett et Tom Preston-Werner.
* Quelles sont les plates-formes similaires ? : GitLab, SourceForge, Cloud Source Repositories, GitKraken, Apache Allura.
* Pourquoi utiliser une telle plate-forme ? : Il permet de mettre en avant la qualité de son code et ainsi montrer ses capacités et sa plus-value lorsque l'on recherche un emploi. Considéré comme un véritable réseau social, il permet aussi aux développeurs de contribuer à des projets open source et l’aspect backup régulier du code, le versioning, le rend plus intéressant concernant le développement web.

**Partie 4 :**

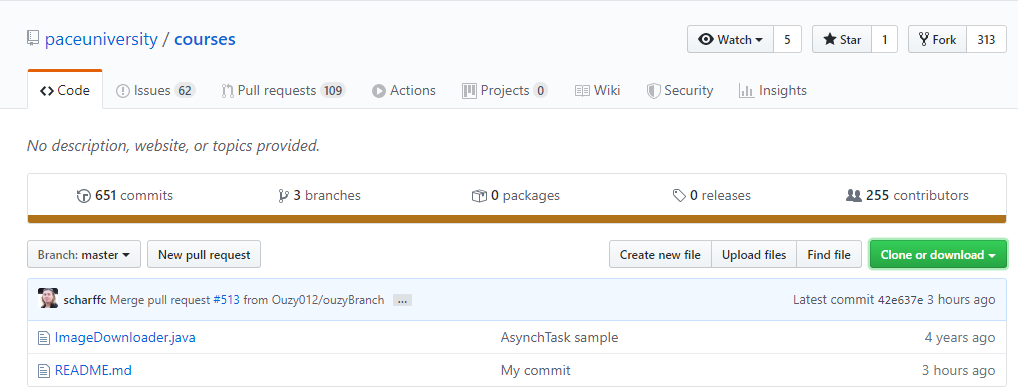
Définition des termes suivants dans le contexte de Git

* Repository : C’est simplement un emplacement de fichier ou vous stockez tous les fichiers liés à votre projet.
* Commit : C’est une modification individuelle d’un fichier ou d’un ensemble de fichiers et vous pouvez ramener le code à n'importe quelle version en un seul clic.
* Push : permet de télécharger du contenu d’un dépôt local sur un référentiel distant, c’est la façon dont vous transférez les validations de votre référentiel local vers un dépôt distant.
* Branch : C’est Une référence symbolique de branche git à un seul commit git et est une version parallèle du code qui peut être modifiée sans affecter les autres versions.
* Fork : C’est une copie d'un dépôt qui permet d'expérimenter librement avec les changements sans affecter le projet original.
* Merge : il permet de reconstituer une histoire fourchue donc de prendre les lignes de développement indépendantes crées et de les intégrer dans une seule branche
* Clone : il permet de cibler un référentiel existant et créer un clone, ou une copie du référentiel cible.
* Pull : il permet de mettre à jour la version locale d'un référentiel à distance.
* Pull request : permet d'informer les autres sur les modifications que vous avez appliquées à une branche d'un référentiel sur GitHub.

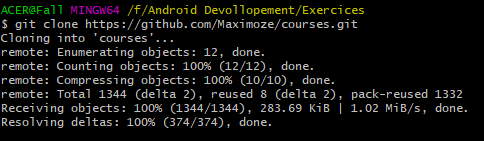
**Partie 6 :**

Listes des commandes et la stratégie utilisés :

* On commence par faire un fork du repository :



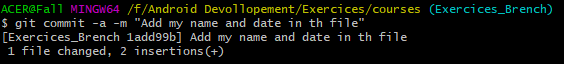
* Ensuite on clone le repository comme suit :



* Puis on crée une nouvelle branch



* On se déplace vers le branch
* On effectue un nano sur le fichier Readme.md pour ajouter les renseignements
* Puis on effectue un commit sur le fichier



* Enfin on réalise le push afin de validez les modifications

