

Git commandes

- `git clone` -> Cloner un dépôt git existant depuis un serveur distant
- `git pull` -> Récupérer les modifications les plus récentes du dépôt distant
- `git add` -> Ajouter des fichiers modifiés à la zone de staging, préparant ainsi les changements pour le prochain commit
- `git commit` -> Enregistrer les fichiers dans la zone de staging avec un message de commit.
- `git push` -> Envoyer les commits locaux vers le dépôt distant.

JSON

JSON (JavaScript Object Notation) est un format simple basé sur des paires clé-valeur pour échanger des données. Il supporte uniquement les valeurs primitives : null, true, false, les entiers, les nombres à virgule (doubles), et les chaînes de caractères.

Java Beans

- **Java Bean** est une convention permettant d'interagir à l'exécution avec une instance de classe considérée comme un conteneur de propriétés.
- La classe ***Introspector*** retourne un **BeanInfo** décrivant la classe passée en paramètre en tant que Java Bean.

Propriétés d'un Java Bean

- Une **propriété** est définie par la présence d'une méthode commençant par *get* ou *is* (getter) et/ou une méthode commençant par *set* (setter).
- La méthode *getPropertyDescriptors()* permet de récupérer un tableau d'objets *PropertyDescriptor* décrivant les propriétés accessibles d'une classe Java

Problème de performance avec BeanInfo

- **Problème** : Actuellement, le **BeanInfo** et les **propriétés** d'une classe sont recalculés à chaque fois, même si elles ne changent jamais.
- Cela entraîne une perte de performances car les propriétés d'une classe sont toujours les mêmes.

ClassValue

- **Solution** : Utiliser un **ClassValue** pour **mettre en cache** un tableau de propriétés pour chaque classe.
- **ClassValue** permet de stocker et de réutiliser des informations calculées pour une classe tout en garantissant que les classes peuvent être déchargées de la mémoire si nécessaire.

Conclusion

- Il est possible de récupérer les attributs d'une classe avec Java Beans en utilisant des methodes comme *getPropertyDescriptors()*.
- Afin de pallier les problèmes de performance liés à *BeanInfo*, on peut utiliser *ClassValue* pour mettre en place un cache et optimiser la récupération des données.