

L'API METHODHANDLES

- Contenue dans le package `java.lang.invoke` et introduite en JAVA 7.
- Une référence typée directement exécutable à une méthode, un constructeur, un champ ...
- Mécanisme qui permet de rechercher, d'adapter et d'invoquer des méthodes.

CONSTRUCTION D'UN METHODHANDLES

La construction d'un MethodHandle se fait en 4 étapes :

- Création de la recherche (Lookup).
- Création du type de la méthode (MethodType).
- Trouver le MethodHandle.
- Invoquer la méthode.

CONSTRUCTION D'UN LOOKUP

Un objet `MethodHandles.lookup` contient les droits associés à une classe.

- `MethodHandles.lookup()` crée un objet `Lookup` avec les droits de la classe.
- `MethodHandles.publicLookup()` crée un objet qui n'a accès qu'aux méthodes publiques.
- `MethodHandles.privateLookupIn(...)` crée un objet qui a accès à toutes les méthodes d'une classe.

CREATION D'UN METHODTYPE

Afin de créer un MethodHandle, il faut fournir à l'objet une définition de son type qui est obtenu grâce à la classe MethodType.

- La création d'un MethodType est faite à partir de la méthode `MethodType.methodType(...)`.
implémenté dans MethodType
- Il faut alors fournir en paramètre, le type de retour de la méthode ainsi que tous ses paramètres.

TROUVER LE METHODHANDLE

L'avant dernière étape est de trouver la méthode que nous souhaitons invoquer. Pour ce faire, il faut utiliser les méthodes suivantes :

- `Lookup.findVirtual()` est utilisé pour les méthodes d'objets.
- `Lookup.findStatic()` est utilisé pour les fonctions statiques.
- `Lookup.findConstructor()` permet de créer une méthode se comportant comme le constructeur d'une classe.

INVOQUER LE METHODHANDLES

3 méthodes sont à notre disposition pour invoquer une méthode :

- `MethodHandle.invoke()`, le nombre d'argument est imposé. Le transtypage et le unboxing sont autorisés.
- `MethodHandle.invokeWithArugments()` est l'appel le moins restrictif.
- `MethodHandle.invokeExact()` est l'appel le plus restrictif. Le nombre d'arguments ainsi que les types sont imposés.

QUELQUES METHODES FONCTIONNELLES DE L'API

- `dropArguments` : cette méthode permet de supprimer un argument.
- `insertArguments` : cette méthode permet d'ajouter un argument.
- `guardWithTest` : cette méthode prend 3 `MethodHandle` en paramètre et permet de faire un `if..else...`