

Ilyass Farsal

Amine Sidotmane

Rapport du 1^{er} Devoir

Problèmes rencontrés et solutions :

- **Supprimer les animaux d'une cage :**

Pour supprimer les animaux d'une cage #on était obligé de créer une autre méthode « AnimalOthers » sur la classe MyServiceTp.java, qui sera appeler par l'url : /animals/byCage/{param} avec méthode HTTP Delete.

- **Ajout d'un animal dans une cage vide :**

Celle-ci retourne NULL pour la méthode getResidents();
Implémenter la methode addAnimal(Cage,Animal) dans la classe "Center" qui permet de vérifier si la cage et vide ou pas avant d'ajouter l'animal.

- **Trouver une cage proche d'une position donnée :**

Implémenter la méthode near(Position) pour comparer les position à un degrés près (Ex: 1.23556 est proche de 0.6698 et 1. 965613...)

- **Ajouter un animal dans une cage pleine :**

mettre à jour la méthode addAnimal() ajoutée au model "Center" pour vérifier la capacité d'une cage avant d'ajouter l'animal si elle est pleine on retourne un message d'information et l'animal n'est pas ajouté, et grâce à un attribut "Full" ajouté au model "Cage" l'appel à cette méthode avec le même cage ne se fera plus tant que la cage est pleine.

- **Affichage de plusieurs animaux sur la console :**

vu que le System.out écrit tout le résultat sur une seule ligne, cela à générer un blocage lors de l'affichage de plusieurs animaux, on a modifié la méthode printSource() pour indenté le résultat et pouvoir afficher autant d'animaux que possible.

- **Ajouter un animal dans une cage :**

#Créer d'abord une méthode add_cage(Cage cage) sur la classe MyClient qui fait appel au service par l'url « /cages" et par la méthode "POST".

Sur le service on a du gérer cette appel en implémentant une méthode CagesCrud pour gérer l'ajout d'une nouvelle cage et y ajouter un animal par la suite.

- **Rechercher un animal dont le nom contient des « espaces » :**

La recherche d'un animal dont le nom ne contient aucun espace, marche sans problème, par contre l'url ne peut pas avoir des espaces. Nous n'avons pas eu le temps de penser à une autre alternative.

Améliorations :

- ✓ Ajout d'une méthode « init » sur le fichier MyClient.java, qui prend 2 paramètres : l'url et la méthode http. Le but de cette méthode est d'alléger le code en évitant de réécrire à chaque fois les mêmes instructions pour chaque appel de méthode client.
- ✓ Améliorer la méthode d'affichage qui permet désormais de structurer les balises avant de les afficher.
- ✓ Gérer l'encodage au niveau de la fonction d'affichage.
- ✓ Présenter les scénarios sous forme d'un menu qui permet à l'utilisateur d'entrer le nombre qui correspond au scénario qu'il souhaite exécuter.
- ✓ Prise en compte de la capacité de la cage avant d'y ajouter des animaux.