



Cégep **André-Laurendeau**

420-235-AL (Hiver 2024)

Travail pratique 2

Structures de données.

Échéance : 10 avril à minuit

David Giasson et Michel Généreux

Objectif

Coder et manipuler diverses structures de données adaptées aux besoins d'une application logicielle, tout en appliquant correctement les pratiques clés de la programmation orientée-objet.

Directives

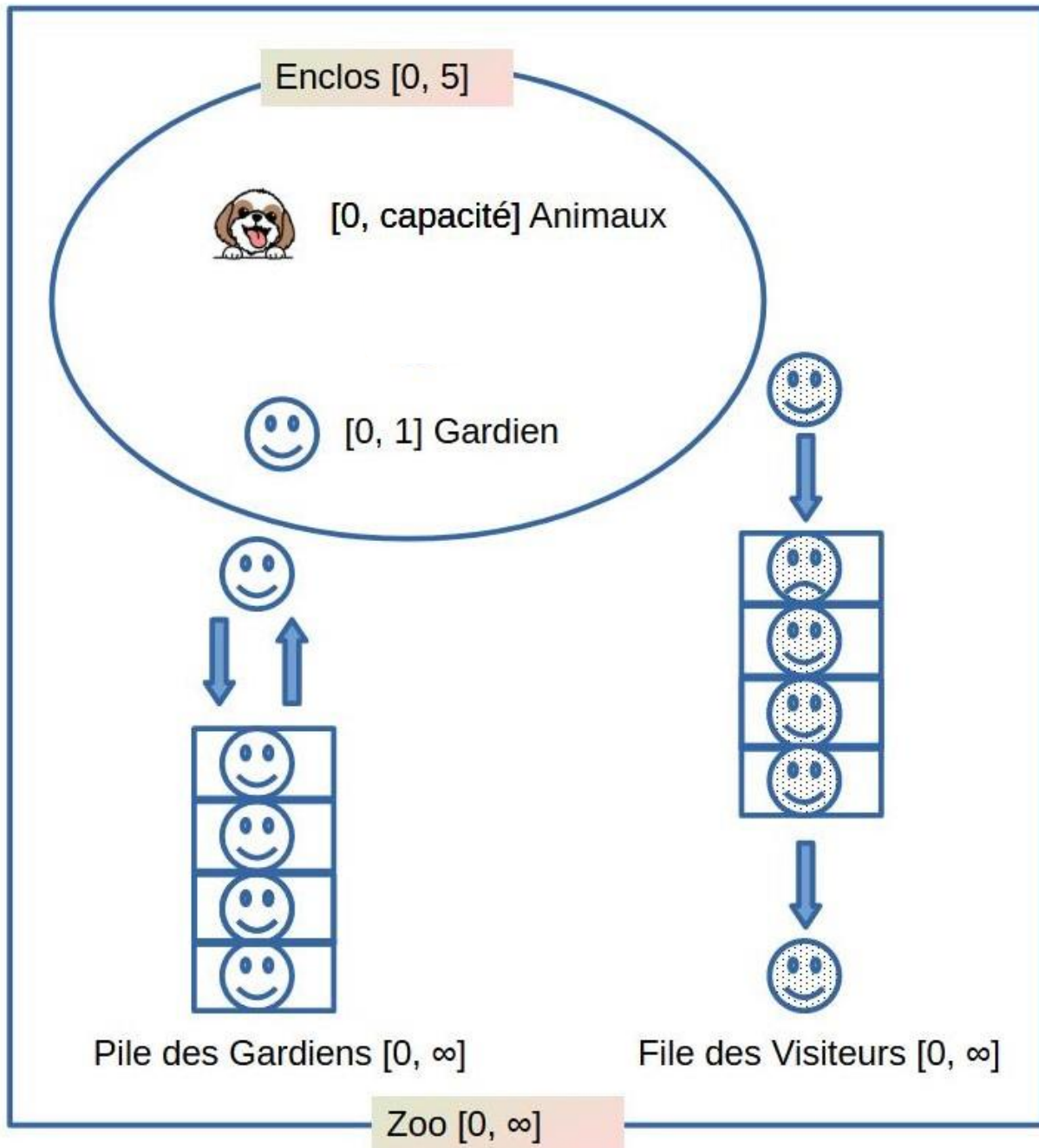
- Ce travail peut être fait **seul ou en équipe de deux** :
 - Si le travail est fait en équipe, indiquez le nom des deux coéquipiers en commentaire en haut du fichier Main.java (mais ne modifiez pas le reste du fichier dans votre remise).
- Ce travail doit être fait à partir de **GitHub Campus** : <https://classroom.github.com/a/UFnyZRYg>
 - Choisissez d'abord votre nom et équipe parmi la liste des étudiants de votre groupe;
 - Faites des *commit* et des *push* régulièrement tout au long de votre développement.
- Il est strictement défendu d'utiliser des classes provenant des librairies Java pour les structures de données (p.ex. *ArrayList*, *LinkedList*, etc.). Si vous avez absolument besoin d'importer une classe Java (autre que *Math*), vérifiez d'abord avec votre enseignant.
- Le travail doit être remis au plus tard à la date officielle indiquée sur Léa. Après cette date, 10% de pénalité sera enlevé par jour de retard.
- Toute forme de plagiat entrainera automatiquement la note zéro (0) ainsi qu'un rapport officiel à votre dossier. N'oubliez-pas d'indiquer vos références si vous utilisez des extraits de code trouvés sur internet ou provenant d'une autre personne.
- La qualité du français est également importante. Jusqu'à 10% de la note finale de votre travail pourrait être retirée pour cause d'un mauvais usage du français.

Critères d'évaluation

- Fonctionnement correct du programme (validation à l'aide de tests unitaires)
- Modélisation des classes demandées (attributs, méthodes, encapsulation, etc.)
- Qualité du code (noms des méthodes et variables, formatage, lisibilité, etc.)
- Utilisation judicieuse de Git et GitHub

Mise en contexte

Votre mandat est de coder les classes qui serviront à simuler l'organisation d'un zoo. Un zoo est constitué d'une série d'enclos dans lesquels se trouvent des animaux d'une même « famille ». Un ou plusieurs gardiens peuvent être assignés à ces enclos. Des visiteurs sont admis dans les enclos de leur choix où ils peuvent observer les gardiens entraîner les animaux présents. Un zoo gère l'embauche de gardiens selon leur ancienneté, donc en utilisant une pile, tandis que les visiteurs sont gérés à l'aide d'une file.



Classes à coder

Classe **Animal** (5%)

Un **Animal** est défini par :

- Une **famille**, où la « famille » fait référence à une « classe » d'animaux, comme les mammifères ou les oiseaux; L'énumération **Famille** (dans Famille.java) donne l'ensemble des valeurs possibles pour cette variable.
- Une **espèce**, qui est une sous-catégorie de la famille plus spécifique à chaque animal; Étant donné qu'il existe de très nombreuses espèces, on utilise une simple *String* pour les représenter.
- Un **nom** (sans espaces au début ou à la fin)
- Un **poids** (en kilogrammes)

En plus des accesseurs nécessaires, les méthodes suivantes sont requises par le programme principal :

- public **Animal**(Famille famille, String espece, String nom, double poids) :
 - Le constructeur de la classe; L'espèce doit être écrite entièrement en lettres minuscules, et sans espaces au début ou à la fin de la *String*.
- public double **setPoids**(double poids) :
 - Le mutateur pour le poids de l'animal; Les autres attributs n'ont que des accesseurs car ils sont tous inchangeables (une fois l'objet Animal instancié).

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe **Enclos** (10%)

Un **Enclos** est défini par :

- Une **famille** (ne peut être modifiée après la création de l'enclos)
- Un **nom** (ne peut être modifié après la création de l'enclos)
- Un **nombre de gardiens**
- Un **nombre d'animaux**
- Une **capacité maximum**, soit le nombre maximum d'animaux que peut accueillir l'enclos
- Un tableau des **animaux** présents dans l'enclos

En plus des accesseurs nécessaires, les méthodes suivantes sont utilisées par le programme principal :

- Public **Enclos**(Famille famille, String nom, int capaciteMax) :
 - Le constructeur de la classe.
- public int **augmenterGardiens**() et **diminuerGardiens**() :
 - Ces deux méthodes permettent d'incrémenter ou de décrémenter le nombre de gardiens assignés à cet enclos. Il ne peut pas y avoir moins de 0 gardiens.

- public boolean **ajouterAnimal**(Animal animal) :
 - Ajoute un animal à l'enclos s'il n'est pas plein (capacité maximum atteinte) et si l'animal est de la même famille que l'enclos. Retourne *false* sinon.
- public void **ajouterAnimal**(Animal[] animaux) :
 - Surcharge de la méthode précédente qui ajoute un ensemble d'animaux à l'enclos; Retourne *true* si tous les animaux ont été ajoutés avec succès.
- public boolean **contientEspece**(String espece) :
 - Retourne *true* si au moins l'un des animaux présents dans l'enclos correspond à l'espèce passée en paramètre.
- public void **afficherEnclos**() :
 - Méthode utilisée lors de l'affichage du Zoo sur la console; Référez-vous à l'annexe 1 pour le formatage attendu.

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe Gardien (10%)

Un **Gardien** est défini par :

- Un **id** : un identifiant numérique unique pour chaque gardien, généré automatiquement par le constructeur (valeurs à partir de 1000 – ne peut être modifié une fois établi)
- Un **nom** (ne peut être modifié)
- Une **specialité** (une famille d'animaux – ne peut être modifiée)
- **hrsExperience** : un nombre réel d'heures d'expérience, qui peuvent s'accumuler
- Un **enclos** auquel le gardien est assigné

En plus des accesseurs nécessaires, les méthodes suivantes sont utilisées par le programme principal :

- public **Gardien**(String nom, double hrsExperience, Famille specialite) :
 - Le constructeur de la classe.
- public void **ajouterExperience**(int minutes) :
 - Augmente le nombre d'heures d'expérience (nb : le paramètre est en minutes)
- public boolean **assignerA**(Enclos enclos) :
 - Méthode permettant d'assigner le gardien à un enclos; Il faut cependant s'assurer que soit la spécialité du gardien correspond à la même famille que celle de l'enclos, ou alors qu'il y a déjà au moins un autre gardien assigné à cet enclos. Si l'assignation est valide, n'oubliez pas de retirer le gardien de son enclos actuel et de signaler à l'enclos cible que son nombre de gardiens a augmenté. Retourne *false* en cas d'échec.
- public void **quitterEnclos**() :
 - Méthode permettant de retirer le gardien de son enclos actuel (s'il n'est pas déjà *null*).
- public String **toString**() :
 - Format attendu : *[nom (#id), heures d'expérience (1 décimale), enclos 'nom de l'enclos']*

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe Visiteur (5%)

Un **Visiteur** est défini par :

- Un **nom** (ne peut être modifié)
- Un **age** (un nombre entier d'années)
- Un tableau d'**espèces** (sous forme de *Strings*) que le visiteur voudrait voir
- Une file d'**enclos choisis** qu'il ira visiter (après avoir acheté son billet)

En plus des accesseurs nécessaires, les méthodes suivantes sont utilisées par le programme principal :

- public **Visiteur**(String nom, int age, String[] especes):
 - Le constructeur de la classe.
- Public void **setEnclosChoisis**(File<Enclos> enclosChoisis)
 - Une fois le visiteur « accueilli » au zoo et son billet acheté, il reçoit une file d'enclos qu'il ira visiter. Vous n'avez pas à valider cette liste.
- public Enclos **prochainEnclos**() :
 - Méthode pour retirer le prochain enclos à visiter (le visiteur entre alors dans cet enclos). Retourne *null* s'il n'y a plus d'enclos restants.
- public String **toString**() :
 - Format attendu : `[nom, âge ans, nombre d'enclos en file enclos restants]`

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe Zoo (30%)

Un **Zoo** est défini par :

- **nom** : Le nom du zoo (ne peut pas être modifié)
- **enclos** : Un tableau d'objets Enclos
- **pileGardiens** : Une structure de données de type « pile » pour les gardiens
- **fileVisiteurs** : Une structure de données de type « file » pour les visiteurs

En plus des accesseurs nécessaires, les méthodes suivantes sont utilisées par le programme principal :

- public **Zoo**(String nom, Enclos[] enclos) :
 - Le constructeur de la classe; Pas besoin de copier les enclos reçus en paramètre.
- public int **getNbEnclos**() :
 - Retourne le nombre d'enclos dans le zoo.
- public int **getNbTotalAnimaux**() :
 - Retourne le nombre total d'animaux présents dans le zoo (tous les enclos combinés).

- public Gardien **engagerGardien**(String nom, double hrsExperience, Famille specialite, Enclos enclos) :
 - Méthode pour ajouter un nouveau gardien à la pile et l'assigner à un enclos spécifique. S'il est impossible d'assigner le gardien à l'enclos spécifié, il n'est pas engagé (donc il ne sera pas ajouté à la pile des gardiens), et la méthode retournera *null*.
- public Gardien **engagerGardien**(String nom, double hrsExperience, Famille specialite) :
 - Surcharge de la méthode précédente qui va automatiquement choisir un enclos approprié auquel assigner le nouveau gardien en fonction de sa spécialité. Si plusieurs enclos correspondent à la spécialité du gardien, on choisira celui qui a le minimum de gardiens déjà présents. Si aucun enclos ne convient, le gardien n'est pas engagé.
- public Gardien **renvoyerGardien()** :
 - Méthode pour « retirer » le dernier gardien engagé (donc celui au sommet de la pile). Cependant, si ce gardien est le seul qui est assigné à son enclos, alors on ne peut pas le congédier, et la méthode retourne plutôt *null*.
- public Pile **getGardiensDe**(Enclos enclos) :
 - Méthode qui permet de retourner une pile des gardiens assignés à un enclos spécifique. Il s'agit en quelque sorte de retourner un sous-ensemble de la pile globale des gardiens qui n'inclue que les gardiens assignés à cet enclos.
- public double **accueillirVisiteur**(Visiteur visiteur) :
 - Méthode pour ajouter un visiteur dans la file d'attente du zoo.
 - Ce visiteur doit d'abord passer à la billetterie, où le prix de son billet est calculé en fonction des espèces que le visiteur souhaite voir : on lui donnera accès à tous les enclos contenant des animaux des espèces visées, à condition que ces enclos soient surveillés par au moins un gardien.
 - Attention, la file des enclos choisis ne doit pas contenir de doublons (c-à-d. plus d'une fois le même enclos).
 - Le **prix** de l'entrée est de 5.50\$ par enclos choisi plus une surcharge de 1.00\$ par animal présent dans le zoo (qu'ils soient visités ou non). Par contre, c'est gratuit pour les visiteurs de 6 ans ou moins, et un rabais de 50% s'applique pour les visiteurs entre 7 et 13 ans (inclusivement) ainsi que pour ceux de 65 ans et plus.
 - Une fois terminé, la méthode assigne la file des enclos choisi au visiteur et retourne le prix d'entrée calculé pour ce visiteur.
- public Visiteur **prochainVisiteur()** :
 - Méthode pour retirer le prochain visiteur (il entre dans le zoo et commence sa visite). Retourne *null* lorsqu'il n'y a plus de visiteurs en attente.
- public void **afficherTout()** :
 - Méthode utilisée lors de l'affichage du Zoo sur la console; Référez-vous à l'annexe 1 pour le formatage attendu.

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe File (12%)

Une **File** est définie par :

- Un **nombre d'éléments** présents dans la file
- Un pointeur sur son **premier** nœud
- Optionnellement, un pointeur sur son **dernier** nœud

La classe File servira à contenir soit des **Visiteur**, soit des **Enclos**. Vous pouvez faire deux copies de cette classe (avec des noms légèrement différents), ou alors utiliser un paramètre de type générique `<T>` pour éviter de dupliquer votre code (points bonus le cas échéant).

En plus des accesseurs nécessaires, les méthodes suivantes pourraient vous être utiles :

- `public boolean contient(T valeur):`
 - Retourne *true* si la valeur est présente quelque part dans la file.
- `public void ajouter(T valeur):`
 - Ajoute la valeur à la fin de la file.
- `public T retirer():`
 - Retire et retourne la première valeur de la file.
- `public String toString() :`
 - Pour convertir la file en *String*. Référez-vous à l'annexe 1 pour le formatage.

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Classe Noeud (3%)

Un **Noeud** est défini par :

- Une **valeur** (un Visiteur, un Enclos, ou un type générique (bonus))
- Un pointeur sur le nœud **suivant**

Cette classe peut être interne à la classe **File**. De plus, les attributs de cette classe peuvent être publics et il n'y a pas de méthodes exigées dans cette classe, quoique vous trouverez peut-être utile d'ajouter un constructeur et une méthode *toString()*.

Classe Pile (10%)

Une **Pile** est définie par :

- Les constantes **TAILLE_INITIALE = 5** et **RATIO_AGRANDISSEMENT = 2**
- Un **nombre d'éléments** présents dans la pile
- Un tableau de **gardiens**

La classe Pile servira seulement à contenir des **Gardien**, donc il n'est pas nécessaire d'utiliser un type générique `<T>` ici.

En plus des accesseurs nécessaires, les méthodes suivantes pourraient vous être utiles :

- public Gardien **get**(int index) :
 - Bien qu'il s'agisse d'une pile (et donc que seul l'élément au « sommet » de la pile devrait être accessible), il peut tout de même être utile d'ajouter une méthode donnant accès (en lecture seulement) à n'importe quel autre élément selon son index.
- public Gardien **peek**() :
 - Retourne l'élément au sommet de la pile mais sans le retirer.
- public void **push**() :
 - Ajoute un élément au sommet de la pile.
- public Gardien **pop**() :
 - Retire et retourne l'élément au sommet de la pile.
- public String **toString**() :
 - Pour convertir la pile en *String*. Référez-vous à l'annexe 1 pour le formatage.

Vous pouvez aussi ajouter n'importe quelle autre méthode ou attribut qui vous semble pertinent.

Qualité du code (5%)

Portez toujours attention à la qualité de votre code, notamment :

- Identificateurs clairs et pertinents
- Indentation et espacement adéquats
- Lignes et méthodes relativement courtes

De plus, des tests unitaires vous seront fournis pour valider votre travail. **Votre code doit compiler et passer ces tests sans qu'ils aient été modifiés** (sauf ajustements mineurs aux endroits précisés).

Utilisation de Git et GitHub (5%)

Faites régulièrement des **commit** durant votre travail; Il devrait y avoir au moins une dizaine de *commits* accompagnés de messages pertinents dans votre référentiel (*repo*).

Remise (5%)

Une fois terminé, faites un dernier **push** de votre travail sur **GitHub**; N'oubliez pas d'inclure tous vos fichiers de code source java ainsi qu'un projet **IntelliJ** valide (fichier « TP2_VotreNom.iml » et le dossier caché « .idea »). Si vous avez fait le travail en équipe, une seule remise est nécessaire.

Annexe 1 : Exemple d'exécution du Main.java

Cet exemple ne couvre pas tous les cas possibles, mais donne une bonne idée du résultat attendu.

Pour vous assurer d'une bonne couverture, il faut lire attentivement les consignes pour chaque classe.

```
=====LE ZOO AVANT LES VISITES=====
Spike le thon a été ajouté à l'enclos 'Les Poissons'.
Finn le saumon a été ajouté à l'enclos 'Les Poissons'.
Neptune le dauphin a été ajouté à l'enclos 'Les Cétacés'.
Flipper le marsouin a été ajouté à l'enclos 'Les Cétacés'.
Monty le python a été ajouté à l'enclos 'Les Reptiles'.
Jaws le alligator a été ajouté à l'enclos 'Les Reptiles'.
Polly le perroquet a été ajouté à l'enclos 'Les Oiseaux'.
Apollo le corbeau a été ajouté à l'enclos 'Les Oiseaux'.
Simba le lion a été ajouté à l'enclos 'Les Mammifères'.
Zara le zèbre a été ajouté à l'enclos 'Les Mammifères'.
Le gardien M. Smith (#1000) a été assigné à l'enclos 'Les Poissons' (1 gardiens).
Le gardien B. Patel (#1001) a été assigné à l'enclos 'Les Cétacés' (1 gardiens).
Le gardien F. Lyding (#1002) a été assigné à l'enclos 'Les Reptiles' (1 gardiens).
Le gardien P. Grandbois (#1003) a été assigné à l'enclos 'Les Oiseaux' (1
gardiens).
Le gardien L. Skelgaard (#1004) a été assigné à l'enclos 'Les Mammifères' (1
gardiens).

===== LE ZOO AVANT LES VISITES =====
Zap le merlan a été ajouté à l'enclos 'Les Poissons'.
Obélix le sanglier a été ajouté à l'enclos 'Les Mammifères'.
Le gardien H. Bourassa (#1005) a été assigné à l'enclos 'Les Oiseaux' (2 gardiens).
Le gardien R. Berry (#1006) a été assigné à l'enclos 'Les Mammifères' (2 gardiens).
[Alice, 13 ans] arrive au zoo; Son entrée coûte 14,25$ et donne accès à 3 enclos.
[Robert, 65 ans] arrive au zoo; Son entrée coûte 8,75$ et donne accès à 1 enclos.
[Judith, 56 ans] arrive au zoo; Son entrée coûte 23,00$ et donne accès à 2 enclos.
[Edgar, 6 ans] arrive au zoo; Son entrée coûte 0,00$ et donne accès à 1 enclos.
Le zoo a vendu des billets pour un total de 46,00$.

Bienvenu au zoo de Granby!
Voici la pile des gardiens:
  7 gardiens: [M. Smith (#1000), 10,0 hrs, enclos 'Les Poissons'] | [B. Patel
  (#1001), 5,0 hrs, enclos 'Les Cétacés'] | [F. Lyding (#1002), 20,0 hrs, enclos
  'Les Reptiles'] | [P. Grandbois (#1003), 15,0 hrs, enclos 'Les Oiseaux'] | [L.
  Skelgaard (#1004), 10,0 hrs, enclos 'Les Mammifères'] | [H. Bourassa (#1005), 8,0
  hrs, enclos 'Les Oiseaux'] | [R. Berry (#1006), 12,0 hrs, enclos 'Les
  Mammifères'] | [+ 3 cellules vides].
Et la file des visiteurs:
  4 éléments: [Alice, 13 ans, 3 enclos restants] -> [Robert, 65 ans, 1 enclos
  restants] -> [Judith, 56 ans, 2 enclos restants] -> [Edgar, 6 ans, 1 enclos
  restants] -> <null>.
Le zoo est peuplé par 12 animaux répartis dans 5 enclos:
L'enclos 'Les Poissons' contient 3 Poisson:
  Le thon nommé Spike pèse 100,0 kg
  Le saumon nommé Finn pèse 100,0 kg
```

Le merlan nommé Zap pèse 50,0 kg
Il y a 1 gardien(s) assignés à cet enclos.
L'enclos 'Les Cétacés' contient 2 Cetace:
Le dauphin nommé Neptune pèse 250,0 kg
Le marsouin nommé Flipper pèse 250,0 kg
Il y a 1 gardien(s) assignés à cet enclos.
L'enclos 'Les Reptiles' contient 2 Reptile:
Le python nommé Monty pèse 300,0 kg
Le alligator nommé Jaws pèse 300,0 kg
Il y a 1 gardien(s) assignés à cet enclos.
L'enclos 'Les Oiseaux' contient 2 Oiseau:
Le perroquet nommé Polly pèse 0,8 kg
Le corbeau nommé Apollo pèse 0,8 kg
Il y a 2 gardien(s) assignés à cet enclos.
L'enclos 'Les Mammifères' contient 3 Mammifere:
Le lion nommé Simba pèse 200,0 kg
Le zèbre nommé Zara pèse 200,0 kg
Le sanglier nommé Obélix pèse 150,0 kg
Il y a 2 gardien(s) assignés à cet enclos.

===== DÉBUT DES VISITES =====

[Alice, 13 ans, 3 enclos restants] entre dans le zoo.
[Alice, 13 ans, 3 enclos restants] entre dans l'enclos 'Les Poissons'.
3 animaux et 1 gardiens sont présents: M. Smith,
[Alice, 13 ans, 2 enclos restants] quitte l'enclos après 8 minutes.
[Alice, 13 ans, 2 enclos restants] entre dans l'enclos 'Les Mammifères'.
3 animaux et 2 gardiens sont présents: R. Berry, L. Skelgaard,
[Alice, 13 ans, 1 enclos restants] quitte l'enclos après 2 minutes.
[Alice, 13 ans, 1 enclos restants] entre dans l'enclos 'Les Oiseaux'.
2 animaux et 2 gardiens sont présents: H. Bourassa, P. Grandbois,
[Alice, 13 ans, 0 enclos restants] quitte l'enclos après 23 minutes.
[Alice, 13 ans, 0 enclos restants] quitte le zoo après un total de 33 minutes.
[Robert, 65 ans, 1 enclos restants] entre dans le zoo.
[Robert, 65 ans, 1 enclos restants] entre dans l'enclos 'Les Reptiles'.
2 animaux et 1 gardiens sont présents: F. Lyding,
[Robert, 65 ans, 0 enclos restants] quitte l'enclos après 20 minutes.
[Robert, 65 ans, 0 enclos restants] quitte le zoo après un total de 20 minutes.
[Judith, 56 ans, 2 enclos restants] entre dans le zoo.
[Judith, 56 ans, 2 enclos restants] entre dans l'enclos 'Les Oiseaux'.
2 animaux et 2 gardiens sont présents: H. Bourassa, P. Grandbois,
[Judith, 56 ans, 1 enclos restants] quitte l'enclos après 4 minutes.
[Judith, 56 ans, 1 enclos restants] entre dans l'enclos 'Les Mammifères'.
3 animaux et 2 gardiens sont présents: R. Berry, L. Skelgaard,
[Judith, 56 ans, 0 enclos restants] quitte l'enclos après 23 minutes.
[Judith, 56 ans, 0 enclos restants] quitte le zoo après un total de 27 minutes.
[Edgar, 6 ans, 1 enclos restants] entre dans le zoo.
[Edgar, 6 ans, 1 enclos restants] entre dans l'enclos 'Les Cétacés'.
2 animaux et 1 gardiens sont présents: B. Patel,
[Edgar, 6 ans, 0 enclos restants] quitte l'enclos après 29 minutes.
[Edgar, 6 ans, 0 enclos restants] quitte le zoo après un total de 29 minutes.

===== FIN DES VISITES =====

Le gardien R. Berry (#1006) a été retiré de l'enclos 'Les Mammifères' (1 gardiens).

Le dernier gardien arrivé au zoo [R. Berry (#1006), 12,4 hrs, enclos 'null'] a été congédié.
Le gardien H. Bourassa (#1005) a été retiré de l'enclos 'Les Oiseaux' (1 gardiens).
Le dernier gardien arrivé au zoo [H. Bourassa (#1005), 8,5 hrs, enclos 'null'] a été congédié.
Le gardien [L. Skelgaard (#1004), 10,4 hrs, enclos 'Les Mammifères'] ne peut pas être congédié car son enclos serait vide.

===== LE ZOO APRÈS LES VISITES =====

Bienvenu au zoo de Granby!

Voici la pile des gardiens:

5 gardiens: [M. Smith (#1000), 10,1 hrs, enclos 'Les Poissons'] | [B. Patel (#1001), 5,5 hrs, enclos 'Les Cétacés'] | [F. Lyding (#1002), 20,3 hrs, enclos 'Les Reptiles'] | [P. Grandbois (#1003), 15,5 hrs, enclos 'Les Oiseaux'] | [L. Skelgaard (#1004), 10,4 hrs, enclos 'Les Mammifères'] | [+ 5 cellules vides].

Et la file des visiteurs:

0 éléments: <null>.

Le zoo est peuplé par 12 animaux répartis dans 5 enclos:

L'enclos 'Les Poissons' contient 3 Poisson:

Le thon nommé Spike pèse 100,0 kg
Le saumon nommé Finn pèse 100,0 kg
Le merlan nommé Zap pèse 50,0 kg

Il y a 1 gardien(s) assignés à cet enclos.

L'enclos 'Les Cétacés' contient 2 Cetace:

Le dauphin nommé Neptune pèse 250,0 kg
Le marsouin nommé Flipper pèse 250,0 kg

Il y a 1 gardien(s) assignés à cet enclos.

L'enclos 'Les Reptiles' contient 2 Reptile:

Le python nommé Monty pèse 300,0 kg
Le alligator nommé Jaws pèse 300,0 kg

Il y a 1 gardien(s) assignés à cet enclos.

L'enclos 'Les Oiseaux' contient 2 Oiseau:

Le perroquet nommé Polly pèse 0,8 kg
Le corbeau nommé Apollo pèse 0,8 kg

Il y a 1 gardien(s) assignés à cet enclos.

L'enclos 'Les Mammifères' contient 3 Mammifere:

Le lion nommé Simba pèse 200,0 kg
Le zèbre nommé Zara pèse 200,0 kg
Le sanglier nommé Obélix pèse 150,0 kg

Il y a 1 gardien(s) assignés à cet enclos.