

Introduction

Cette documentation combine les informations des fichiers `Employe.java`, `GestionPersonnel.java`, `Ligue.java`, `EmployeConsole.java`, `LigueConsole.java`, et `PersonnelConsole.java`. Elle explique le fonctionnement global du système de gestion du personnel des ligues, qui permet de gérer les employés, les ligues, et les interactions entre eux via une interface en ligne de commande.

```
C:\Users\adobe>java Application C:\Users\adobe\AppData\Local\Temp\jpservice-java-0x6c-1c-fc-win32-x86_64\jpservice-pgmscript.jar;openjdk.compiler.exe\nvmlu  
password : password : login  
Gestion du personnel des ligues  
c : Gérer le compte root  
l : Gérer les ligues  
q : Quitter  
  
Select an option : Gestion du personnel des ligues  
c : Gérer le compte root  
l : Gérer les ligues  
q : Quitter  
  
Select an option : l  
|  
|-----  
|  
|-----  
|  
|-----  
|  
Gérer les ligues  
l : Afficher les ligues  
a : Ajouter une lige  
e : Sélectionner une lige  
q : Back  
  
Select an option : Gérer les ligues  
l : Afficher les ligues  
a : Ajouter une lige  
e : Sélectionner une lige  
q : Back  
  
Select an option :
```

```
/**
 * Retourne le prénom de l'employé.
 *
 * @return Le prénom de l'employé.
 */
public String getPrenom() {
    return prenom;
}

/**
 * Modifie le prénom de l'employé.
 *
 * @param prenom Le nouveau prénom de l'employé.
 */
public void setPrenom(String prenom) {
    this.prenom = prenom;
}
```

Documentation combinée pour le système de gestion du personnel des ligues

Employe.java

- Ce fichier contient la définition de la classe `Employe`, qui modélise les employés du système. Il inclut des attributs tels que le nom, le prénom, l'identifiant et les méthodes associées pour gérer les informations relatives à chaque employé.

EmployeConsole.java

- Ce fichier gère l'interaction utilisateur pour les employés via une interface en ligne de commande. Il permet aux utilisateurs d'interagir avec le système en exécutant différentes commandes.

LigueConsole.java

- Similaire à `EmployeConsole.java`, cette classe permet l'interaction avec les ligues. Les utilisateurs peuvent gérer les ligues à travers des commandes en ligne de commande.

Ligue.java

- La classe `Ligue` représente les différentes ligues dans le système. Elle contient des détails tels que le nom de la ligue, les employés associés et les événements programmés.

PersonnelConsole.java

- Cette classe regroupe toutes les fonctionnalités de gestion du personnel dans une interface en ligne de commande centralisée. Elle facilite l'accès aux différentes opérations liées aux employés et aux ligues.

Classes principales - Employe.java

Attributs de la classe Employe

- La classe Employé comprend plusieurs attributs qui définissent chaque employé. Ces attributs incluent le nom, le prénom, l'adresse e-mail, le mot de passe, la ligue à laquelle l'employé est rattaché, ainsi que les dates d'arrivée et de départ. Ces informations sont essentielles pour gérer les employés au sein d'une ligue.

Méthodes principales de la classe Employe

- Les méthodes principales de la classe Employe permettent d'interagir avec les attributs de la classe. Par exemple, les méthodes 'getNom()' et 'setNom(String nom)' permettent d'accéder et de modifier le nom de l'employé. D'autres méthodes, comme 'estAdmin(Ligue ligue)', vérifient si l'employé a des droits d'administration.

Gestion des droits administratifs

- La méthode 'estAdmin(Ligue ligue)' est cruciale pour déterminer si un employé est un administrateur de sa ligue. De plus, si un employé est supprimé, la méthode 'remove()' transfère les droits d'administration au super-utilisateur, garantissant ainsi une gestion fluide.

Exceptions liées aux dates

- La classe Employe gère les exceptions liées aux dates d'arrivée et de départ. Par exemple, 'DateInvalideException' est lancée si une date est déjà dans le passé, et 'DateIncohérenteException' si la date de départ précède la date d'arrivée. Cela garantit l'intégrité des données des employés.

GestionPersonnel.java

Point central de gestion

- La classe GestionPersonnel est essentielle pour la gestion des ligues et des employés, facilitant ainsi toutes les opérations liées à ces entités.

Attributs de la classe

- Les attributs principaux incluent 'ligues', qui est une collection triée des ligues, et 'root', qui désigne le super-utilisateur du système.

Méthodes principales

- Les méthodes comme 'getGestionPersonnel()' et 'sauvegarder()' sont cruciales pour récupérer l'instance unique de la classe et pour sauvegarder les données respectivement.

Gestion des ligues

- La méthode 'getLigues()' permet de récupérer la liste des ligues, tandis que 'addLigue(String nom)' et 'remove(Ligue ligue)' gèrent respectivement l'ajout et la suppression de ligues.

Accès au super-utilisateur

- La méthode 'getRoot()' permet d'accéder au super-utilisateur, garantissant que les opérations critiques sont exécutées par une entité autorisée.

Ligue.java

Nom de la ligue

- Chaque instance de la classe Ligue possède un attribut ``nom`` qui représente le nom unique de la ligue. Ce nom est essentiel pour l'identification et la gestion de la ligue.

Administrateur de la ligue

- L'attribut ``administrateur`` désigne l'employé qui gère la ligue. Cet administrateur a des responsabilités et des droits spécifiques pour superviser les activités de la ligue.

Méthodes pour gérer le nom

- Les méthodes ``getNom()`` et ``setNom(String nom)`` permettent d'accéder et de modifier le nom de la ligue. Ces méthodes sont essentielles pour la manipulation des données de la ligue.

Gestion de l'administrateur des ligues

- Les méthodes ``getAdministrateur()`` et ``setAdministrateur(Employe administrateur)`` permettent de récupérer et de définir l'administrateur de la ligue, assurant ainsi une bonne gestion des droits administratifs.

Ajout d'un nouvel employé

- La méthode ``addEmploye(...)`` permet d'ajouter un nouvel employé à la ligue. Cette fonctionnalité est cruciale pour maintenir la liste des employés à jour.

Suppression d'un employé

- La méthode ``remove(Employe employe)`` permet de supprimer un employé spécifique de la ligue, garantissant ainsi que seul le personnel actif est conservé.

Accès aux employés

- La méthode ``getEmployes()`` est utilisée pour retourner la liste complète des employés de la ligue, facilitant ainsi l'accès aux informations nécessaires pour la gestion.

Suppression de la ligue

- La méthode ``remove()`` permet de supprimer la ligue ainsi que tous ses employés. Cela est utile lorsque la ligue n'est plus nécessaire.

Classes d'interface en ligne de commande

Classe EmployeConsole

- La classe EmployeConsole est conçue pour gérer les employés à travers une interface en ligne de commande, permettant une interaction directe et efficace avec les données des employés.

Méthode afficher

- La méthode ``afficher(Employe employe)`` est utilisée pour afficher toutes les informations pertinentes liées à un employé donné, facilitant ainsi la consultation des données.

Méthode editerEmploye

- La méthode ``editerEmploye(Employe employe)`` crée un menu interactif qui permet à l'utilisateur de modifier les informations d'un employé de manière structurée et intuitive.

Méthodes de changement d'attributs

- Les méthodes ``changerNom``, ``changerPrenom``, ``changerMail``, ``changerPassword``, ``changerDataArrivee``, et ``changerDateDepart`` offrent des options spécifiques pour modifier les attributs d'un employé. Cela permet une flexibilité dans la gestion des informations.
-

LigueConsole.java

menuLigues()

- Affiche le menu principal pour gérer les ligues, permettant aux utilisateurs de naviguer facilement entre les différentes options disponibles.

afficherLigues()

- Affiche la liste complète des ligues existantes, fournissant aux utilisateurs une vue d'ensemble des ligues gérées.

ajouterLigue()

- Permet d'ajouter une nouvelle ligue, facilitant ainsi l'expansion des options disponibles pour les utilisateurs.

ajouterEmploye(Ligue ligue)

- Ajoute un nouvel employé à une ligue, renforçant la gestion des ressources humaines au sein de l'organisation.

editerLigue(Ligue ligue)

- Crée un menu pour éditer les informations d'une ligue spécifique, assurant que les données restent à jour et pertinentes.

gererEmployes(Ligue ligue)

- Crée un menu pour gérer les employés d'une ligue, permettant une gestion efficace du personnel.

supprimer(Ligue ligue)

- Supprime une ligue existante, offrant ainsi une flexibilité dans la gestion des ligues en fonction des besoins de l'organisation.

PersonnelConsole.java

Point d'entrée principal de l'application

- La classe PersonnelConsole est le point d'entrée principal de l'application, servant de hub pour la gestion des ligues et des employés.

Méthode menuQuitter()

- La méthode menuQuitter() présente un menu spécifique pour quitter l'application, assurant que l'utilisateur a conscience des options disponibles.

Méthode start()

- La méthode start() est responsable du démarrage de l'application, initiant ainsi toutes les fonctionnalités disponibles pour l'utilisateur.

Méthode menuPrincipal()

- Cette méthode crée le menu principal, offrant aux utilisateurs diverses options pour gérer les employés et les ligues de manière intuitive.

Fonctionnement global

1. Authentification

- L'application débute son fonctionnement par une étape cruciale : l'authentification. L'administrateur root est invité à entrer son mot de passe. Si le mot de passe saisi est correct, l'accès à l'application est accordé, ce qui permet de garantir la sécurité des données et des opérations.

2. Menu Principal

- Le menu principal de l'application offre plusieurs options essentielles pour la gestion. Il permet à l'utilisateur de : gérer les employés via 'EmployeConsole', gérer les ligues via 'LigueConsole', ou quitter l'application. Cette interface centralisée simplifie l'accès aux fonctionnalités clés.

3. Gestion des Ligues

- Dans cette section, l'utilisateur peut effectuer diverses actions sur les ligues. Il est possible de créer, modifier ou supprimer des ligues, ainsi que de gérer les employés associés à une ligue. De plus, l'administrateur d'une ligue peut être changé, ce qui offre une flexibilité dans la gestion des ressources humaines.

4. Gestion des Employés

- Cette fonctionnalité permet de modifier les détails d'un employé, notamment son nom, prénom, e-mail, mot de passe, ainsi que les dates d'arrivée et de départ. Il est également possible de supprimer un employé, ce qui aide à maintenir à jour les informations des ressources humaines.

Travail Accompli sur l'Application "Maison de Ligue"

1. Représentation Graphique des Menus

- Exécution de l'application : L'application a été lancée avec succès, permettant d'observer les menus affichés dans la ligne de commande.
- Création de l'arbre heuristique : Un graphe a été créé où chaque menu et chaque option est représenté par un nœud. Les arêtes ont été utilisées pour représenter l'imbrication des nœuds.

2. Modèle Conceptuel de Données (MCD)

- Analyse des données : Les données affichées en ligne de commande et le code de la couche métier ont été analysés pour créer un MCD.
- Création du MCD : Les entités et leurs relations ont été représentées. Les attributs `dateArrivee` et `dateDepart` ont été ajoutés à l'entité `Employe`.

3. Modification de la Couche Métier

- Ajout de variables d'instance : Des variables de type `LocalDate` pour `dateArrivee` et `dateDepart` ont été ajoutées.
- Constructeur, getters, setters : Un constructeur a été ajouté pour initialiser les dates. Des getters et setters ont été implémentés pour les dates.
- Gestion des exceptions : Des exceptions sont déclenchées en cas de dates incohérentes (par exemple, `dateDepart` avant `dateArrivee`).

4. Tests Unitaires

- Tests pour les dates : Des tests unitaires ont été écrits pour vérifier la cohérence des dates.
- Tests pour les setters : Des tests unitaires ont été ajoutés pour vérifier le bon fonctionnement des setters.

5. Gestion des Objets et Administrateurs

- Suppression d'objet : Le comportement lors de la suppression d'un objet a été vérifié.
- Modification de l'administrateur : Le comportement lors de la modification de l'administrateur d'une ligue a été vérifié.

6. Script de Création de Tables

Création du script SQL : Un fichier .sql contenant les instructions de création des tables a été créé et versionné dans le dépôt du projet.

7. Modification du Dialogue en Ligne de Commande

Sélection d'un employé : La possibilité de sélectionner un employé avant de choisir de le modifier ou de le supprimer a été ajoutée.

Modification de l'administrateur : La possibilité de modifier l'administrateur d'une ligue a été ajoutée.

Saisie des dates : La saisie des dates de l'employé avec traitement des erreurs a été implémentée.

Conclusion

Ce système de gestion du personnel des ligues permet une gestion complète des employés et des ligues via une interface en ligne de commande. Il offre des fonctionnalités pour créer, modifier, supprimer des ligues et des employés, ainsi que pour changer l'administrateur d'une ligue.