|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
|  | Liste des méthodes de chacune des classes Cette documentation s’adresse aux dévéloppeurs et détaille les méthodes des classes ‘type’, ‘tache’, ‘liste . | | | | |
|  | | | | | |
|  | LES METHODES DE LA CLASSE : Type | | | | |
|  |  | **verifStringFormat**  Cette méthode privée vérifie si une chaîne de caractères donnée est conforme aux contraintes de format. Elle renvoie true si la chaîne est conforme et false sinon.. |  |  | **Getters et setters pour les attributs id\_type, libelle, code\_couleur et bdd**  Ces méthodes permettent d'obtenir et de définir les valeurs des attributs correspondants. |
|  |  | **createType()**  Cette méthode permet de créer un nouveau type de tâche dans la base de données. Elle vérifie d'abord si les chaînes de caractères libelle et code\_couleur sont conformes aux contraintes de format. Si c'est le cas, elle insère un nouvel enregistrement dans la table type.  . |  |
|  |  | **updateType()**  Cette méthode permet de mettre à jour les informations d'un type de tâche existant dans la base de données. Elle vérifie d'abord si les chaînes de caractères libelle et code\_couleur sont conformes aux contraintes de format. Si c'est le cas, elle met à jour l'enregistrement correspondant dans la table type en fonction de l'attribut id\_type. |  |
|  |  | **deleteType()**  Cette méthode permet de supprimer un type de tâche de la base de données en fonction de l'attribut id\_type. Elle supprime l'enregistrement correspondant dans la table type. |  |  |  |
|  | | | | | |
|  | LES METHODES DE LA CLASSE : TACHE | | | | |
|  |  | **verifStringFormat**  Cette méthode privée vérifie si une chaîne de caractères donnée est conforme aux contraintes de format. Elle renvoie true si la chaîne est conforme et false sinon. |  |  | |  |  | | --- | --- | |  | **deleteTask()**  Cette méthode permet de supprimer un tache de la base de données en fonction de l'attribut id\_tache. Elle supprime l'enregistrement correspondant dans la table tache. | |
|  |  | **createTask()**  Cette méthode permet de créer une nouvelle tâche dans la base de données. Elle vérifie d'abord si les chaînes de caractères nom et description sont conformes aux contraintes de format. Si c'est le cas, elle insère un nouvel enregistrement dans la table tache. |  |  | **Getters et setters pour les attributs id\_tache, nom, description et est\_realise**  Ces méthodes permettent d'obtenir et de définir les valeurs des attributs correspondants. |
|  |  | **updateTask()**  Cette méthode permet de mettre à jour les informations d'un type de tâche existant dans la base de données. Elle vérifie d'abord si les chaînes de caractères nom et description sont conformes aux contraintes de format. Si c'est le cas, elle met à jour l'enregistrement correspondant dans la table tache en fonction de l'attribut id\_tache. |  |  | **validTask()**  Cette méthode permet de valider une tache, de marquer une tache comme realisee en mettant à jour la valeur de l’attribut est\_realise. Elle met à jour l’enregistrement correspondant dans la table tache en fonction de l’attribut id\_tache. |
|  | | | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
|  | LES METHODES DE LA CLASSE : listE | | | | | |
|  |  | | **verifStringFormat**  Cette méthode privée vérifie si une chaîne de caractères donnée est conforme aux contraintes de format. Elle renvoie true si la chaîne est conforme et false sinon. |  |  | **updateList()**  Cette méthode met à jour les attributs nom et description d'une liste existante dans la base de données. Si les chaînes de caractères sont conformes, elle met à jour la liste dans la table liste. |
|  |  | | **createList()**  Cette méthode permet de créer une nouvelle liste dans la base de données avec les attributs nom et description. Si la chaîne de caractères est conforme, elle insère la liste dans la table liste. |  |  | **deleteList()**  Cette méthode supprime la liste courante de la base de données en utilisant l'attribut id\_liste. |
|  |  | | **addTask(Tache tache)**  Cette méthode ajoute une instance de la classe Tache à la liste de tâches taches et appelle la méthode createTask de l'objet Tache pour insérer la tâche dans la base de données avec la référence à la liste courante. |  |  | **Getters et setters pour les attributs id\_liste, nom, description**  Ces méthodes permettent d'obtenir et de définir les valeurs des attributs correspondants. |
|  | | | | | | |
|  | LES METHODES DE LA CLASSE : bdd | | | | | |
|  |  | | Bdd() Le constructeur de la classe Bdd. Il établit une connexion à la base de données MySQL à l'aide de l'URL de la base de données, du nom d'utilisateur et du mot de passe. L'URL est "jdbc:mysql://localhost:3306/toutdouxlist", le nom d'utilisateur est "root" et aucun mot de passe n'est fourni. |  |  | **getMaConnection()**  Cette méthode renvoie l'instance de Connection qui représente la connexion à la base de données. Cette connexion est utilisée par d'autres classes pour exécuter des requêtes SQL. |
|  |  | |  |  |  |  |
|  |  |
|  |  |
|  |  | |  | |
|  | LES METHODES DE LA CLASSE : utilsateur | | | | | |
|  |  | | **connect()**  Cette méthode tente de connecter l'utilisateur en vérifiant les informations d'identification dans la base de données. Renvoie true si la connexion est réussie, sinon false. |  |  | **insert()**  Cette méthode insère un nouvel utilisateur dans la base de données si l'utilisateur n'est pas connecté et que les données sont au bon format. Renvoie true si l'insertion est réussie, sinon false. |
|  |  | | **update()**  Cette méthode met à jour les informations de l'utilisateur dans la base de données si l'utilisateur est connecté et que les données sont au bon format. Renvoie true si la mise à jour est réussie, sinon false. |  |  | **verifStringFormat**  Cette méthode privée vérifie si une chaîne de caractères text est au bon format. Renvoie true si le format est correct, sinon false. |
|  |  | | **delete()**  Supprime l'utilisateur de la base de données si l'utilisateur est connecté. |  |  | **verifExistEmail()**  Cette méthode vérifie si l'adresse e-mail de l'utilisateur existe déjà dans la base de données. Renvoie true si l'adresse e-mail existe, sinon false. |
|  | | | | | | |
|  |
|  |  | |  |  |  |  |
|  |  | |  |  |  |  |
|  |  |
|  |  |
|  |  | |  | |

|  |
| --- |
|  |
|  |
|  |
|  |  | |  |  |  |  |
|  |  |
|  | LES METHODES DE LA CLASSE : Consolescanner | | | | | |
|  |  | | **nextInt()**  Cette méthode lit un entier à partir de l'entrée utilisateur. Si l'entrée n'est pas un entier, affiche un message d'erreur et demande à l'utilisateur de saisir à nouveau un entier. |  |  | **choixInt(int nb1, int nb2, String )**  Cette méthode Demande à l'utilisateur de saisir un entier compris entre nb1 et nb2. Si l'entrée ne satisfait pas cette condition, affiche un message d'erreur et demande à l'utilisateur de saisir à nouveau un entier |
|  |  | | **nextDouble ()**  Cette méthode lit un nombre décimal (double) à partir de l'entrée utilisateur. Si l'entrée n'est pas un nombre décimal, affiche un message d'erreur et demande à l'utilisateur de saisir à nouveau un nombre décimal. |  |  | **choixInt(int nb1, int nb2, int defaut, String message)**  Demande à l'utilisateur de saisir un entier compris entre nb1 et nb2. Si l'entrée ne satisfait pas cette condition, renvoie la valeur par défaut defaut. |
|  |  | | **nextLine()**  Cette méthode lit une chaîne de caractères à partir de l'entrée utilisateur |  |  | **choixString(String[] options, String message)**  Demande à l'utilisateur de saisir une chaîne de caractères parmi les options fournies dans le tableau options. Si l'entrée ne correspond pas à l'une des options, affiche un message d'erreur et demande à l'utilisateur de saisir à nouveau une chaîne de caractères. |
|  | | | | | | |
|  |
|  |  | | **choixString(String[] options, int defaut)**  Demande à l'utilisateur de saisir une chaîne de caractères parmi les options fournies dans le tableau options. Si l'entrée ne correspond pas à l'une des options, renvoie la chaîne de caractères par défaut à l'indice defaut dans le tableau options. |  |  | **indexOf(String value, String[] table) ()**  Cette méthode privée, recherche l'indice de la chaîne value dans le tableau table. Si value est trouvé dans le tableau, renvoie son indice. Sinon, renvoie -1. |
|  |  | |  |  |  |  |
|  |  |
|  |  |
|  |  | |  | |