## **Université Abdelmalek Essaadi**

## **Faculté des Sciences et Techniques**

## **Département Génie Informatique**

**TANGER**

1. ***Présentation de Java***

******

***Petit historique du langage :***

* On peut faire remonter la naissance de Java à 1991.
* SUN conçoit un langage applicable à de petits appareils électriques (on parle de code embarqué).
* Java fondé sur une - syntaxe très proche de C++ & concept de machine virtuelle (Pascal)
* Traduire un programme source, non pas directement en langage machine, mais dans un pseudo langage universel
* Ce code intermédiaire (formé de byte codes), se trouve ainsi compact et portable sur n’importe quelle machine
* Si cette dernière dispose d’un programme approprié (machine virtuelle) permettant de l’interpréter dans le langage de la machine concernée.

***Quelques caractéristiques de Java***

* Une de ses plus grandes forces est son excellente portabilité : une fois votre programme créé, il fonctionnera automatiquement sous Windows, Mac, Linux, etc.
* On peut faire de nombreuses sortes de programmes avec Java :

- des applications, sous forme de fenêtre ou de console ;

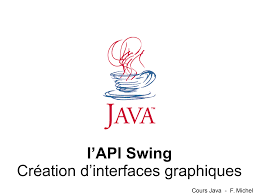
- des applets, qui sont des programmes Java incorporés à des pages web ;

- des applications pour appareils mobiles, avec J2ME , android;

- et bien d'autres ! J2EE, JMF, J3D pour la 3D...

Java est un langage fortement typé Un langage de programmation est dit fortement typé quand le compilateur vérifie que l’on ne fait avec les objets et les variables du programme que ce qui est autorisé par leur type. Cette vérification a pour effet d’accroître la fiabilité de l’exécution du programme. Java, C++ et C# sont fortement typés. L’étape de compilation y est essentielle

1. ***Présentation de Swing***

******

En Java, Swing fournit un ensemble de bibliothèques pour créer une interface graphique (GUI), pour le faire fonctionner indépendamment sur différentes plates-formes. Swing est certainement celui qui est le plus utilisé, car ses propriétés de légèreté, contrôle et de personnalisation facile.

Avant Swing, AWT(Abstract Windowing Toolkit) est celui qui permet de créer différentes interfaces graphiques. Cependant, en raison de ses inconvénients sur la plate-forme, lourd et peu de composants, il a été remplacé par Java Swing, qui est intégré au package AWT. Notez que certains composants AWT restent en Java et dans certaines situations.

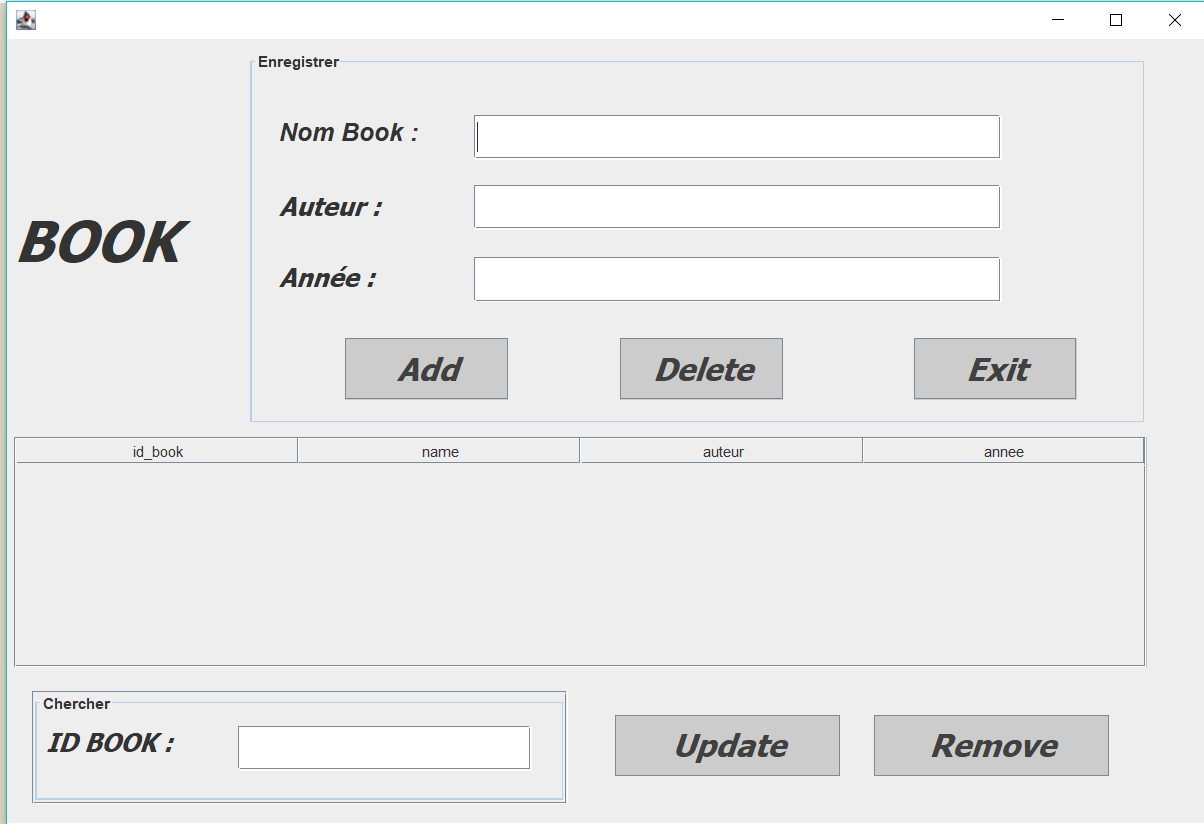
1. ***Présentation de Mini-Projet***

***Le but du projet :***  
Le but de notre projet consiste en l’élaboration d’une application  
desktop permettant la gestion d’un ensemble des livres et des librairies.  
L'application a été modélisé tel que je l'ai comprise et compte tenu des moyens temporels dont je dispose.

***Le sujet du projet :***  
Créer un portail de gestion des livres et des librairies (Gestion Book, Book Store ) : Ajout de nouveaux livres , leur attribuer des librairies, permettre aux utilisateurs d’avoir les livres et les librairies via une interface Desktop.

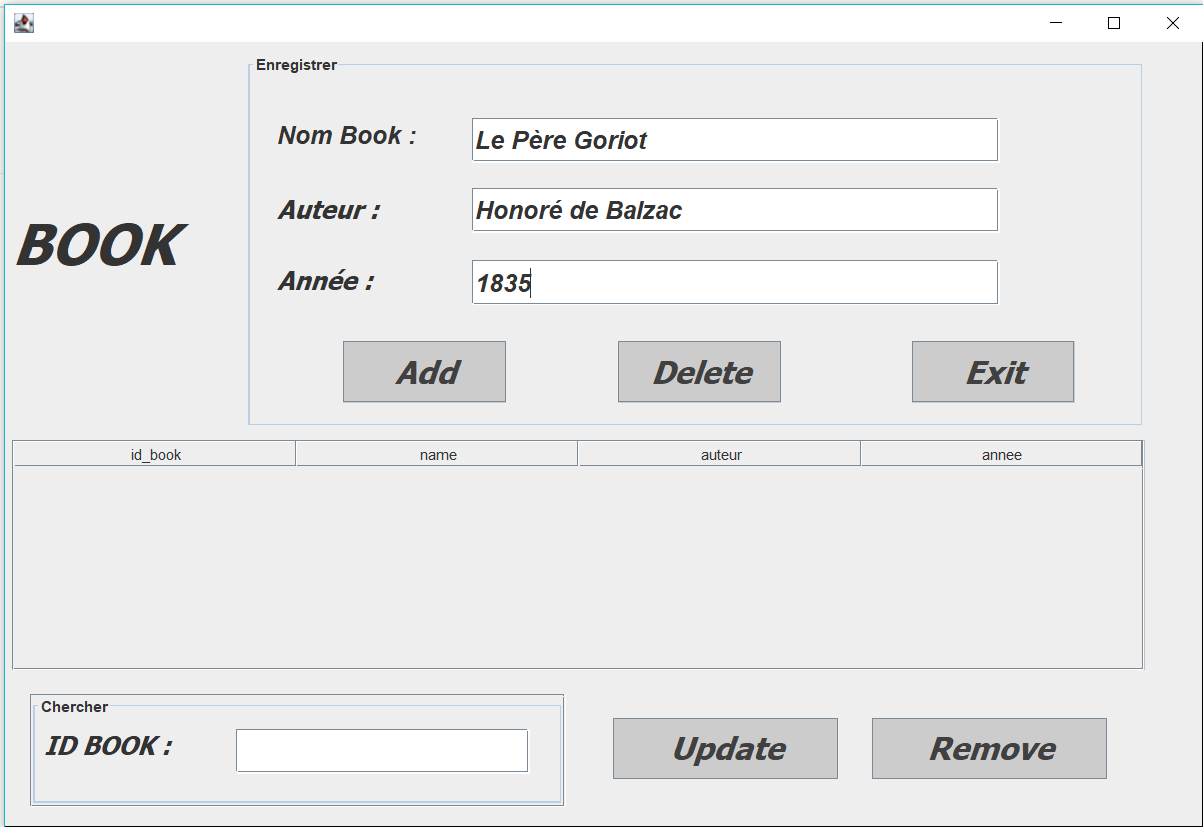
***La présentation de notre projet :***

* Voici le premier Interface qui géré les livres :
* ***Ajouter un Livre :***

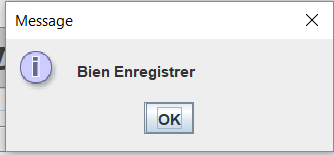


Maintenant en passe au cas où l’utilisateur a choisi d’ajouter un livre, l’utilisateur peut

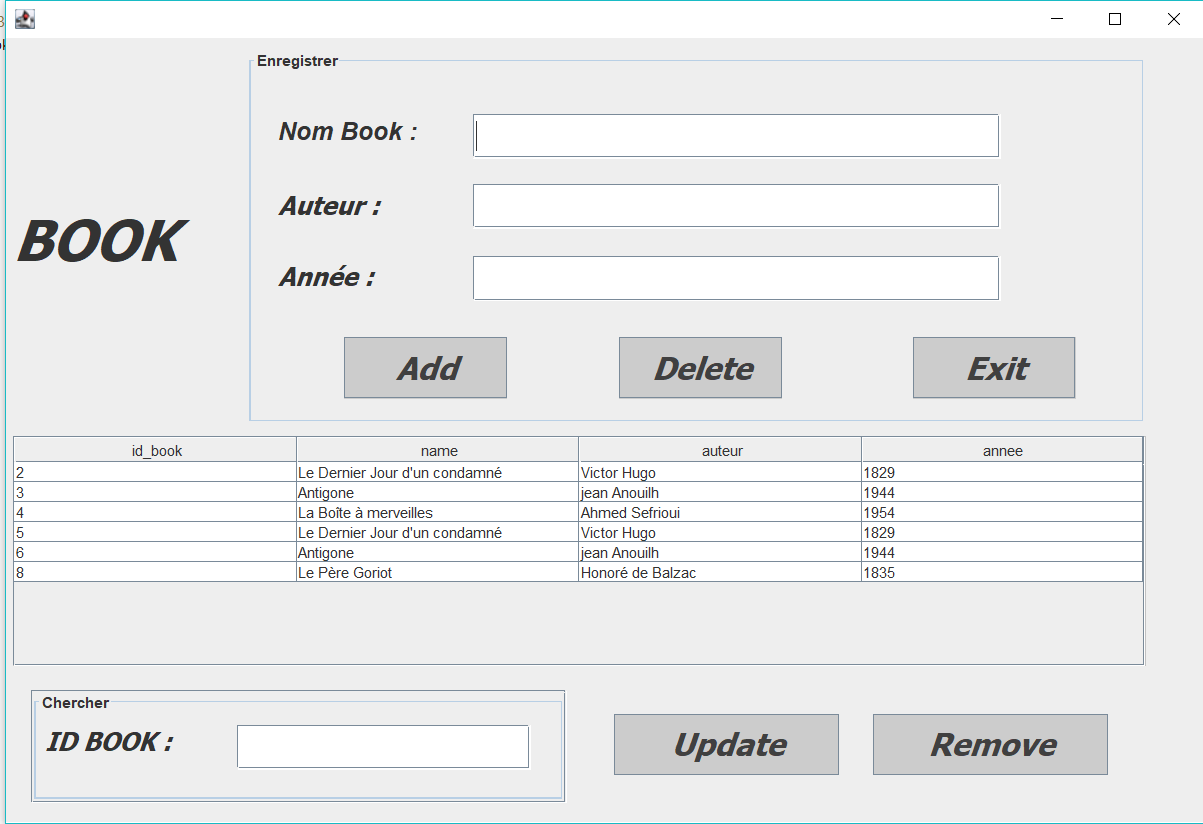
Saisie les données de livre, On voir cette exemple :



Après l’utilisateur clique sur le bouton « Add » , il recevoir ce message de confirmation :



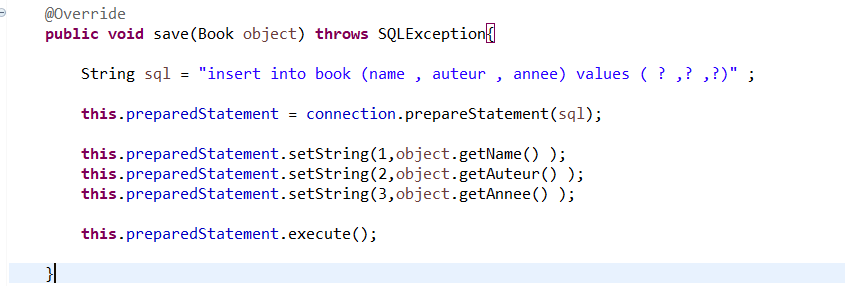
Après ce choix avec bien sûr des exemples significatifs qui seront ajouter par la suite directement dans notre base de données nous verrons que l’interface est la même que de la création de livre c’est le même fonctionnement et la même partie du code :



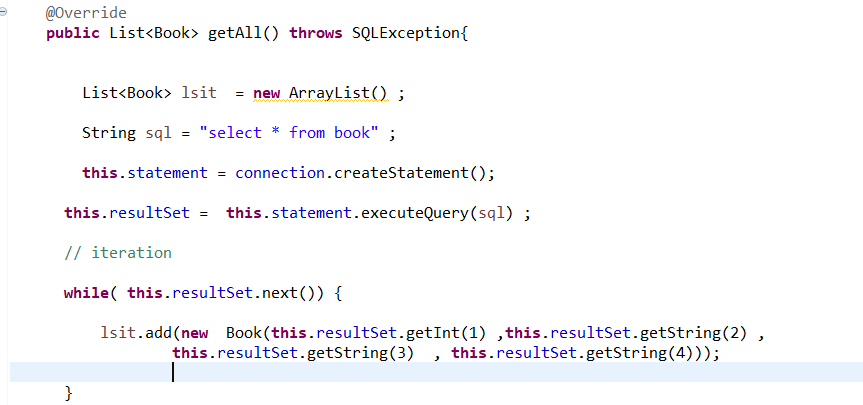
L’opération est bien validée et pour vérifier le fonctionnement  
voici notre base de données qui contient les livres avec leur   
 Id :



La partie du code qui contient cette méthode qui permet d’ajouter un livre est la suivante :

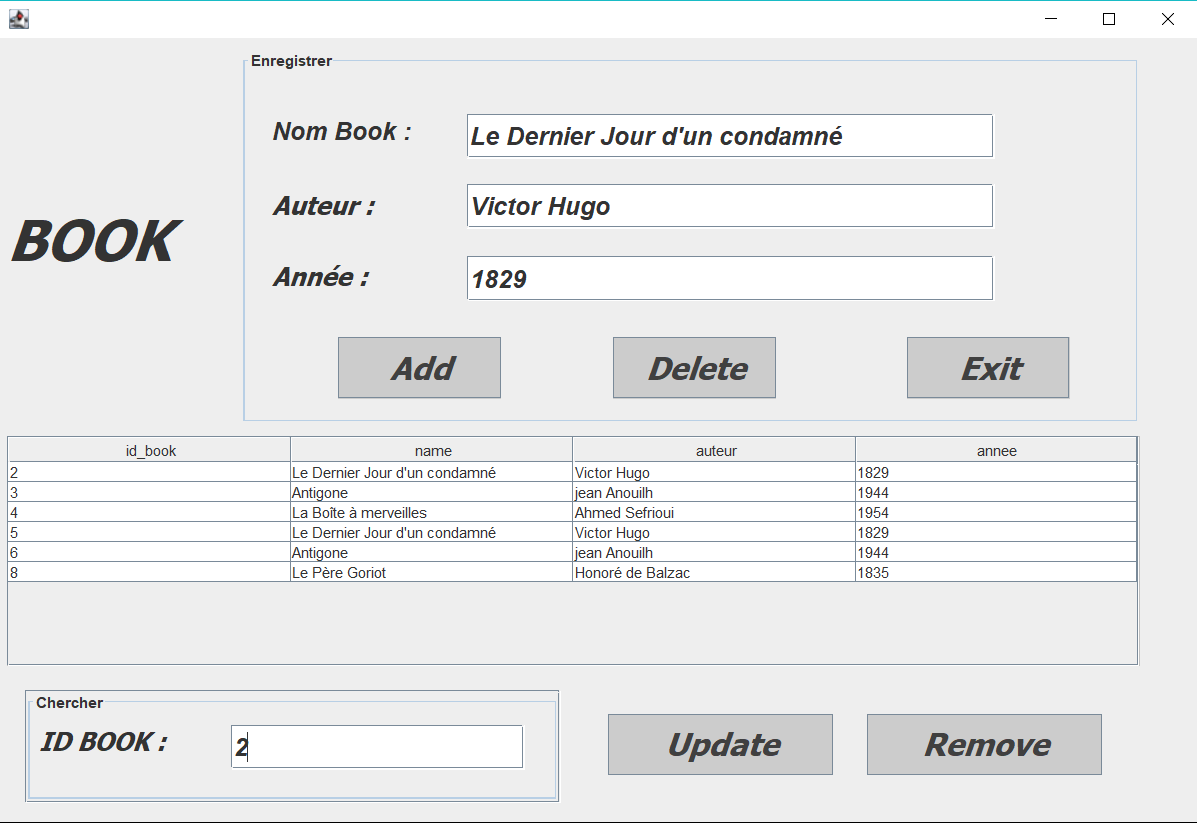


La partie du code qui permet d’afficher les informations des livres est la suivante :

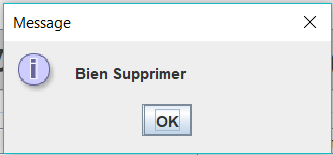


* ***Supprimer un Livre :***

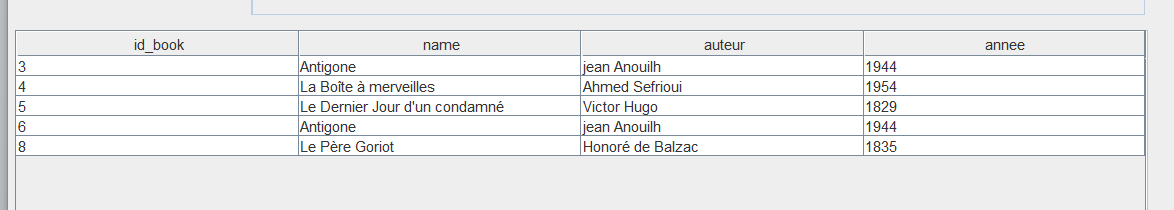
Passons maintenant au cas où l’utilisateur a choisi de supprimer un livre bien sûr il faut avoir une référence pour voir le livre consacre de la suppression il va entrer l’Id du livre voulu et voici ce qui sera afficher :



Après l’utilisateur clique sur le bouton « Remove » , il recevoir ce message de confirmation :



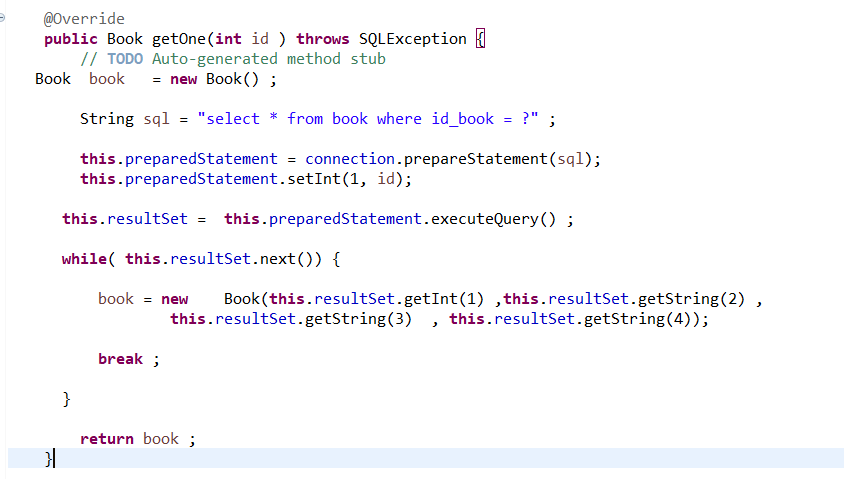
La suppression est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui ne contient plus le livre avec la Id :2



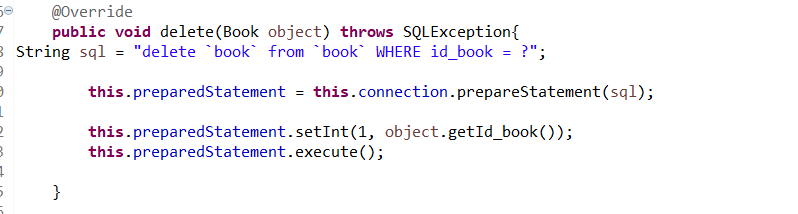


La partie du code qui contient cette méthode qui permet de supprimer un livre via son Id est la suivante :

-sélection de livre via son Id :

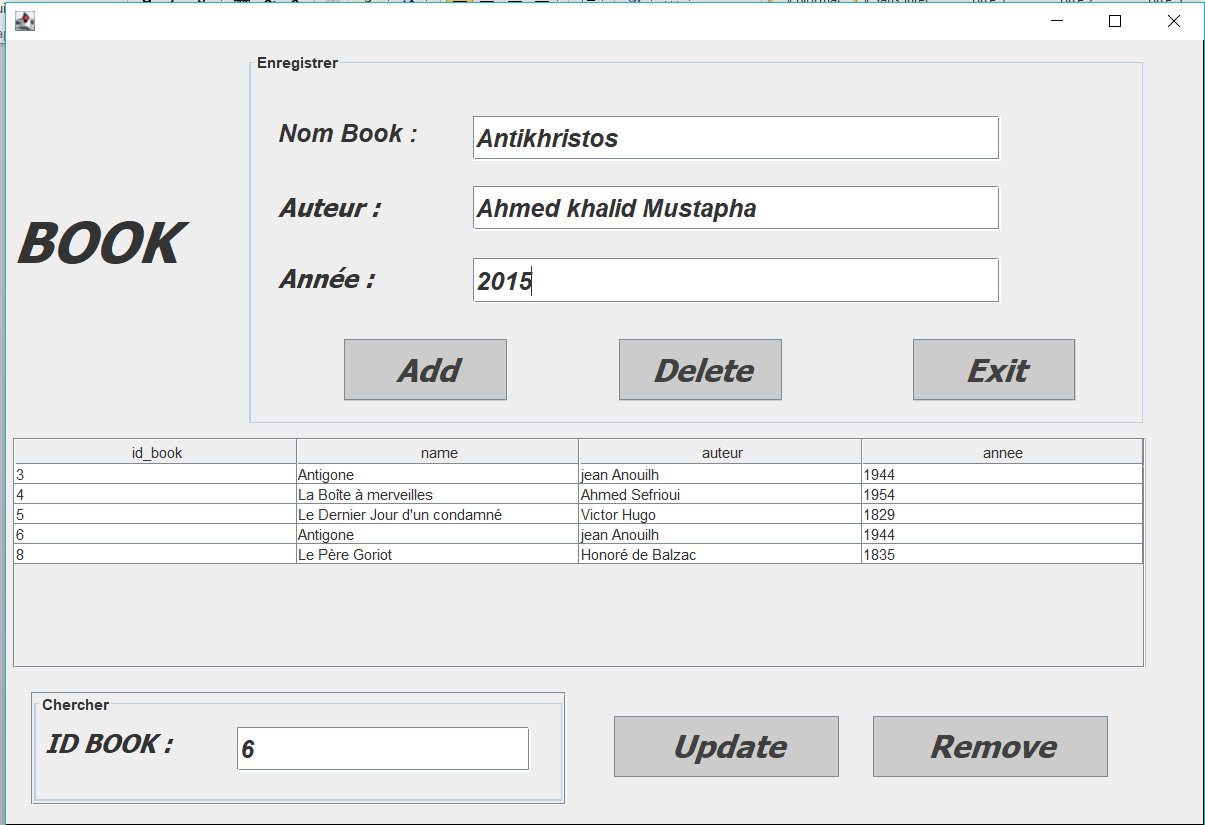


-supprimer de livre via son Id :

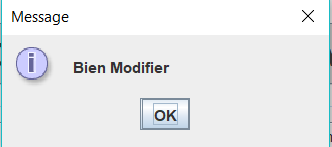


* ***Modifier un livre :***

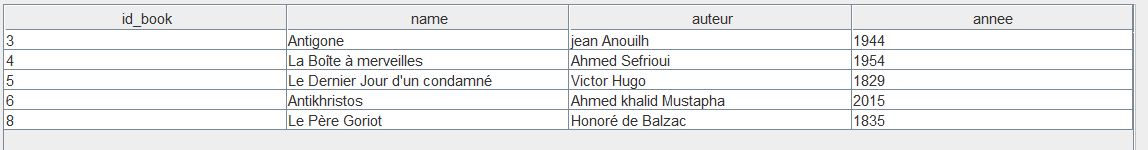
Passons maintenant au cas où l’utilisateur a choisi de modifier un livre bien sûr il faut avoir une référence pour voir le livre consacre de la modification il va entrer l’Id du livre voulu et voici ce qui sera afficher :



Après l’utilisateur clique sur le bouton « Update » , il recevoir ce message de confirmation :



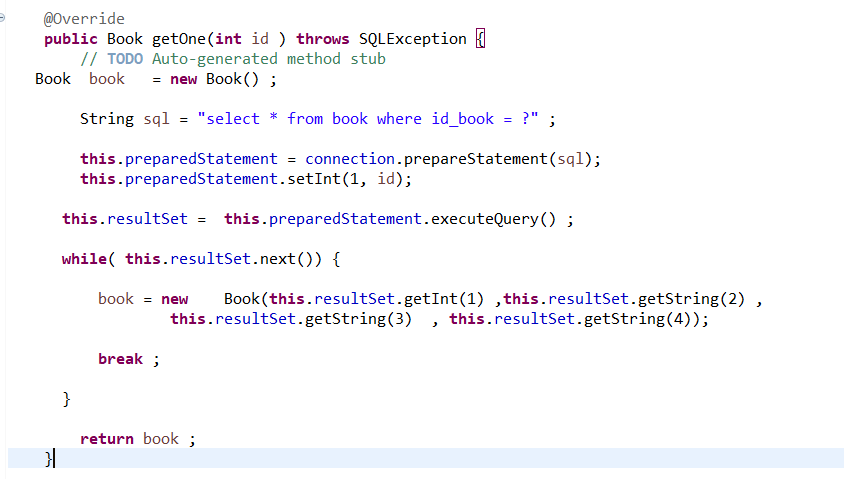
La modification est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui contient le livre avec la Id 6 après la modification :



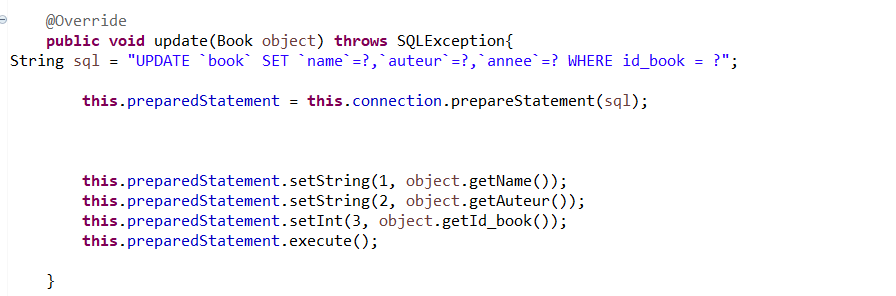


La partie du code qui contient cette méthode qui permet de modifier un livre via son Id est la suivante :

-sélection le livre via son Id :

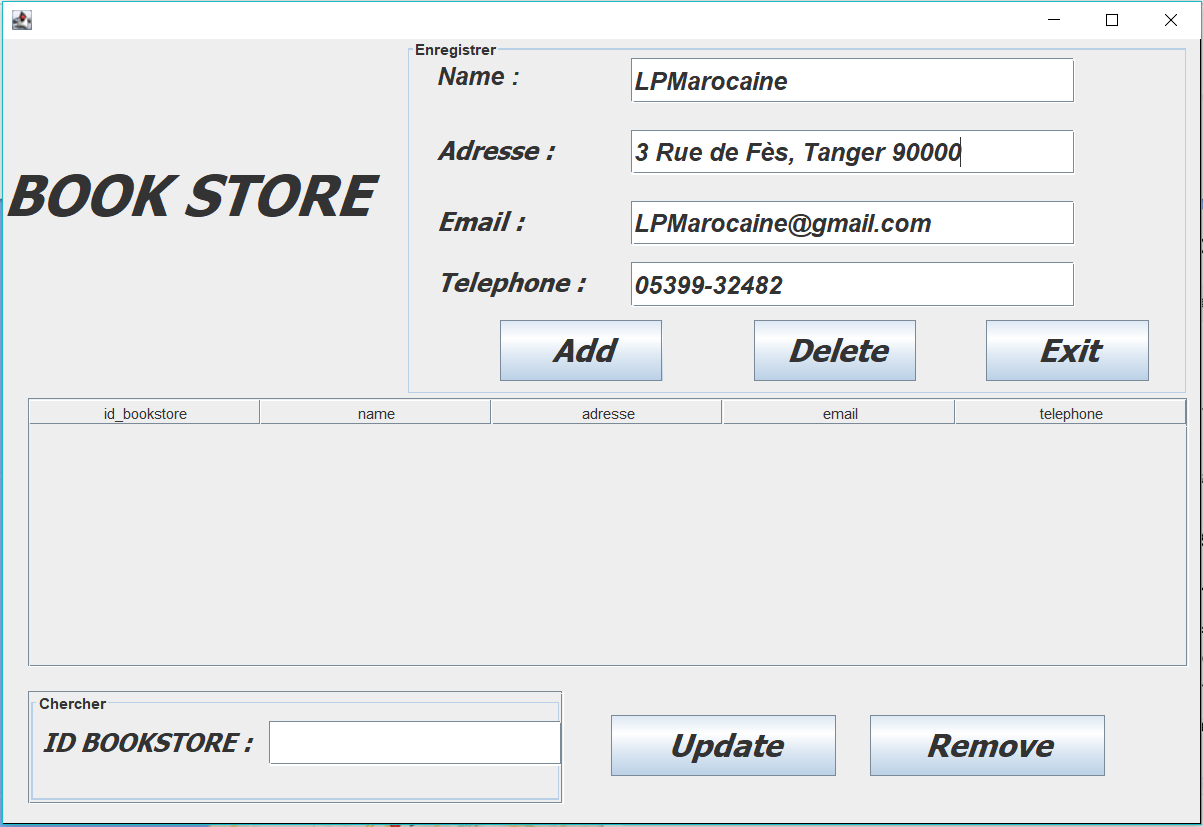


-modifier le livre via son Id :

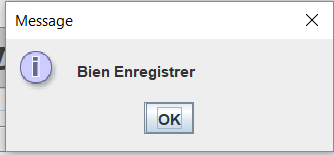


* Voici deuxième Interface qui géré les librairies :
* ***Ajouter une librairie :***

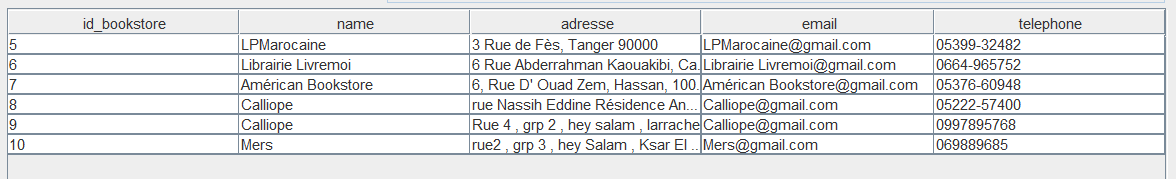
Maintenant en passe au cas où l’utilisateur a choisi d’ajouter une librairie, l’utilisateur peut

Saisie les données de librairie, On voir cette exemple : 

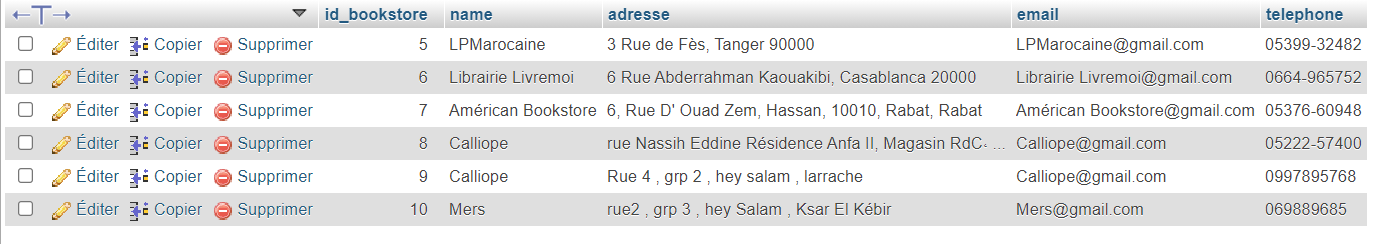
Après l’utilisateur clique sur le bouton « Add » , il recevoir ce message de confirmation :



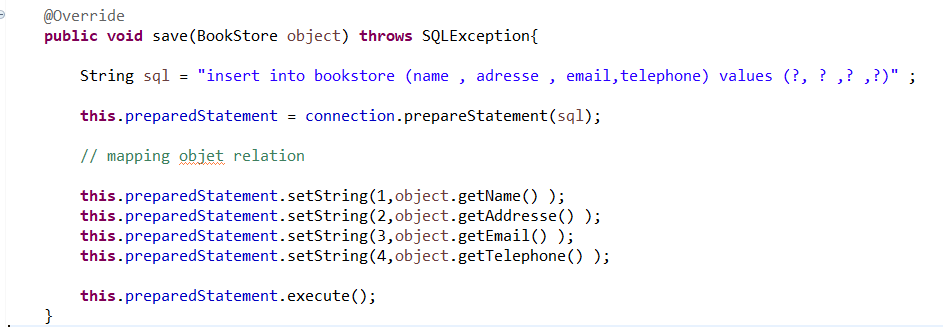
Après ce choix avec bien sûr des exemples significatifs qui seront ajouter par la suite directement dans notre base de données nous verrons que l’interface est la même que de la création de librairie c’est le même fonctionnement et la même partie du code :



L’opération est bien validée et pour vérifier le fonctionnement  
voici notre base de données qui contient les librairies avec leur   
 Id :



La partie du code qui contient cette méthode qui permet d’ajouter une librairie est la suivante :

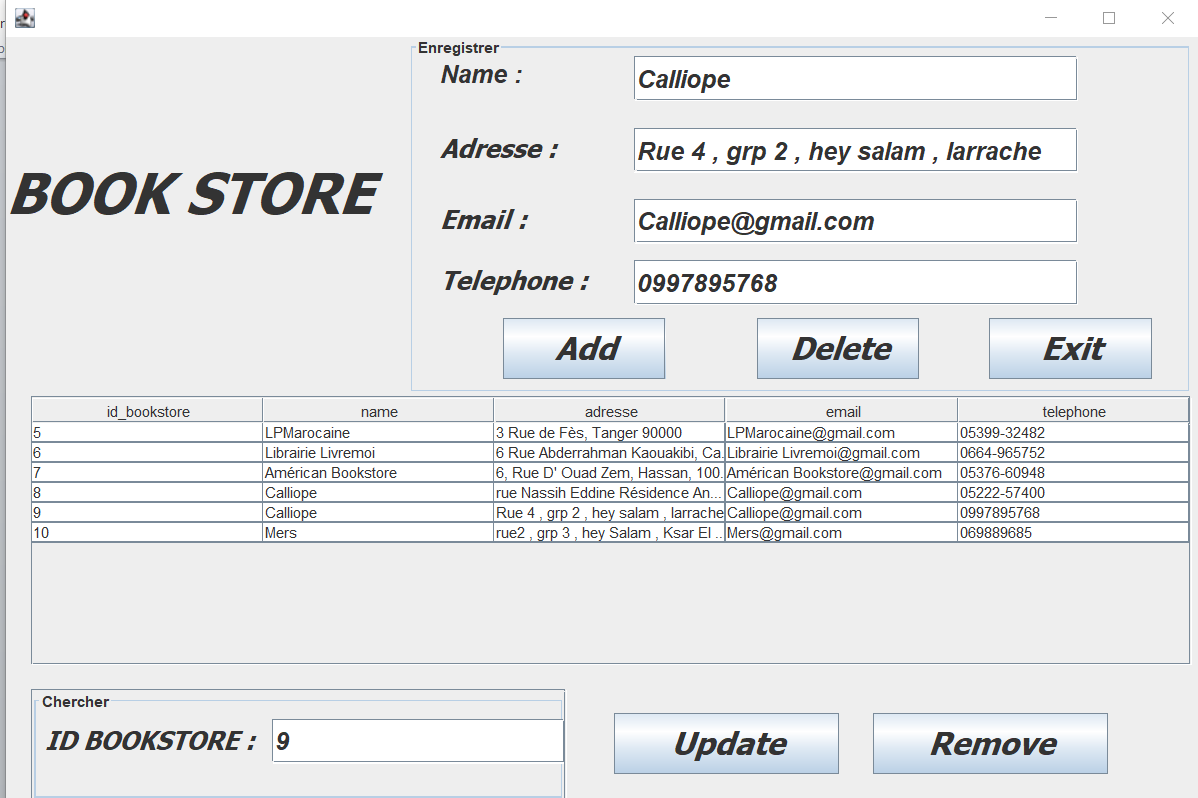


La partie du code qui permet d’afficher les informations de librairie est la suivante :

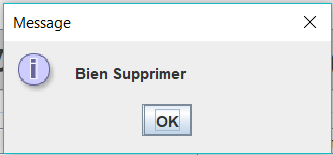


* ***Supprimer une Librairie :***

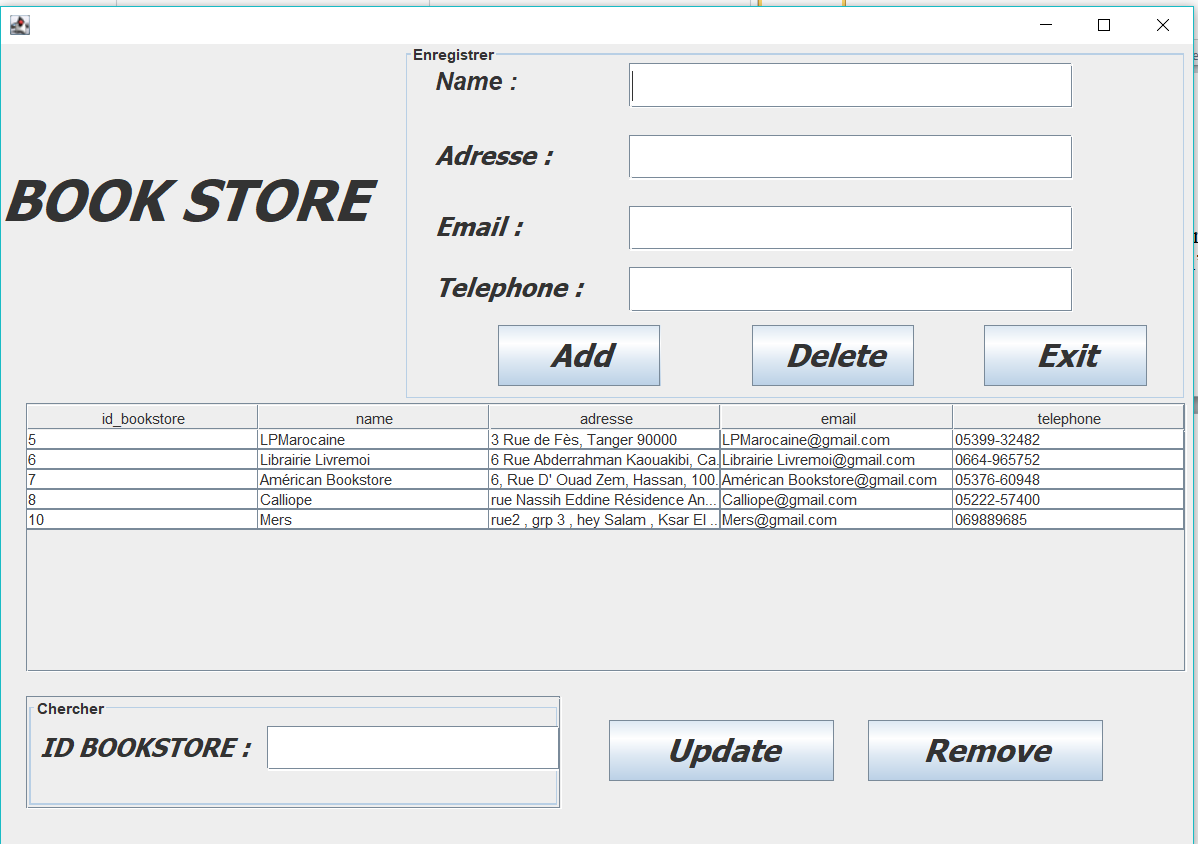
Passons maintenant au cas où l’utilisateur a choisi de supprimer une librairie bien sûr il faut avoir une référence pour voir la librairie consacre de la suppression il va entrer l’Id du librairie voulu et voici ce qui sera afficher :

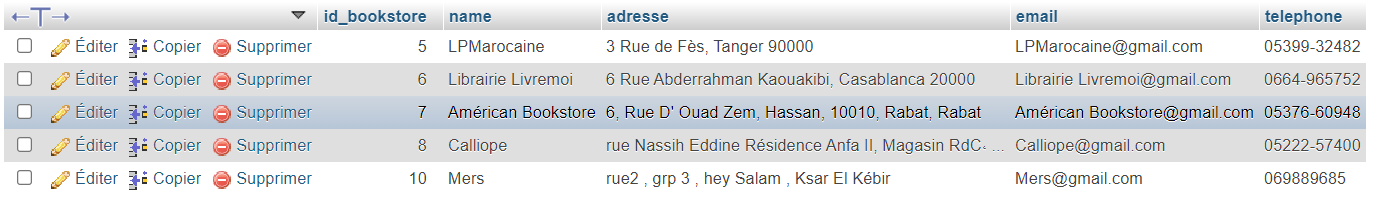


Après l’utilisateur clique sur le bouton « Remove » , il recevoir ce message de confirmation :



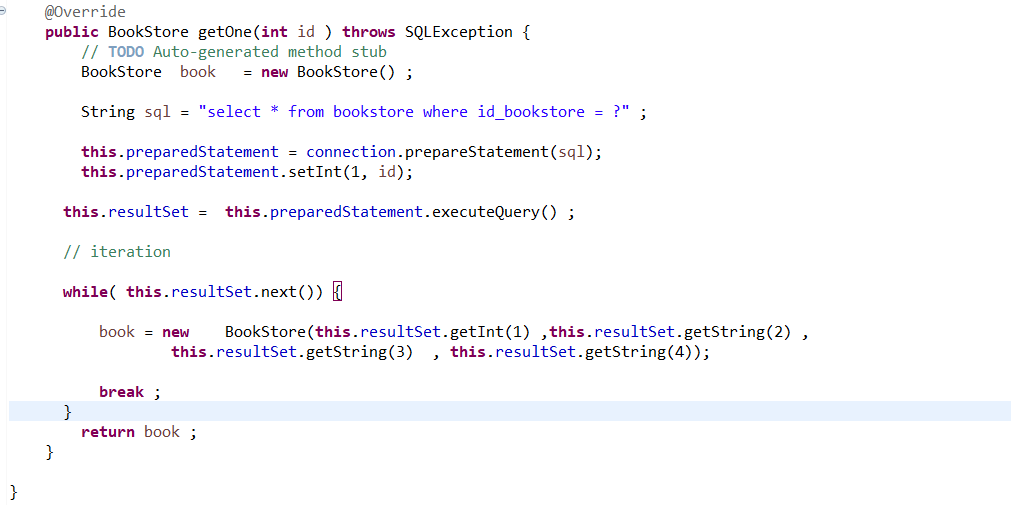
La suppression est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui ne contient plus le livre avec la Id :9



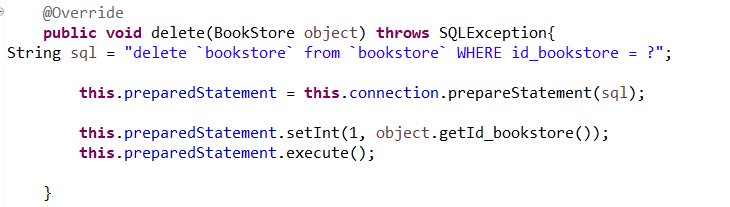


La partie du code qui contient cette méthode qui permet de supprimer un livre via son Id est la suivante :

-sélection de librairie via son Id :

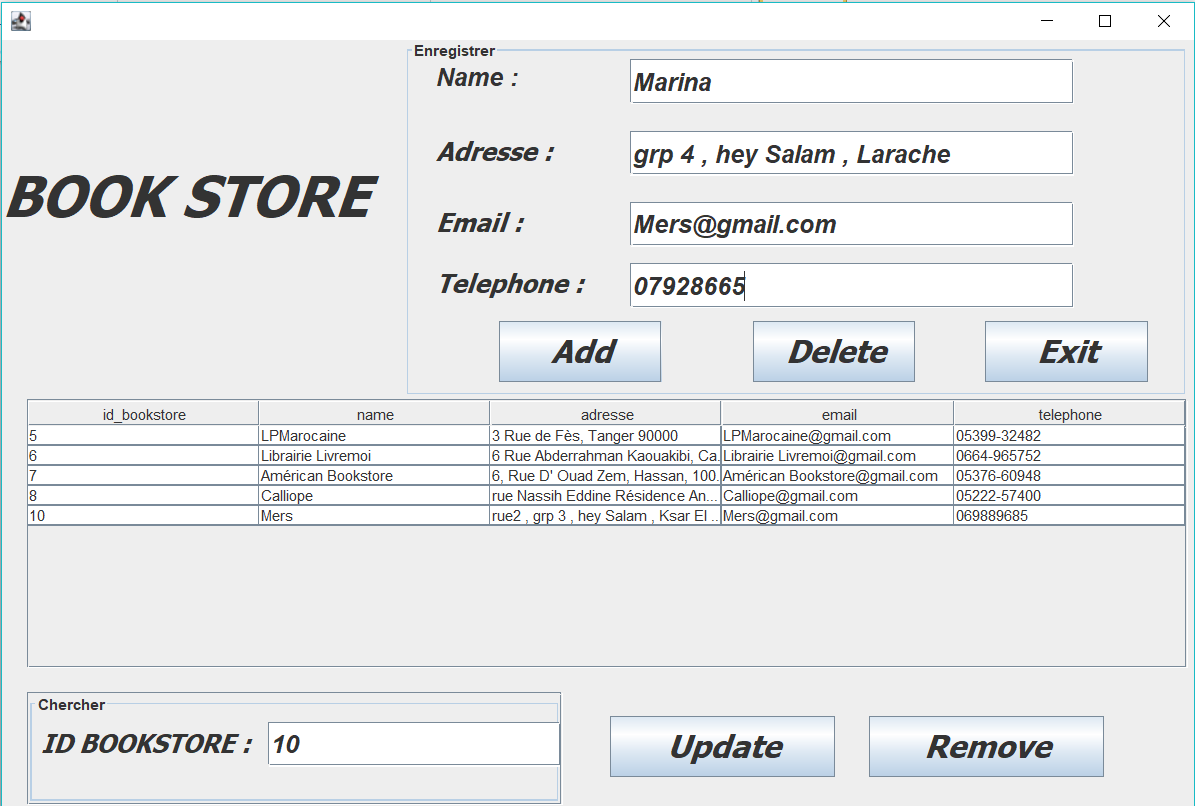


-supprimer de librairie via son Id :

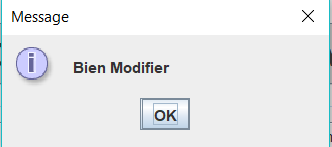


* ***Modifier une librairie :***

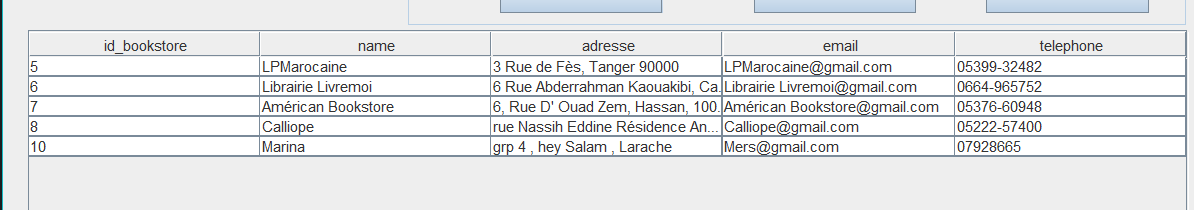
Passons maintenant au cas où l’utilisateur a choisi de modifier une librairie bien sûr il faut avoir une référence pour voir la librairie consacre de la modification il va entrer l’Id du librairie voulu et voici ce qui sera afficher :

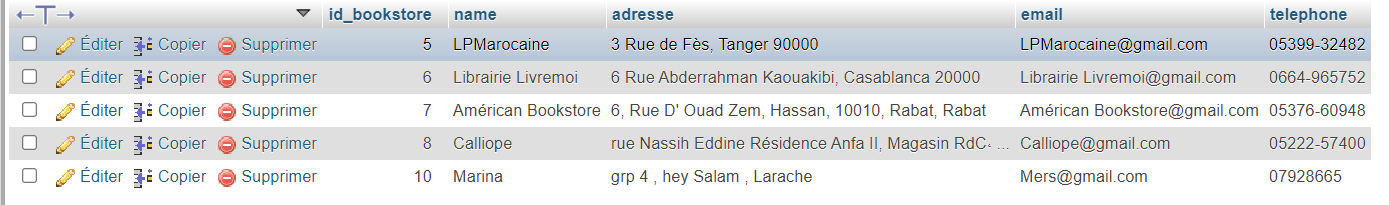


Après l’utilisateur clique sur le bouton « Update » , il recevoir ce message de confirmation :



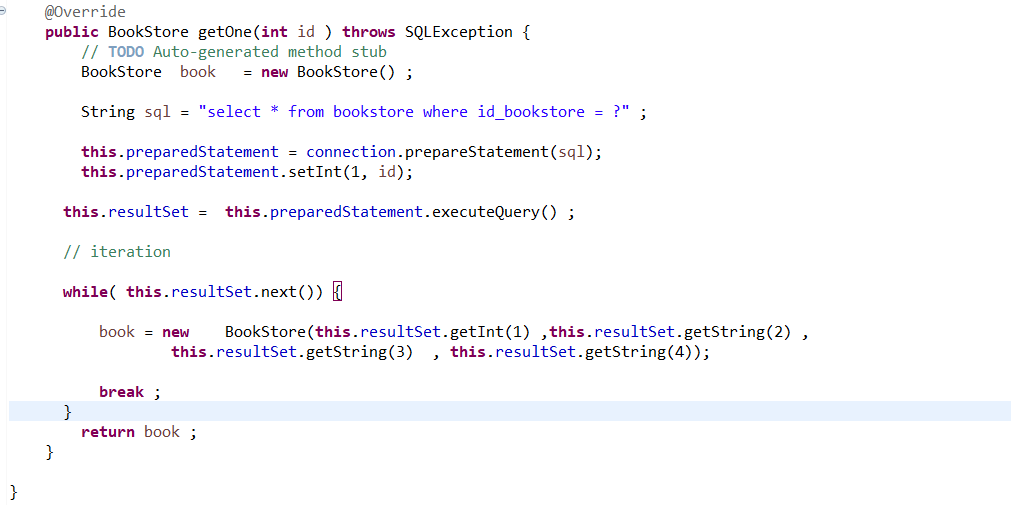
La modification est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui contient le livre avec la Id 10 après la modification :



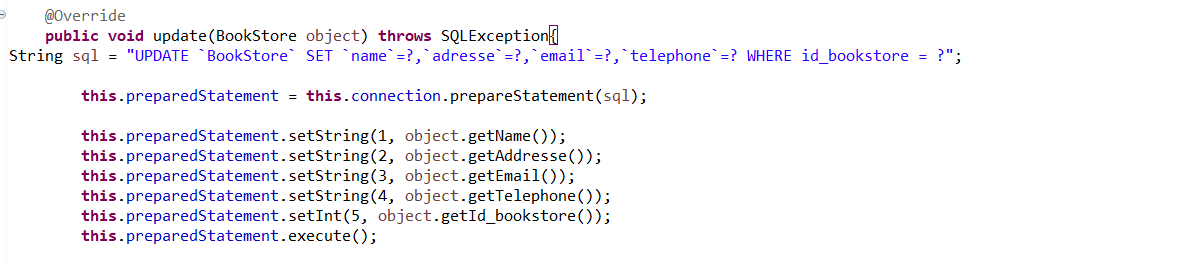


La partie du code qui contient cette méthode qui permet de modifier un livre via son Id est la suivante :

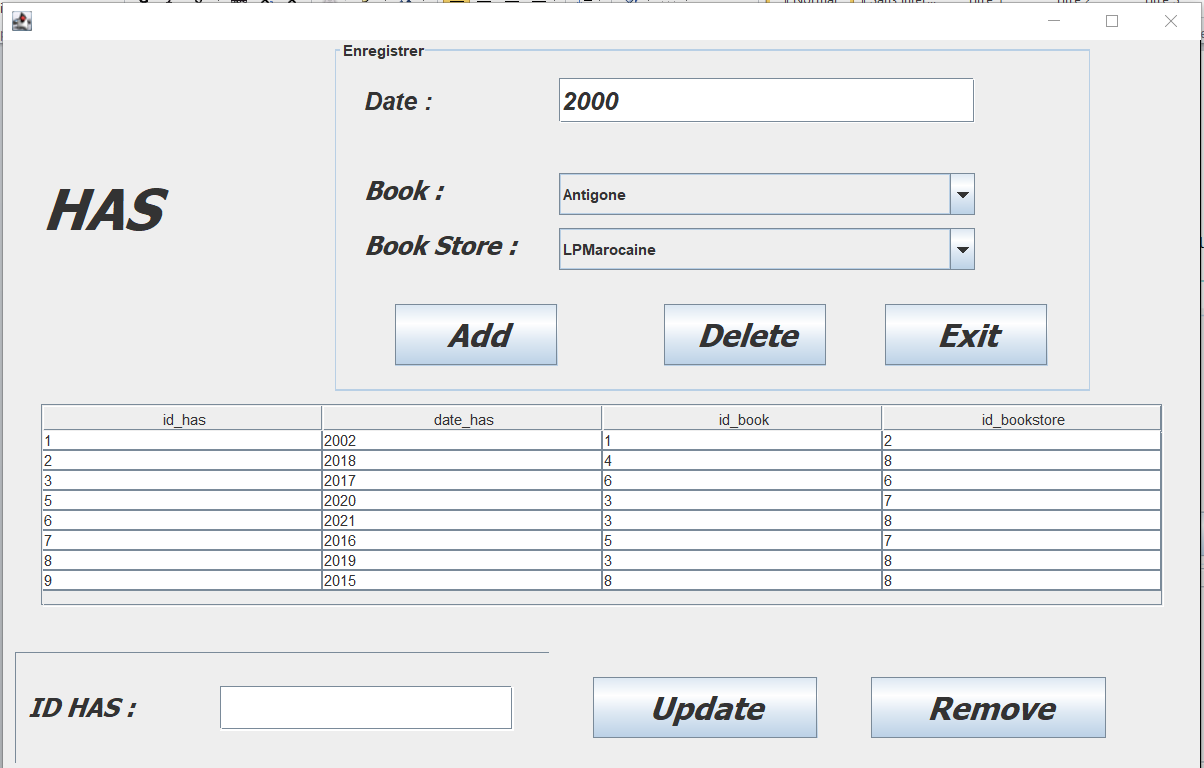
-sélection la librairie via son Id :



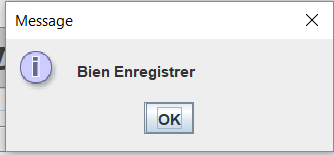
-modifier la librairie via son Id :



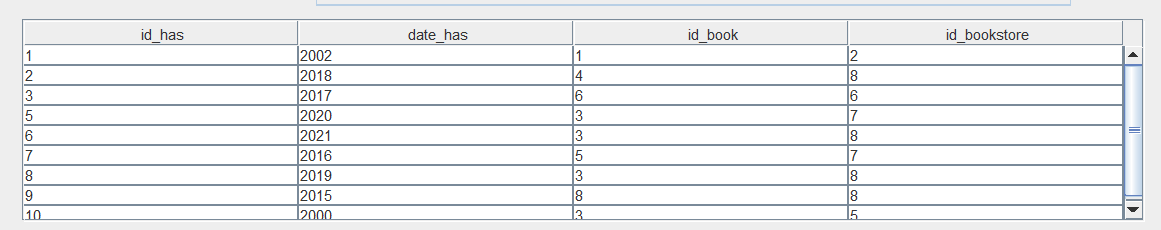
* Voici Troisième Interface qui géré les commandes :
* ***Ajouter une commande:***

Maintenant en passe au cas où l’utilisateur a choisi d’ajouter une commande, l’utilisateur peut Saisie les données de commande, On voir cette exemple : 

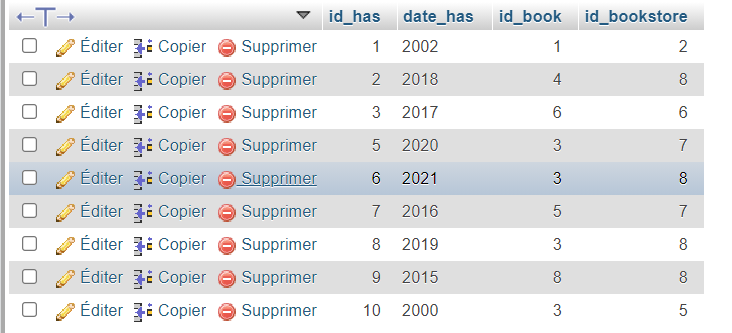
Après l’utilisateur clique sur le bouton « Add » , il recevoir ce message de confirmation :



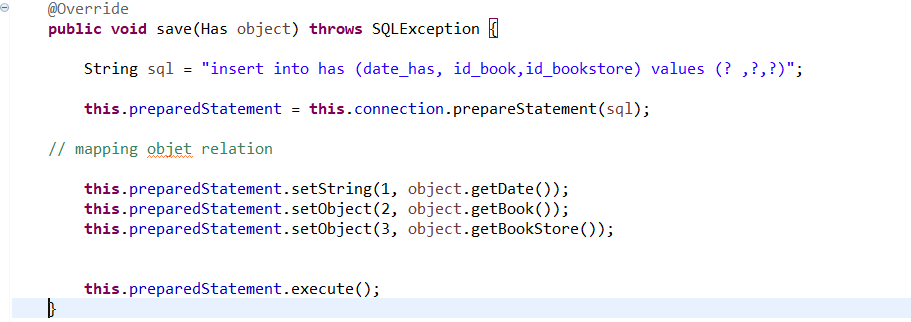
Après ce choix avec bien sûr des exemples significatifs qui seront ajouter par la suite directement dans notre base de données nous verrons que l’interface est la même que de la création de commande c’est le même fonctionnement et la même partie du code :



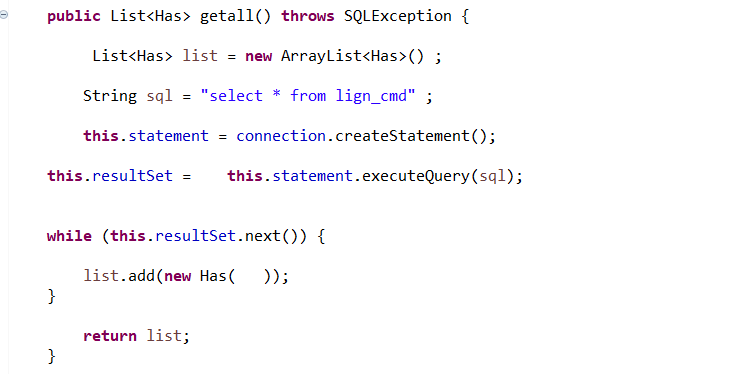
L’opération est bien validée et pour vérifier le fonctionnement  
voici notre base de données qui contient les commandes avec leur   
 Id :



La partie du code qui contient cette méthode qui permet d’ajouter une commande est la suivante :

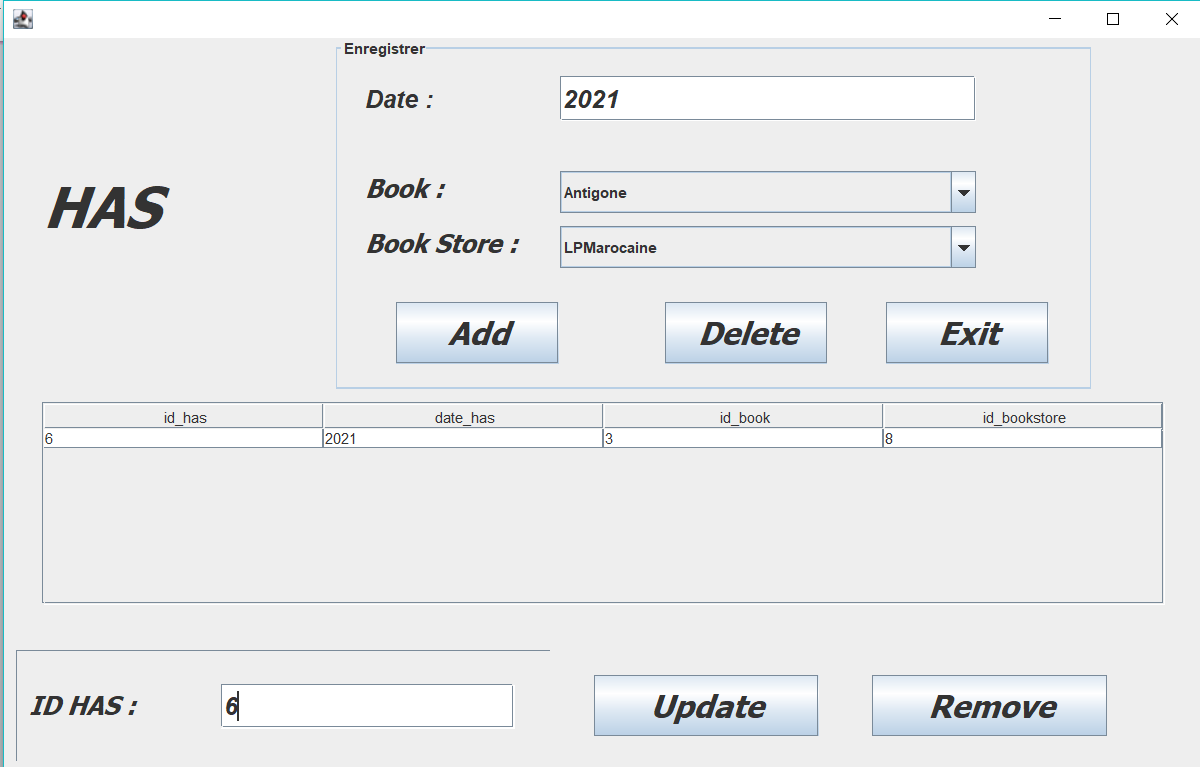


La partie du code qui permet d’afficher les informations des livres est la suivante :

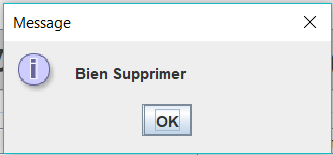


* ***Supprimer une commande:***

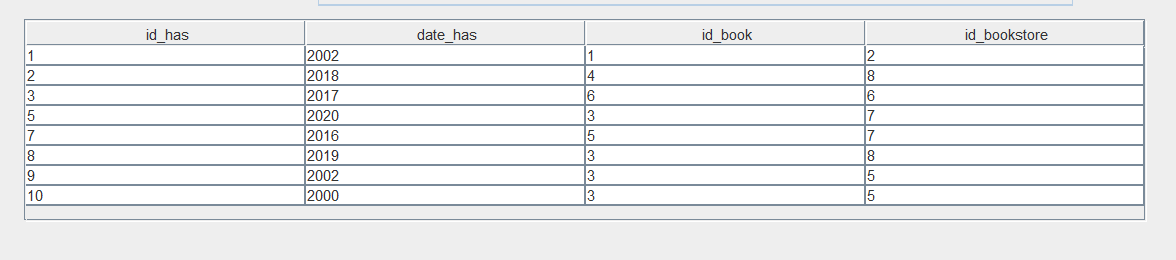
Passons maintenant au cas où l’utilisateur a choisi de supprimer une commande bien sûr il faut avoir une référence pour voir la commande consacre de la suppression il va entrer l’Id du commande voulu et voici ce qui sera afficher :

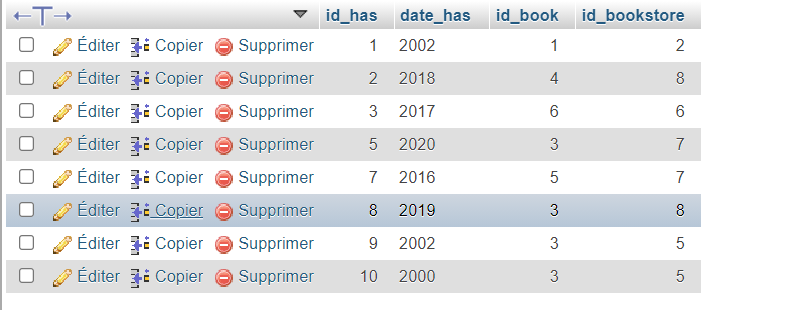


Après l’utilisateur clique sur le bouton « Remove » , il recevoir ce message de confirmation :



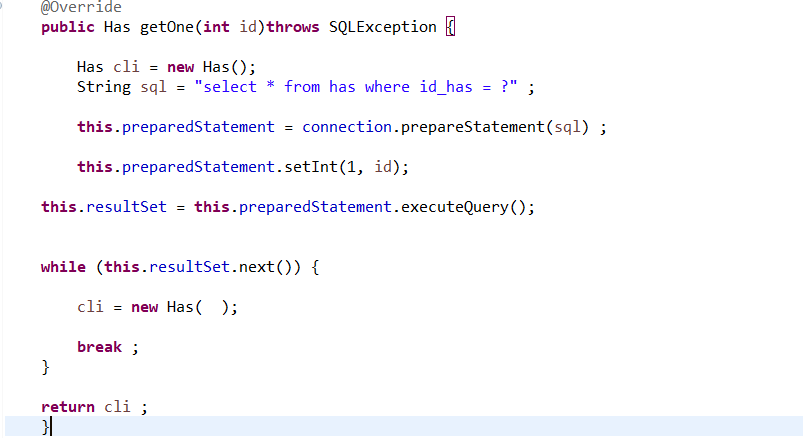
La suppression est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui ne contient plus le livre avec la Id :6



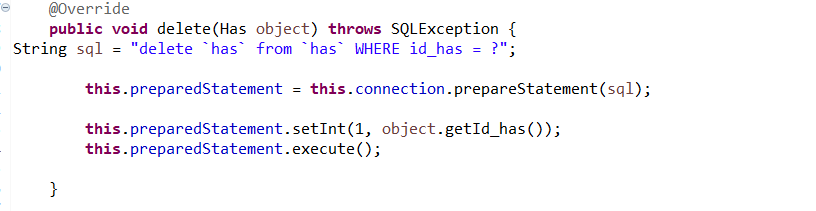


La partie du code qui contient cette méthode qui permet de supprimer une commande via son Id est la suivante :

-sélection de commande via son Id :

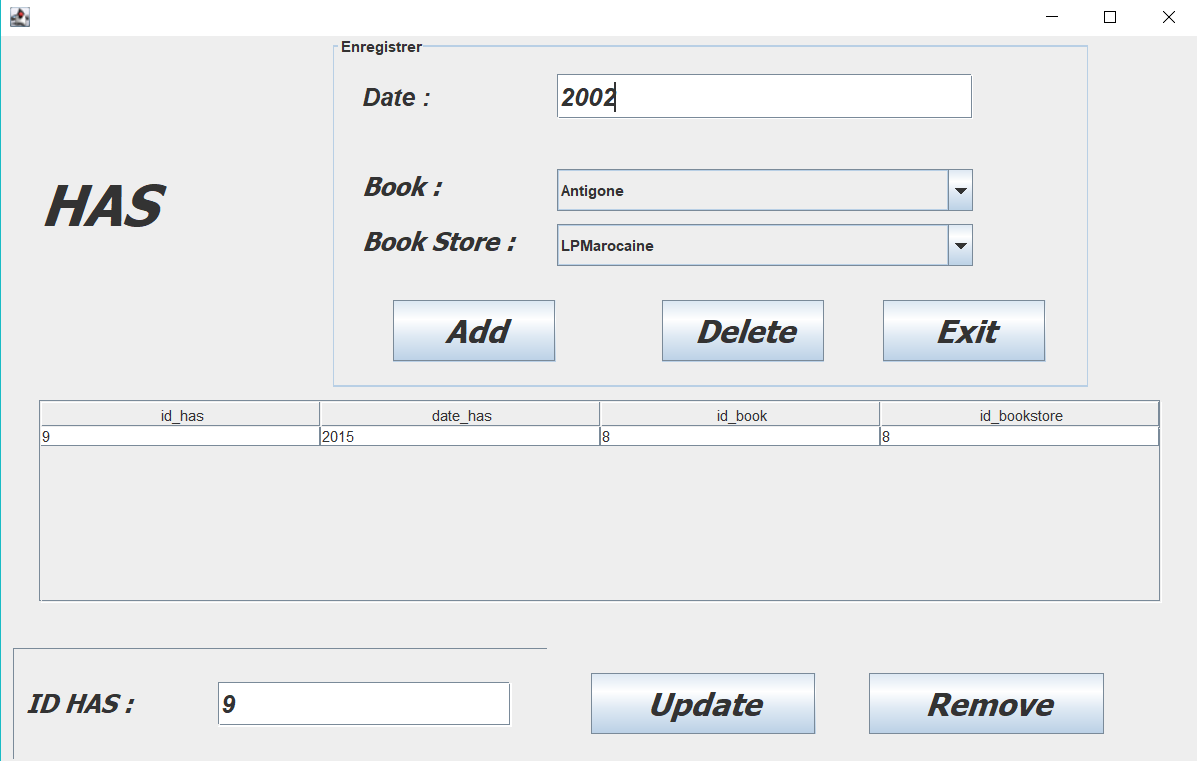


-supprimer du commande via son Id :

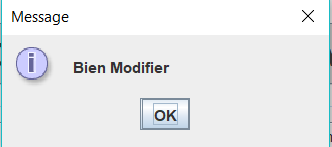


* ***Modifier une commande :***

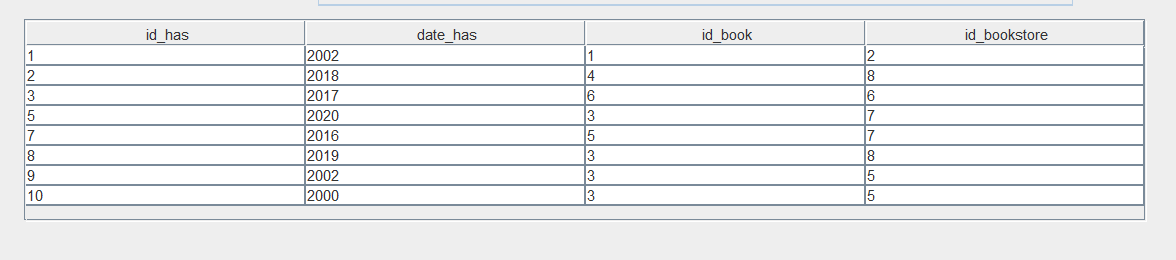
Passons maintenant au cas où l’utilisateur a choisi de modifier une commande bien sûr il faut avoir une référence pour voir la commande consacre de la modification il va entrer l’Id du commande voulu et voici ce qui sera afficher :

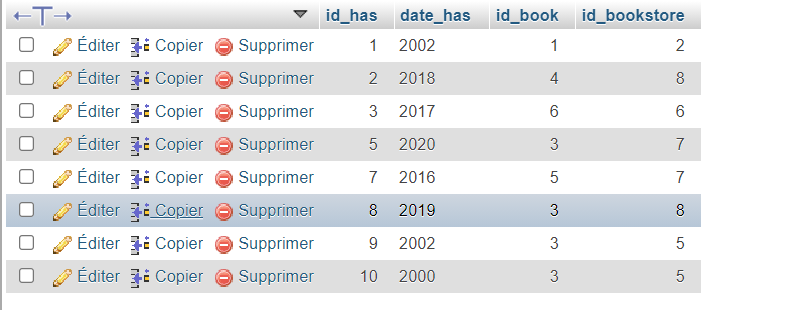


Après l’utilisateur clique sur le bouton « Update » , il recevoir ce message de confirmation :



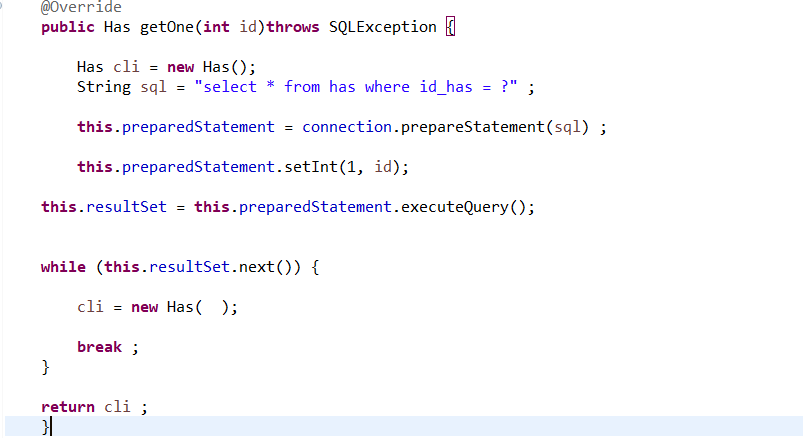
La modification est validée et pour vérifier le fonctionnement voici notre base de données et tableau d’affichage qui contient la commande avec la Id 9 après la modification :





La partie du code qui contient cette méthode qui permet de modifier une Commande via son Id est la suivante :

-sélection la commande via son Id :



-modifier la commande via son Id :

