

Java : Partie 4

Persistance de Données : Exercices

Mise à jour de Août 2013

Auteur : Chouaïb LAGHLAM

Sommaire

Préalable	2
Exercice N° 00700 : Lire un fichier texte	3
Exercice N° 00710 : Transformer un fichier texte.....	3
Exercice N° 00720 : Écrire dans un fichier texte	4
Exercice N° 00730 : Lire un fichier	5
Exercice N° 00740 : Lire une BD.....	6
Exercice N° 00750 : Lire et Écrire dans une BD	6
Exercice N° 00760 : utiliser une BD Oracle.....	6
Exercice N° 00770 : Importer Excel dans une BD	6
Exercice N° 00780 : Paramètres pour accès Multi-Bases.....	7

Préalable

- ➔ Dans cette partie «Persistance de données», l’affichage, dans les différents exercices, se fait sur la **console** (Une fenêtre Dos sous Windows).
- ➔ Créez dans votre espace de travail (Workspace), et dans le projet nommé « **prj_Java_030_Persistance**», Le package «**pack_exos**»
- ➔ Chaque exercice doit être codé dans une classe dotée de la méthode main et nommée du nom de l’exercice.

Exemple :

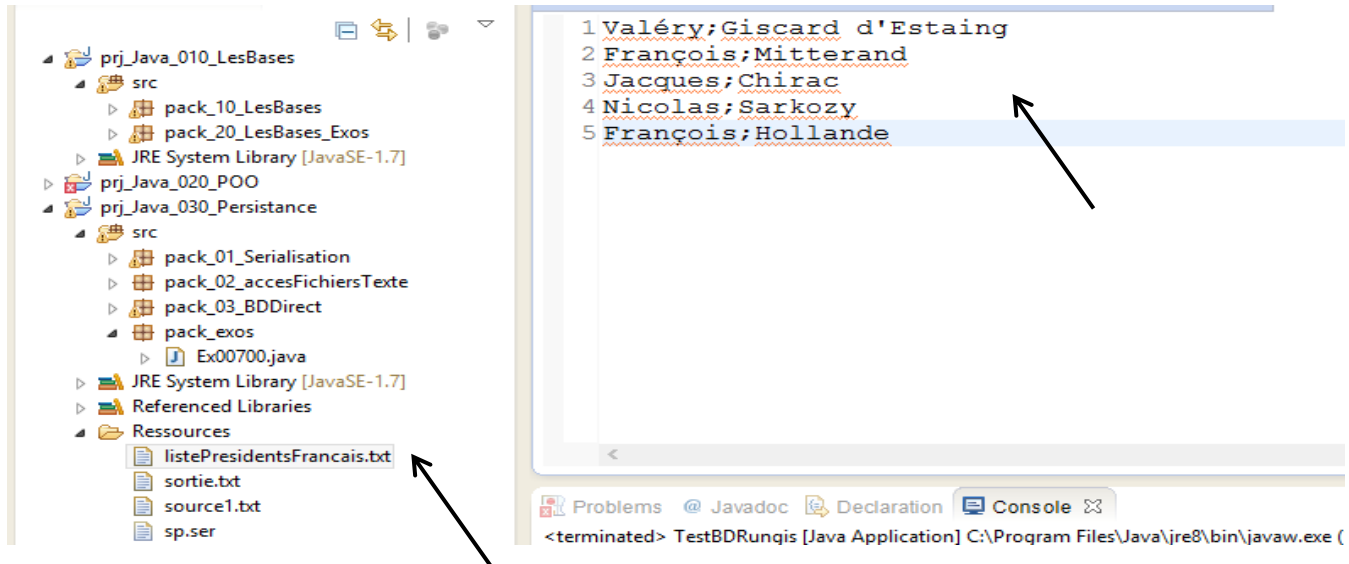
Exercice 700 :

- Il y aura la classe java : Exercice 00700.java avec la méthode main,
- Il y aura éventuellement d’autres classes à créer : elles porteront le nom demandé.

- ➔ Vous allez manipuler des fichiers texte, Excel, ... : créez un dossier « **Ressources** » dans votre projet Et placez tous ces fichiers dans ce dossier.

Exercice N° 00700 : Lire un fichier texte

- 1) Créez dans «Ressources», un fichier texte «listePresidentsFrancais.txt» ainsi :



- 2) Codez la méthode main afin qu'elle affiche sur la console, les présidents contenus dans le fichier ci-dessus Évidement : vous séparerez le nom et le prénom et donc vous n'afficherez pas la virgule.

Aide : pensez à la méthode split pour les chaînes de caractères

Exercice N° 00710 : Transformer un fichier texte

Votre travail va consister à transformer le contenu d'un fichier d'entrée «clients1.txt» en un fichier « clients2.txt» Le fichier d'entrée contient 5 lignes par client :

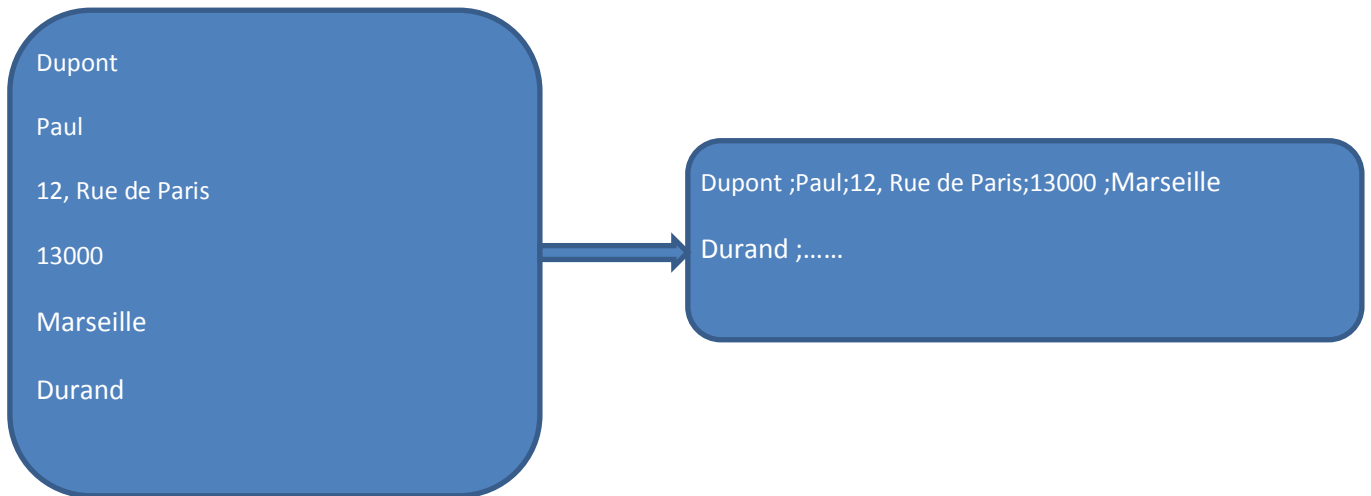
- Une ligne pour décrire le nom,
- Une ligne pour décrire le prénom,
- Une ligne pour décrire l'adresse,
- Une ligne pour décrire le code Postal,
- Une ligne pour décrire la ville.

Le fichier de sortie doit contenir une ligne par client : chaque ligne contient les 5 infos citées ci-dessus séparées par un point-virgule.

Java : Partie 4

Persistence de Données : Exercices

Exemple :



Exercice N° 00720 : Écrire dans un fichier texte

Vous faites un code en boucle qui enregistre les participants à un marathon :

- ➔ Demandez à l'utilisateur : nom, prénom, sexe (f ou h), âge du participant,
- ➔ Ajoutez au fichier texte **«participants.txt»** : une ligne qui contiendra les infos ci-dessus + La date système + l'heure système,
- ➔ Lorsque l'utilisateur a fini la saisie : affichez le contenu du fichier sur la console.

Exemple :

```
<terminated> Ex00720 [Java Application] C:\Program Files\Java\jre8\bin\javaw.exe (14 juil. 2014 16:03:39)
=====
Marathon de Paris
=====
nom participant : (quit pour finir)
MARTIN
prénom participant :
Olivier
sexe participant (f ou h):
h
age participant :
21
nom participant : (quit pour finir)
BOULANGER
prénom participant :
Anne
sexe participant (f ou h):
f
age participant :
24
nom participant : (quit pour finir)
quit
```

Java : Partie 4

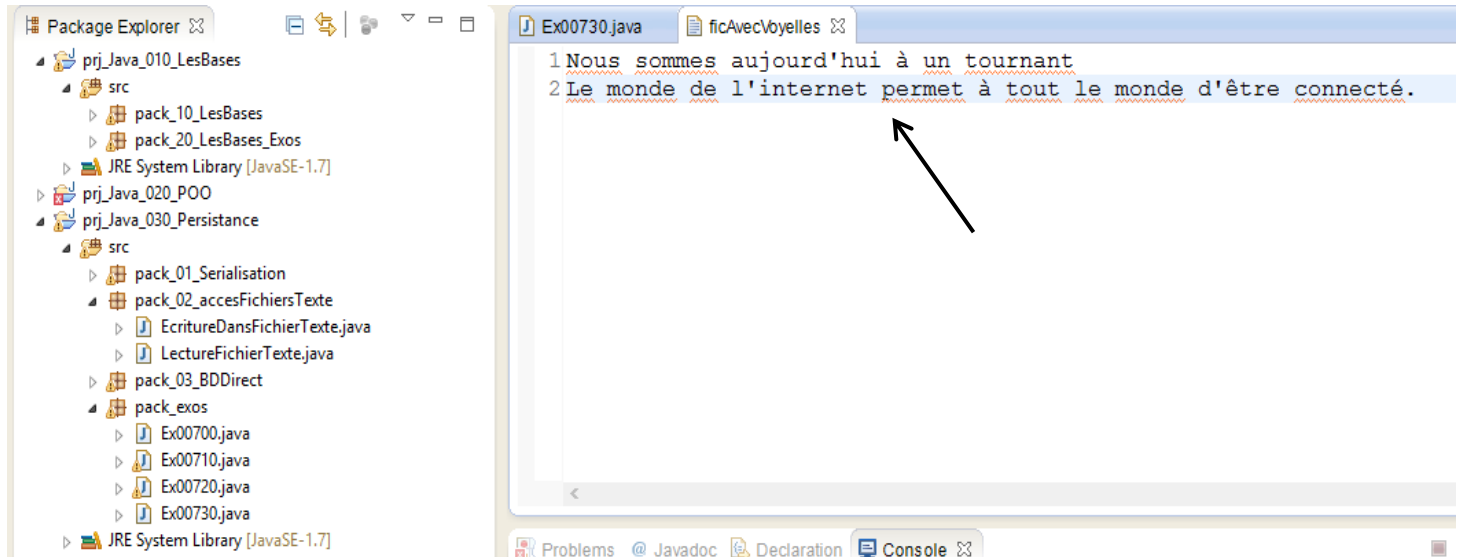
Persistance de Données : Exercices

Exercice N° 00730 : Lire un fichier

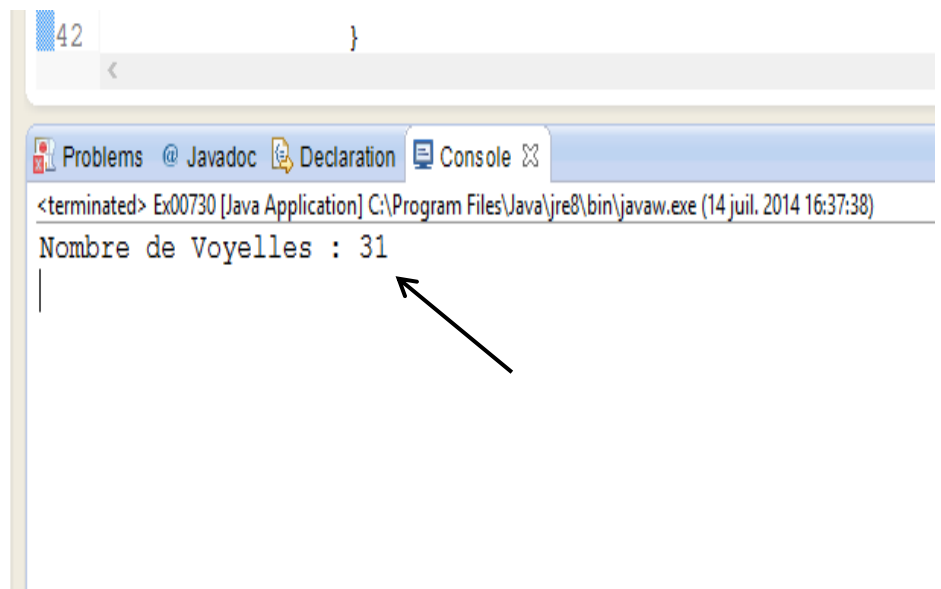
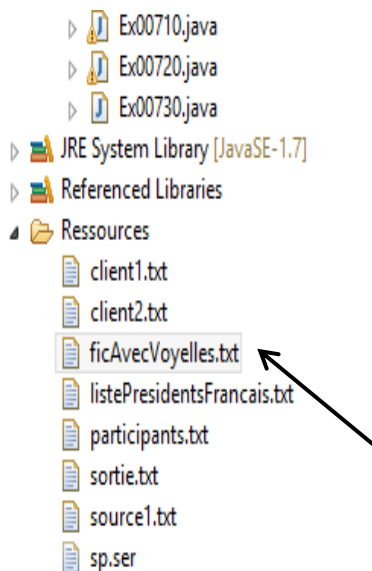
Remplissez à la main un fichier texte «**article.txt**» avec quelques lignes de texte de votre choix, Lisez ce fichier pour compter le nombre de voyelles qu'il contient et affichez ce nombre.

Exemple :

Avec un fichier contenant :



Cela donne :



Exercice N° 00740 : Lire une BD

- 1) Créez dans MySQL, une BD nommée «**BDTest**» et créez dedans une table :

Nom table	Nom colonne	Type	Clé primaire	Auto incrémentée
Tennismen	ID	Int	Oui	Oui
Tennismen	Nom	Varchar		
Tennismen	Prenom	Varchar		
Tennismen	Age	Int		

- 2) Remplissez cette table par des informations sur des tennismen actuels (évitez des titi, tata, dupont,)
3) Dans votre code Java, affichez sur la console, le contenu de la table SQL ci-dessus.

Exercice N° 00750 : Lire et Écrire dans une BD

- 1) Créez dans SQL Server 2012, la BD et la table créées ci-dessus dans MySQL : la table dans SQL Server doit rester vide,
2) Dans votre code java, lisez la table de MySQL pour la copier dans celle de SQL Server.
3) Prévoir de rejeter les lignes doublons : pour cela exécutez :
a. Votre code une première fois : il remplit la table SQL Server,
b. Votre code une fois encore : les lignes sont rejetées : affichées sur la console.

Exercice N° 00760 : utiliser une BD Oracle

Dans le cours, vous avez utilisé une classe java « ParametresBD.java » qui paramètre les accès à MySQL et à SQL Server 2012.

Vous êtes chargé ici de :

- 1) Trouvez et téléchargez le JDBC pour Oracle 11g et plus,
2) Complétez la classe « ParametresBD.java » afin qu'elle permette l'accès au SGBDR Oracle,
3) Créez dans Oracle, la BD et la table créées dans l'exercice 740. Remplissez à la main cette table,
4) Votre code Java lit cette table et affiche les tennismen sur la console.

Exercice N° 00770 : Importer Excel dans une BD

Créez deux Fichiers Excel ainsi :

- 1) Fichier « **Clients.csv** » (classeur à enregistrer sous forme csv) :

Java : Partie 4

Persistence de Données : Exercices

	J21					
	A	B	C	D	E	F
1	numclient	nom	adresse	codePostal	ville	
2		1 DUPONT	André	75003	Paris	
3		2 DURAND	Laure	13000	Marseille	
4		3 AUROUX	Karine	94000	Créteil	
5						
6						
7						

2) Fichier « **Commandes.csv** » :

	A	B	C	D	
1	numclient	numeroCde	montant	date	
2	1	1000	1200	12/12/2012	
3	2	2000	7300	10/02/2013	
4	3	3000	1785	17/06/2001	
5	4	4000	999	01/01/2003	
6					

- 3) Créez dans SQL Server la BD « **gestionCommerciale** » avec les tables « **clients** » et « **commandes** » ,
- 4) Lisez les fichiers Excel et remplissez les tables SQL :
 - a. La 1^{ère} ligne d'Excel ne doit pas être ajoutée à la table,
 - b. N'ajoutez une commande, dans la table SQL, que si elle correspond à un client existant.

Exercice N° 00780 : Paramètres pour accès Multi-Bases

Vous avez travaillé, en cours, avec la classe java « **ParametresBD.java** » qui permet de centraliser un code qui accéder à une BD,

Cette classe a un petit défaut : si je dois accéder à des BD de types différents (MySQL, SQL Server, DB2,) dans le même code : je ne peux pas le faire. Car cette classe ne permet d'accéder qu'un seul SGBDR à la fois.

Votre travail, si vous l'acceptez, est faire une nouvelle version nommée « **ParametresBDMulti.java** » afin que le type de SGBDR soit modifiable pour le code appelant avant l'appel des méthodes de cette classe.

Testez le bon fonctionnement de cette nouvelle classe avec MySQL et MS SQL Server 2012 ou plus