

Mission	N° 6J : Création des classes Java représentant la couche métier du projet
Connaissances requises	<div>➔ Avoir suivi un cours UML sur le diagramme de classes</div> <div>➔ Avoir suivi un cours sur l'architecture MVC</div> <div>➔ Avoir les connaissances Java en Programmation Orientée Objets</div>
Outils nécessaires	➔ Eclipse, WampServer, SQL Server Management Studio, Oracle 11g
Vous avez besoin	<div>➔ Du diagramme de classes métier fait dans la mission 3,</div> <div>➔ De la répartition des classes Java en architecture MVC, réalisée dans la mission 4</div> <div>➔ Ou du MLD Merise,</div> <div>➔ De la BD « BDFrais » déjà installée sur les types SGBDR cités ci-dessus</div>
Recommandations	

Mission 6J – Création de la

Couche Métier en java : Bibliothèque de classes

(Archive Java)

Présentation :

Vous allez ici créer une bibliothèque de classes Java (une archive Java) qui contiendra uniquement les classes métiers de la couche « **modèles** »,

Cette bibliothèque sera utilisée par les autres projets : gestion graphique des données, gestion Web des frais,etc.

Selon les modules que vous avez suivis, vous partez soit du diagramme de classes décomposé en MVC, soit du Modèle Logique de données fait dans Merise.

Travail à faire :

- 1) Assurez-vous, que vous possédez la BD «**BDFrais**» sous MySQL, SQL Server et Oracle :
Si ce n'est pas le cas : voir avec votre formateur la mission n° 10,
- 2) Reprendre le diagramme de classes décomposé en 3 couches selon l'architecture MVC,
- 3) Créez un projet Java nommé «**GSBLibrary**»,
- 4) Créez un package nommé «**com.gsb.modeles**»,
- 5) Créez dans ce package la classe «**EtatSQL**» qui servira à renseigner sur l'état d'exécution des instructions SQL :

Concepteur Développeur Informatique

Projet GSB : Mission 6J : Création de la couche métier en Java

Classe	Propriété	Type	Méthode
EtatSQL	CodeEtat	String	
EtatSQL	TableSQL	String	
EtatSQL	LibelleEtat	String	
			Constructeur
			getteurs

- 6) Créez dans ce package les classes :
- a. Qui correspondant à la couche « modèles » du **MVC** ou qui correspondent aux entités du MLD de Merise (si nous n’avez suivi le cours d’UML) : il s’agit des classes :

Classe	Table SQL correspondante
Utilisateur	utilisateurs
Region	regions
Departement	departements
Secteur	secteurs
Responsable	responsables
Delegue	delegates
Visiteur	visiteurs
Fraisforfait	fraisforfaits
EtatFiche	etatsFiches
Fichesfrais	fichesfrais
EtatFrais	etatsFrais
Lignefraisforfait	lignesfraisforfaits
LigneFraisHorsForfait	lignesfraishorsforfait

- 1) **Complétez chaque classe ci-dessus (sauf la classe Utilisateur) par :**
- a. Un constructeur recevant toutes les propriétés,
 - b. Les setteurs et les getteurs,
 - c. Les méthodes métiers vides suivants :

Méthode	Statique	réponse	Paramètres
creer	Non	EtatSQL	Non
modifier	Non	EtatSQL	Non
supprimer	Non	EtatSQL	Non
lire	Non	Objet du type de la classe	Non
Ist....	Oui	Collection objets de Type Vector<nom classe>	Non

- 2) Créez dans ce package, une classe « **Test** » avec la méthode « **main** »,
- 3) Ajoutez à votre package, la classe «**ParametresBD**» fournie par le formateur,

Concepteur Développeur Informatique

Projet GSB : Mission 6J : Création de la couche métier en Java

- 4) Vous allez maintenant coder chaque méthode et la tester dans la méthode main :
- a. Chaque méthode communique avec la BD « **BDFraîs** » pour insérer, modifier, supprimer, lire les données de tables SQL,

b. Rôle de chaque méthode :

Méthode	Rôle
creer	Transforme l'objet courant en ligne nouvelle dans la table SQL, La méthode crée un objet de type EtatSQL, le renseigne et le renvoie au code appelant : CodeEtat=000 si l'instruction SQL s'est bien terminée sinon CodeEtat= -100
modifier	Modifie une ligne dans la table SQL en fonction de l'objet courant, La méthode crée un objet de type EtatSQL, le renseigne et le renvoie au code appelant : CodeEtat=000 si l'instruction SQL s'est bien terminée sinon CodeEtat= -100
supprimer	Supprime une ligne dans la table SQL en fonction de l'objet courant, La méthode crée un objet de type EtatSQL, le renseigne et le renvoie au code appelant : CodeEtat=000 si l'instruction SQL s'est bien terminée sinon CodeEtat= -100
lire	Lit une ligne SQL dans la table SQL avec le paramètre reçu et renvoie un objet de type de la classe
Ist....	Renvoie toutes les lignes de la table SQL, sous forme d'objets de type de la classe, regroupés dans une collection de type Vector.

- ATTENTION :
- o Dans un premier temps : codes les 5 méthodes de la classe « **Region** » pour qu'elles communiquent réellement avec la BD,

o Dans les autres classes, les méthodes répondront pour l'instant avec un message sur la console « Je suis la méthode De la classe »,

o Dans les missions à venir, vous reviendrez codes les méthodes au fur et à mesure des besoins.

- 5) Créez et testez la classe «**Utilisateur**» avec :
- a. Un constructeur,

b. Des getteurs et pas de setteurs,

c. Méthode « **verifier** » qui utilise l'objet courant et vérifie dans la BD si le login et le mot de passe existe dans la table utilisateurs. Elle renvoie un booléen.
- 6) Une fois que toutes les méthodes de toutes les classes sont codées et testées :
- a. **Supprimez** la classe «**Test**»,

b. Générez une **bibliothèque de classes** (une archive Java) à partir de ce projet
- 7) Testez l'utilisation de cette bibliothèque :
- a. Créez un projet java nommé «**GSBLibraryTest**»,

b. Référez dans ce projet, la bibliothèque créée ci-dessus,

c. Essayez de créer des objets à partir des classes de cette bibliothèque,
- Mission terminée.
- Auteur : Chouaïb LAGHLAM (chouaib.laghlam@laposte.net)

Page 3