D01A\_Démo – Modificateurs d’accès

Table des matières

[1 Général 2](#_Toc137893651)

[2 Mise en situation 3](#_Toc137893652)

[2.1 Requis du projet 3](#_Toc137893653)

[2.1.1 Les données requises pour l’application sont les suivantes : 3](#_Toc137893654)

[2.1.2 L’application doit pouvoir : 3](#_Toc137893655)

[2.1.3 Exigence de sécurité : 3](#_Toc137893656)

[2.1.4 Informations additionnelles 4](#_Toc137893657)

[3 Choix des packages et classes 5](#_Toc137893658)

[4 Code – première partie 7](#_Toc137893659)

[4.1 Projet IntelliJ 7](#_Toc137893660)

[4.2 package ca.codexcvm.compagnie 7](#_Toc137893661)

[4.3 package ca.codexcvm.livres 8](#_Toc137893662)

[4.4 Package ca.codexcvm.ventes 9](#_Toc137893663)

# Général

On explore les modificateurs d’accès à partir d’une mise en situation.

Les étudiants codent avec l’enseignant.

|  |
| --- |
| **Le code résultant du démo est la base du laboratoire qui suit. Les étudiants DOIVENT écrire le code à mesure de la démonstration pour faire ce prochain laboratoire.** |

# Mise en situation

On désire modéliser une application de vente pour la librairie **Codex CVM**.

L’entreprise supporte plusieurs succursales et vend des livres.

|  |
| --- |
| NOTE : pour cet exercice, on utilise **protected** comme **default** : accessible à l’intérieur du même package. |

## Requis du projet

### Les données requises pour l’application sont les suivantes :

* Nom de l’entreprise: Codex CVM
* Une marge de profit globale à l’entreprise, à appliquer dans le calcul du prix des livres = 10%. Le prix coutant du livre + cette marge de profit est le prix payé par une succursale.
* Des Succursales
  + No de la succursale
  + Nom de la succursale
  + Une marge de profit à appliquer au calcul de prix des livres = 25%, appliqué en supplément de la marge de profit de compagnie
  + Des livres
    - Titre du livre
    - Prix coutant du livre (le prix payé par la compagnie)

### L’application doit pouvoir :

Calculer le prix vendant d’un livre à partir du prix coutant :

Prix coutant + marge de profit de compagnie = prix succursale

Prix succursale + marge de profit de succursale = prix vendant

Afficher une facture (1 seul livre) avec le format suivant :

|  |
| --- |
| Nom de la compagnie  Numéro de la Succursale  Nom de la succursale  Titre du livre  Prix du livre (prix vendant) |

### Exigence de sécurité :

Les marges de profit sont des informations confidentielles, on veut s’assurer qu’aucune partie du code pour les livres et les ventes, ne peut accéder à ces informations. Seul le prix vendant devrait être accessible.

### Informations additionnelles

Pour la compagnie on désire regrouper dans une classe séparés tous les utilitaires de tarification (calcul du prix d’un livre vendu à une succursale, rabais de volume, points pour un programme de fidélité etc.)

Pour les succursales on désire regrouper dans une classe séparée tous les utilitaires de tarification (calcul du prix vendant d’un livre, promotions, escomptes etc.)

On construira éventuellement :

* Un module logiciel complet pour la gestion de la compagnie
* un module logiciel complet pour la gestion de l’inventaire des livres.
* un module logiciel complet pour la gestion des succursales.
* un module logiciel complet la gestion des ventes (application de caisses, ventes en ligne etc).

# Choix des packages et classes

À partir des requis, en 5 minutes, chaque étudiant remplis un tableau décrivant les packages et classes qu’il choisirait pour son projet.

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Packages |  |  |  |  |  |
| Classes |  |  |  |  |  |

On discute des résultats, avec quelques exemples de choix des étudiants, plusieurs solutions sont possibles.

Ensuite, on choisit la structure ci-dessous :

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Packages | ca.codexcvm.compagnie | ca.codexcvm.livres | ca.codexcvm.succursale | ca.codexcvm.ventes |
| Classes | Compagnie  TarificationCompagnie | Livre | Succursale  TarificationSuccursale | ApplicationVentes |

Choix des Packages

* 4 Package, 1 par module logiciel principal. Regrouper les classes dans des packages par fonctionnalité est une des raisons principales de faire des packages.

Choix des classes

* **Compagnie**, **Livre**, et **Succursale** parce que ce sont des entités bien définies dans l’énoncé.
* **TarificationCompagnie** et **TarificationSuccursale** parce que c’est spécifié par par l’énoncé.
* **ApplicationVentes** dans laquelle on aura la méthode Main. C’est la classe principale.

# Code – première partie

En démo, on modélise les classes requises pour afficher le prix de succursale (prix coutant d’un livre + marge de profit de compagnie)

La suite de l’application est faite dans le laboratoire qui suit.

## Projet IntelliJ

On créé un projet – C23\Projets\L01D-CodexCVM\

## package ca.codexcvm.compagnie

Créer le Package de la compagnie : ca.codexcvm.compagnie

Public Classe Compagnie

private String nomCompagnie = "Codex CVM";

private double margeProfit = 0.1;

public String getNomCompagnie() {  
 return nomCompagnie; }

protected double getMargeProfit() {  
 return margeProfit; }

Noter:

* Les champs sont **private**
* Le getter de nomCompagnie est **public**
* Le getter de margeProfit est **protected**, dispo juste dans le même package
* On a assigné des valeurs directement dans la déclaration des champs, un choix de programmeur
* Méthodes du mutation (setters) juste si nécessaire.

Public Classe TarificationCompagnie (même package)

public double calculerPrixSuccursale(double prixCoutant){  
 Compagnie compagnie = new Compagnie();  
 return prixCoutant \* ( 1 + compagnie.getMargeProfit());  
 }

Noter:

* Cette classe est identifiée comme une classe d’utilitaires de tarification au niveau de la compagnie
* Il n,y a pas de champs dans cette classe
* Pour le démo, on créé une seule méthode pour calculer le prix d’un livre à la succursale. Comme le prix coutant d’un livre ne sera pas disponible dans cette classe, il doit être passé en paramètre
* La méthode elle-même est publique

## package ca.codexcvm.livres

Créer le Package des livres : ca.codexcvm.livres

Public Classe Livre

private String titre;

private double prixCoutant;

public String getTitre() {  
 return titre; }

**Pas de getter pour prixCoutant**

public void setTitre(String titre) {  
 this.titre = titre; }

public void setPrixCoutant(double prixCoutant) {  
 this.prixCoutant = prixCoutant; }

**On fait une méthode pour retourner le prix succursale**

public double prixSuccursale(){  
 return calculerPrixSuccursale(this.prixCoutant);  
}

**Fonctionne pas : on a besoin d’un objet pour utiliser la méthode**

public double prixSuccursale(){  
 TarificationCompagnie tarificationCompagnie = new TarificationCompagnie();  
 return tarificationCompagnie.calculerPrixSuccursale(this.prixCoutant);  
}

**Fonctionne toujours pas : on a besoin d’un import de la classe pcq pas dans même package**

…tout de suite après le nom du package :

import ca.codexcvm.compagnie.TarificationCompagnie;

**Maintenant ça fonctionne**

Noter:

* Les champs sont **private**
* Le getter de titre est **public**
* Pas de getter pour prixCoutant et puisque private n’est pas accessible ailleurs que dans cette classe, ce champs n’est exposé dans aucune autre classe
* La méthode prixSuccursale() est publique, et n’expose ni le prix coûtant, ni la marge de profit de compagnie aux autres classes

## Package ca.codexcvm.ventes

**On créé un package et une classe pour le programme principal, on veut afficher le nom de la compagnie, un titre de livre et son prix de succursale.**

Créer le Package ca.codexcvm.ventes

On importe les classes Compagnie et Livre

import ca.codexcvm.compagnie.Compagnie;  
import ca.codexcvm.livres.Livre;

Public Classe ApplicationVentes

public static void main(String[] args) {  
  
 **On créé un objet Compagnie et un objet livre**

Compagnie compagnie = new Compagnie();  
 Livre livre = new Livre();

Essayer d’accéder aux champs de compagnie :

System.*out*.println(compagnie.nomCompagnie);

**Ne fonctionne pas**, besoin des getters

System.*out*.println(compagnie.getNomCompagnie());

Essayer de configurer un titre au livre

livre.titre = "Matrix";

**Ne fonctionne pas**, besoin des setters

livre.setTitre("Matrix");

Afficher le titre du livre (on utilise le getter du titre qui est public)

System.*out*.println(livre.getTitre());

Essayer d’accéder à la méthode compagnie.getMargeProfit

System.*out*.println(compagnie.getMargeProfit());

**Ne fonctionne pas**, pas même package et protected

Utiliser plutôt prixSuccursale de Livre (qui est public)

System.*out*.println(livre.prixSuccursale());

Exécuter

Sauvegarder

Passer au laboratoire, à partir de ce début de code.