

TP5 – Architecture en deux couches d'accès aux données

L'objectif de ce TP est d'implémenter une application console basique qui permet de gérer des articles stockés dans une base de données. L'application doit être organisée en deux couches :

- Une couche d'accès aux données **Data Access Layer (DAL)** qui contient toutes les opérations relatives à l'interrogation de la base de données
- Une couche **Présentation** qui contient toutes les opérations relatives à l'interaction avec l'utilisateur en terme de saisie et d'affichage des données.

La base de données

Les articles sont stockés dans une base appelée *Stock* contenant une table **Article** ayant les champs suivants : *Reference (Entier)*, *Designation (Chaine)*, *Prix (Réel)*. Le champ *Reference* joue le rôle de clé primaire.

Implémentation de l'entité Article

Chaque enregistrement de la table **Article** sera représenté par programmation par une classe **Article** ayant les mêmes champs que la table. Donner la définition de cette classe sachant qu'en plus des trois attributs, cette dernière dispose de deux constructeurs, un par défaut et l'autre paramétré.

Implémentation de la couche d'accès aux données

La couche d'accès aux données est représentée par la classe **DALArticle**. La spécification de cette classe est donnée ci-dessous. Compléter les définitions des méthodes de cette classe.

```
class DALArticle
{ // Ajouter les données d'un objet Article passé comme argument
  public static void Add(Article a) { ... }
  // Supprimer un article donné par sa référence
  public static void Delete(int Ref) { ... }
  // Modifier les données d'un article donné par sa référence par les données d'un objet Article
  public static void Update(int Ref, Article a) { ... }
  // Sélectionner un article donné par sa référence
  public static Article SelectByRef(int Ref) { ... }
  // Sélectionner tous les articles
  public static List<Article> SelectAll() { ... }
}
```

Implémentation de la couche Présentation

La couche Présentation est représentée par la classe **ViewArticle**. La spécification partielle de cette classe est donnée ci-dessous. Compléter les définitions des méthodes de cette classe.

```
class ViewArticle
{
  // Affichage des informations concernant un article
  public static void Show(Article a) {...}
  // Affichage des informations concernant une liste d'articles
  public static void Show(List<Article> LesArticles) {...}
  // Saisie des données d'un article à partir de l'utilisateur
  public static Article GetArticleFromUser() {...}
  // Saisie des données d'un article et son ajout à la base de données
  public static void CommandeAjouter() {...}
  // Saisie de la ref de l'article à supprimer puis sa suppression de la base
  public static void CommandeSupprimer() {...}
  // Saisie de la ref de l'article à modifier puis saisie des nouvelles données suivies de la mise
  // à jour
  public static void CommandeModifier() {...}
  // Saisie de la ref de l'article à recherché puis affichage du résultat
  public static void CommandeChercherParReference() {...}
  // Affichage de la liste des articles de la base de données
  public static void CommandeAfficherTout() {...}
  // Un menu appelant les commandes ci-dessus
  public static void Menu() {...}
}
```

Pour information le menu affichera les informations de la manière suivante :

```
/ ***** /
```

Bienvenue à l'application de gestion des articles.

Tapez le numéro qui correspond à l'action que vous voulez faire :

- 1- Ajouter un article
- 2- Modifier un article
- 3- Supprimer un article
- 4- Chercher un article par sa référence
- 5- Lister tous les articles
- 6- Quitter

```
/ ***** /
```

Implémentation de la classe Program

La classe *Program* contient la fonction principale *main* permettant le lancement de l'application. Ce lancement se fera à travers l'appel du menu de la classe *ViewArticle*.