Question 1.1  
Modifier la structure de la base de données utilisée par l'application AchatWebCaisse afin de permettre la souscription en ligne du logiciel WebCaisse.

Question 1.2  
Expliquer en quoi la structure de la table ne permettra pas de gérer l'historique des formules souscrites, indispensable à la détermination du montant mensuel à régler par le client.

Question 1.3  
Proposer une correction de la structure de la table qui réponde au besoin exprimé.

Question 2.1  
Écrire les requêtes permettant d’extraire les informations nécessaires de la base de données fournie  
dans le dossier documentaire.

Question 2.2  
Modifier la requête SQL de la méthode listeConsoAFideliser(int seuilVentes, String dateDeb, String dateFin) fournie par Sylvain Cho, afin de lister les consommateurs qui n'ont pas adhéré au programme de fidélisation et pour lesquels on a enregistré un nombre de ventes supérieur au seuil donné, durant la période donnée (le seuil et la période sont fournis en paramètre).

Question 2.3  
Compléter la méthode testInitConso permettant de combler ce manque.

Question 2.4  
Compléter la méthode testAddMontant permettant de valider les points de fidélité obtenus dans le cas d'un de programme de fidélisation par points.

Question 3.1  
Rédiger le commentaire de la méthode statVente de la classe Statistique expliquant ce qu'elle retourne.

Question 3.2  
Écrire la méthode getNbVentes de la classe Conso qui retourne le nombre de ventes enregistrées dans la collection des ventes du consommateur.

Question 3.3  
Écrire le constructeur de la classe VenteEcommerce qui permet d’initialiser tous les attributs d'une instance de la classe.

Question 3.4  
Compléter le code de la méthode compareLieuVente.

Question 3.5  
Expliquer en quoi la dernière instruction "return totalMag/totalEcom" de la méthode compareLieuVente peut poser problème.

Question 3.6  
Écrire la méthode répondant à ce besoin.