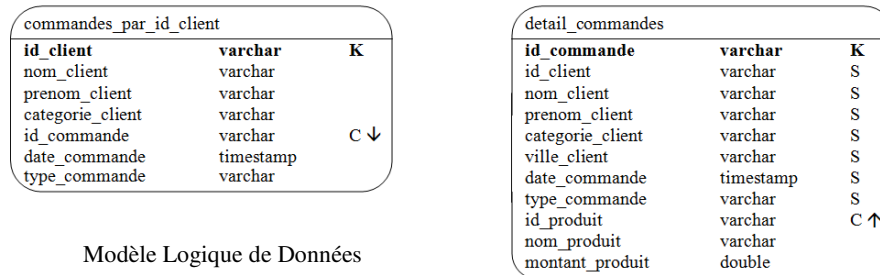


## ⑥ AB-Orders - NoSQL : Cassandra

On s'intéresse à l'entreprise AB-Orders™ qui a mis en place une Base de Données Cassandra afin de gérer les commandes de ses clients.

Le Modèle Logique de Données de la BD Cassandra vous est communiqué ci-dessous :



- Avec la structure des tables du Modèle Logique de Données indiqué, si on ne rajoute pas d'index secondaire ou de vue matérialisée, et qu'on n'utilise pas non plus la clause `ALLOW FILTERING` dans les requêtes, pensez-vous qu'il est possible de réaliser les requêtes suivantes en CQL (on vous demande de répondre aux questions sur votre feuille sans pour le moment écrire le code des requêtes) ?
  - numéro des commandes passées par le client 'C1'. oui ☐ non ☐
  - numéro des commandes passées par le client 'Delune'. oui ☐ non ☐
  - date de la commande '001'. oui ☐ non ☐
  - numéro de tous les clients (sans doublon). oui ☐ non ☐
  - numéro et le nom de tous les clients (sans doublon). oui ☐ non ☐
  - numéro, date et type des commandes passées par le client 'C1', classées par numéro de commande. oui ☐ non ☐
  - numéro, date et type des commandes passées par le client 'C1', classées de la plus vieille à la plus récente. oui ☐ non ☐
  - le nombre de commandes du client 'C1'. oui ☐ non ☐
  - le nom des produits de la commande '001'. oui ☐ non ☐
  - le nom des produits commandés par le client 'C1'. oui ☐ non ☐
  - l'identifiant des commandes qui contiennent le produit 'P1'. oui ☐ non ☐
  - le montant total de la commande '001'. oui ☐ non ☐
- Attendez que votre binôme favori termine la question 1. Une fois que cela est fait, comparez vos réponses avec les siennes. Si vous n'avez pas répondu la même chose à certaines requêtes, essayez de convaincre votre binôme que c'est vous qui avez raison.
- Exécutez sur Cassandra les requêtes de la question 1 afin de voir si vos hypothèses sont exactes. Pour cela :
  - Lancez le client DevCenter
  - Créez une connexion afin de vous connecter au SGBD Cassandra de l'IUT. Pour cela, dans le menu en haut à gauche, cliquez sur l'icône "create new connection (Ctrl+Alt+Shift+N)". Puis, indiquez 162.38.222.152 comme "Contact hosts" et 9034 pour "Native Protocol port". Puis dans l'écran suivant, cochez la case "This cluster requires credentials" et indiquez votre "Login" (nom + première lettre du prénom) ainsi que votre "Password" (num étudiant ou sinon, date de la rentrée).
  - Si cela n'est pas déjà fait, créez un nouveau Script CQL dans la fenêtre du milieu "create a new CQL Script (Ctrl+Shift+N)" puis dans cette fenêtre indiquez que l'on utilise le keyspace qui a pour nom votre nom de login en sélectionnant dans le menu déroulant "in keyspace" le keyspace qui convient.
  - Copiez dans la fenêtre du milieu le code du fichier AB-Orders.cql qui se trouve sur le Moodle. Puis exécutez ce code en appuyant sur le bouton vert ("Execute CQL Script (Alt+F11)"). Cela créera les tables avec des données à l'intérieur.
  - Réalisez les requêtes de la question 1 afin de vérifier vos hypothèses.

commandes_par_id_client		
id_client	varchar	K
nom_client	varchar	
prenom_client	varchar	
categorie_client	varchar	
id_commande	varchar	C ↓
date_commande	timestamp	
type_commande	varchar	

Modèle Logique de Données

detail_commandes		
id_commande	varchar	K
id_client	varchar	S
nom_client	varchar	S
prenom_client	varchar	S
categorie_client	varchar	S
ville_client	varchar	S
date_commande	timestamp	S
type_commande	varchar	S
id_produit	varchar	C ↑
nom_produit	varchar	
montant_produit	double	

4. Ecrire une requête qui interroge la table `commandes_par_id_client` afin de trouver le numéro des commandes passées par les clients de la catégorie 'Client patibulaire'. Cela ne devrait pas marcher car `categorie_client` n'est pas la clé de partition.

- Réalisez cette requête en créant un index secondaire dans la table `commandes_par_id_client`.
- Toutefois, les index secondaires sont déconseillés car ils dégradent les performances notamment lorsque les données sont stockées sur plusieurs nœuds (de la même façon qu'il faut éviter d'utiliser l'instruction `ALLOW FILTERING` pour réaliser une requête). Afin d'améliorer les performances, on va donc plutôt faire la requête grâce à une vue matérialisée. Comme le code d'une vue matérialisée peut être assez lourd à écrire, on va générer cette vue avec l'interface graphique de DevCenter. Pour cela :
  - Dans la fenêtre en haut à droite, cliquez avec le bouton droit sur *Materialized View* de votre keyspace, puis dans le menu contextuel cliquez sur *New Materialized View*.
  - Sélectionnez ensuite les colonnes qui doivent apparaître dans la vue matérialisée
  - Dans la fenêtre suivante indiquez la clé de votre vue matérialisée. On placera dans le fenêtre du haut la ou les colonnes qui seront clé de partition. Et dans la fenêtre du bas, la ou les colonnes qui seront colonnes de clustering. On rappelle que la vue matérialisée doit avoir dans sa clé (clé de partition ou colonne de clustering) toutes les colonnes qui étaient dans la clé de la table d'origine, même si elles ne sont pas forcément utiles pour la requête qu'on veut réaliser.
  - Faire une requête qui utilise la vue matérialisée (dans son `FROM`) et vérifiez qu'il y a bien deux commandes dans le résultat.
- Une fois la vue matérialisée créée, vérifiez qu'elle est bien mise à jour lorsqu'on modifie les données de la table `commandes_par_id_client` ; par exemple, il ne doit plus y avoir qu'une seule commande dans la vue matérialisée après avoir lancé la requête suivante :
 

```
UPDATE commandes_par_id_client
SET categorie_client = 'XXX'
WHERE id_client = 'C7'
AND id_commande = '013';
```

5. Pour terminer, écrire la requête suivante sous DevCenter.

*Le numéro, la date et le type de chaque commande avec le nom, le prénom, la catégorie et la ville du client associé (les commandes ne doivent apparaître qu'une seule fois).*