

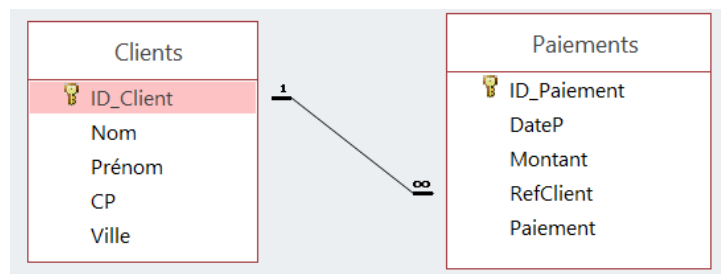
Bases de Données Relationnelles

TP3 : Mise à jour d'une BD (LMD)

Objectif : Mettre à jour une BDR avec des requêtes SQL LMD.

Télécharger la base « *HistoriqueClient* » disponible sur « cours en ligne ». L'objectif de ce TP est de créer des requêtes de mise à jour des données dans le langage SQL LMD.

Ci-dessous le schéma logique de la base « *HistoriqueClient* ». Les tables « *Clients* » et « *Paielements* » représentent respectivement la liste des clients d'une entreprise ainsi que l'historique des factures passées aux clients sur plusieurs années. Pour chaque commande l'entreprise enregistre si son client a payé en totalité sa facture ou non.



Question 2 : Mise à jour (SQL LMD)

Faites les mises à jour suivantes dans la base de données :

1. Créer une requête qui permet de modifier la ville et le CP du client LAPORTE qui a déménagé à Grenoble (CP 38000)
2. Créer la requête qui permet d'avancer d'un mois la date de la facture du client N°10 enregistrée le 09/03/2014 (voir annexe pour la mise à jour des dates).
3. Mr Christian POUILLON vient de régler toutes ses factures impayées jusqu'à 2010. Créer la requête LMD permettant de mettre à jour la base.
4. La facture de Mme Carine TARISSE du 24/07/2016 est annulée. Créer la requête d'action permettant de supprimer cette facture (utiliser une sous-requête pour chercher le numéro de la cliente TARISSE).

Exercice 2 : Archivage

Le gérant souhaite archiver les vieilles factures qui datent d'avant 2005, quelle soient payées ou non. On vous demande alors de faire le nécessaire pour mettre en place cet archivage, sachant que l'année de l'archivage (année minimale des factures à garder dans la table « Paiements ») doit être spécifiée interactivement.

1. Commencer d'abord par créer une table « ArchiveFactures » ayant le même schéma (mêmes champs avec les mêmes types) que la table « Paiements ».
2. Créer ensuite une requête d'insertion qui insère dans la table « ArchiveFactures » les lignes dans la table « Paiement » dont l'année est inférieure à 2005.
3. Créer une requête de suppression qui supprime de la table « Paiements » les lignes qui ont été ajoutées dans la table « ArchiveFactures ». Utiliser une sous-requête pour ne supprimer que les lignes qui ont été insérées dans l'archive.

Annexe :

Fonction	Description	Exemple
DATE_ADD(date, INTERVAL <u>expr</u> unit)	Décale un horaire ou une date	DATE_ADD('2018-05-01',INTERVAL 1 DAY); -> '2018-05-02'
DATEDIFF(type, champ1, champ2)*	Calcule l'intervalle de temps entre deux dates ou deux horaires	DATEDIFF('2007-12-31 23:59:59','2007-12-30'); -> 1 DATEDIFF('2010-11-30 23:59:59','2010-12-31'); -> -31