

Matière : SQL-PL/SQL

Enseignant/e(s) : .....

Nom : .....  
 Prénom : .....  
 Groupe : .....  
 Site : .....

NOTE :

Veuillez répondre sur la feuille de sujet fournie.  
 Veuillez mentionner votre nom et prénom.

L'usage des téléphones portables et de tout objet connecté (montre intelligente, etc.) **est strictement interdit.**  
**SEULE UNE FEUILLE A4 MANUSCRITE EST AUTORISEE. PAS DE PHOTOCOPIE NI D'IMPRIME**

### ENONCE

On considère le schéma suivant de la base de données : PaiementsFactures pour gérer les paiements des factures de commandes.

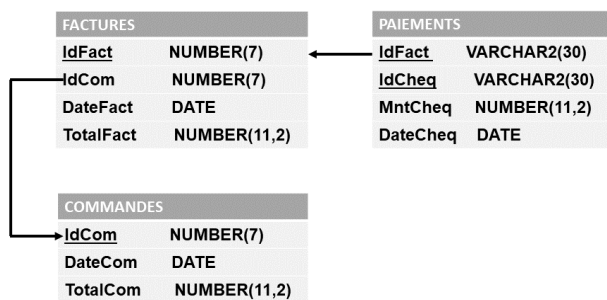


Fig. 1 : schéma logique de la BD

Les clés primaires sont soulignées et les flèches indiquent une contrainte d'intégrité référentielle. La sémantique de chaque table et attribut est donnée dans le tableau suivant :

TABLE	ATTRIBUT	CONTRAINTE	SEMANTIQUE
COMMANDES	La table COMMANDES contient toutes les commandes à encaisser		
	<u>IdCom</u>	PRIMARY KEY	Identifiant de la commande
	DateCom	NOT NULL	La date de la commande
	TotalCom	NOT NULL	Le montant total de la commande. Il est >0
FACTURES	La table FACTURES contient tous les factures émises pour des commandes. A une commande correspond une seule facture.		
	<u>IdFact</u>	PRIMARY KEY	Identifiant de la facture
	IdCom	NOT NULL	L'identifiant de la commande à laquelle correspond la facture. Il est obligatoire.
	DateFact	NOT NULL	La date de l'émission de la facture
PAIEMENTS	TotalFact	NOT NULL	Le montant de la facture. Il doit être le même que le total de la commande
	Contient tous les paiements des factures. Le paiement se fait uniquement par chèque. Et une facture peut être payée en plusieurs fois d'où la clé primaire de cette table qui est ( <u>IdFact, IdCheq</u> )		
	<u>IdFact</u>		L'ID de la facture concerné par le chèque IdCheq
	<u>IdCheq</u>		L'ID du chèque émis pour le paiement d'une facture en totalité ou partiellement
	MntCheq	NOT NULL	Le montant du chèque. Il doit être > 1000 DH
	DateCheq	NOT NULL	La date de l'émission du chèque

Fig. 2 : dictionnaire de données

Avec les règles de gestion suivantes :

**Règle 1** : à une commande, correspond une et une seule facture

**Règle 2** : à une facture peut correspondre plusieurs paiements par chèque

**Règle 3** : le montant d'une facture doit être le même que le total d'une commande

**PARTIE 1 : LANGAGE DE DEFINITION DE DONNEES**

**1.1.** On suppose que les tables COMMANDES et FACTURES sont déjà créées. Créez la table PAIEMENTS sans oublier les contraintes. **(2 points)**

.....

.....

.....

.....

.....

.....

.....

.....

.....

.....

**PARTIE 2 : REQUETES SQL**

**2.1.** Ecrivez une requête SQL qui retourne la liste des paiements de la facture 111. Le schéma de la réponse est (IdCheq, MntCheq, DateCheq) **(1 point)**.

.....

.....

.....

.....

.....

.....

**2.2.** Ecrivez une requête SQL qui retourne la liste des paiements de la commande 222. Le schéma de la réponse est (IdCheq, MntCheq, DateCheq) **(2 point)**.

.....

.....

.....

.....

.....

.....

**2.3.** Donnez une requête SQL qui retourne le nombre de chèques pour le paiement de la commande 222, le total payé et le reste à payer.

Schéma de la réponse (NbrCheq, TotalPaye, ResteAPayer). **(2 points)**

.....

.....

.....

.....

.....

.....





**4.1.** Le total d'une facture doit être le même que celui de la commande qui correspond à la facture (Règle de gestion 3).  
Ecrivez un trigger qui lors de l'ajout d'une facture, dans la table FACTURES, vérifie que le montant de la facture est celui de la commande, sinon, il met le total de la commande comme le total de la facture correspondant (**3 points**)

[illegible]