Exo 2

1- Obtenir l’utilisateur ayant le prénom “Muriel” et le mot de passe “test11”, sachant que l’encodage du mot de passe est effectué avec l’algorithme Sha1.

**select** \*

**from** client

**where** prenom = 'Muriel' **and** password = SHA1('test11');

2- Obtenir la liste de tous les produits qui sont présent sur plusieurs commandes.

**select** nom, **s**(\*)

**from** commande\_ligne

**group** **by** nom

**having** **count**(nom) > 1

**order** **by** **count**(nom) **desc**;

3- Obtenir la liste de tous les produits qui sont présent sur plusieurs commandes et y ajouter une colonne qui liste les identifiants des commandes associées.

**select** nom, **group\_concat**(commande\_id), **count**(nom)

**from** commande\_ligne

**group** **by** nom

**having** **count**(nom) > 1

**order** **by** **count**(nom) **desc**;

4- Enregistrer le prix total à l’intérieur de chaque ligne des commandes, en fonction du prix unitaire et de la quantité

**update** commande\_ligne **set** prix\_total=quantite\*prix\_unitaire

5- Obtenir le montant total pour chaque commande et y voir facilement la date associée à cette commande ainsi que le prénom et nom du client associé

(difficulté très haute) Enregistrer le montant total de chaque commande dans le champ intitulé “cache\_prix\_total”

**select** commande.id, commande.cache\_prix\_total **as** total, commande.date\_achat, client.prenom, client.nom

**from** commande

**join** commande\_ligne **on** commande.id=commande\_ligne.commande\_id

**join** client **on** commande.client\_id=client.id

**group** **by** commande.id;

**update** commande

**join**

(**select** commande\_id, **sum**(prix\_total) **as** total **from** commande\_ligne

**group** **by** commande\_id ) com\_l

**on** commande.id = com\_l.commande\_id

**set** cache\_prix\_total = com\_l.total;

6- Obtenir le montant global de toutes les commandes, pour chaque mois

**select** date\_achat, cache\_prix\_total **as** total

**from** commande

**group** **by** **month**(date\_achat);

7- Obtenir la liste des 10 clients qui ont effectué le plus grand montant de commandes, et obtenir ce montant total pour chaque client.

**select** client.prenom, client.nom, cache\_prix\_total **as** total

**from** commande

**join** client **on** commande.client\_id = client.id

**order** **by** total **desc**

**limit** 10;

8- Obtenir le montant total des commandes pour chaque date

**select** date\_achat, **sum**(cache\_prix\_total) **as** total

**from** commande

**group** **by** date\_achat;

9- Ajouter une colonne intitulée “category” à la table contenant les commandes. Cette colonne contiendra une valeur numérique

**alter** **table** commande

**add** “category” **float**;

10 - Enregistrer la valeur de la catégorie, en suivant les règles suivantes :

“1” pour les commandes de moins de 200€

“2” pour les commandes entre 200€ et 500€

“3” pour les commandes entre 500€ et 1.000€

“4” pour les commandes supérieures à 1.000€

**update** commande

**set** “category” = (**case**

**when** cache\_prix\_total < 200 **then** 1

**when** cache\_prix\_total < 500 **then** 2

**when** cache\_prix\_total < 1000 **then** 3

**else** 4

**end** );

10- Créer une table intitulée “commande\_category” qui contiendra le descriptif de ces catégories

**create** **table** command\_category(

category **int** **primary** **key** **auto\_increment**,

description **varchar**(200) **not** **null**);

11- Insérer les 4 descriptifs de chaque catégorie au sein de la table précédemment créée

**insert** **into** command\_category(category, description)

**values** (1, 'prix\_total < 200'),

(2, 'prix\_total 200-500'),

(3, 'prix\_total 500-1000'),

(4, 'prix\_total > 1000');

12- Supprimer toutes les commandes (et les lignes des commandes) inférieur au 1er février 2019. Cela doit être effectué en 2 requêtes maximum

**delete** commande, commande\_ligne **from** commande **inner** **join** commande\_ligne

**where** commande.id=commande\_ligne.commande\_id **and** date\_achat<'2019-01-02'

;