BTS SIO

Les Données : Interrogation de la base de données ➔ Le langage S.Q.L

```
SELECT ETUDIANT.Nom_studiant, ETUDIANT.Prenom_studiant

FROM ETUDIANT

WHERE ETUDIANT.motiv = « ok »;
```

Bloc 1 – BDD2

BTS SIO

Sommaire

1. Les différents opérateurs de l'algèbre relationnelle. 3
   1.1. Opérateurs relationnels. 3
      1.1.1. La Projection. 3
      1.1.2. La sélection. 3
      1.1.3. La Jointure. 4
   1.2. Opérateurs ensemblistes 4
2. Le processus de construction d'une requête d'interrogation. 5
3. Les outils MySQL 6
4. Le S.Q.L (StructuredQueryLanguage). 7
   4.1. Comment questionner la base de données ? 7
   4.2. Structures de base d'une requête d'interrogation 8
      4.2.1. Opérateurs relationnels. 8
      4.2.2. Les opérateurs ensemblistes. 9
   4.3. Les spécificités de SQL. 11
      4.3.1. Spécificités de certains prédicats SQL. 11
      4.3.2. Le tri : ORDER BY 12
      4.3.3. "Like" ou "Not Like". 12
      4.3.4. Le "isNull". 13
      4.3.5. "Distinct" 13
      4.3.6. Les regroupements et les fonctions associées. 13
      4.3.7. Les Sous-requêtes. 16
   4.4. Les Vues 19
      4.4.1. Création d'une vue. 19
      4.4.2. Suppression d'une vue. 19
5. Applications 20

1. Les différents opérateurs de l'algèbre relationnelle.

L'algèbre relationnelle est un concept mathématique de la théorie des ensembles qui consiste à définir des opérations sur des tables (relations).

1.1. Opérateurs relationnels.

1.1.1. La Projection.

C'est une opération qui consiste à afficher un certain nombre de colonnes.

NUMECLIE	NOMCLIE	ADRECLIE	CODPCLIE	VILLCLIE	TELECLIE	NUMEREPR
100001	Digit Equipement	25, rue des Pins	06200	Valbonne	0493564578	2
100002	F.D.T	10, Impasse de la Poste	06000	Nice	0493564545	2
100010	Ecole Pigier	10, av J Médecin	06200	Nice	0493656562	2

NUMECLIE	NOMCLIE	VILLCLIE
100001	Digit Equipement	Valbonne
100002	F.D.T	Nice
100010	Ecole Pigier	Nice

1.1.2. La sélection.

C'est une opération qui permet d'obtenir toutes les occurrences répondant à un ou plusieurs critères..

NUMECLIE	NOMCLIE	ADRECLIE	CODPCLIE	VILLCLIE	TELECLIE	NUMEREPR
100001	Digit Equipement	25, rue des Pins	06200	Valbonne	0493564578	2
100002	F.D.T	10, Impasse de la Poste	06000	Nice	0493564545	2
100010	Ecole Pigier	10, av J Médecin	06200	Nice	0493656562	2
100011	Forum Immobilier	5, Av de Grasse	06500	Valbonne	0493252525	2
100020	Habitable	2, av de Valbonne	06200	Nice	0493565656	2
200005	Lycée Victor Hugo	Bd Desplaces	13001	Marseille	0491525269	3
300010	A.R.T	20, rue de la mer	83000	Toulon	0494562154	1

NUMECLIE	NOMCLIE	ADRECLIE	CODPCLIE	VILLCLIE	TELECLIE	NUMEREPR
100002	F.D.T	10, Impasse de la Poste	06000	Nice	0493564545	2
100010	Ecole Pigier	10, av J Médecin	06200	Nice	0493656562	2
100020	Habitable	2, av de Valbonne	06200	Nice	0493565656	2

1.1.3. La Jointure.

C'est une <u>opération permettant de mettre en relation deux ou plusieurs tables</u> grâce à leurs colonnes communes correspondant respectivement aux clé primaire et clé étrangère)

Une jointure est nécessaire lorsque les données recherchées se trouvent dans plusieurs tables.

NUMECLIE	NOMCLIE	ADRECLIE	CODPCLIE	VILLCLIE	TELECLIE	NUMEREPR
100001	Digit Equipement	25, rue des Pins	06200	Valbonne	0493564578	2
100002	F.D.T	10, Impasse de la Poste	06000	Nice	0493564545	2
100010	Ecole Pigier	10, av J Médecin	06200	Nice	0493656562	2
100011	Forum Immobilier	5, Av de Grasse	06500	Valbonne	0493252525	2
100020	Habitable	2, av de Valbonne	06200	Nice	0493565656	2
200005	Lycée Victor Hugo	Bd Desplaces	13001	Marseille	0491525269	3
300010	A.R.T	20, rue de la mer	83000	Toulon	0494562154	1

NUMEREPR	NOMRREPR	PRENREPR	ADREREPR	CODPREPR	VILLREPR
1	Durant	Patrick	56, Av du Lycée	83000	Fréjus
2	Dupont	Eric	45, Av du Collège	06000	Nice
3	Richard	Bernard	12, Av de l'Ecole	13001	Marseille

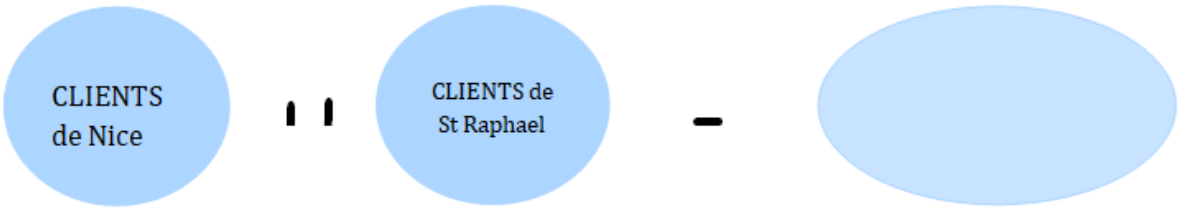
NUMECLIE	NOMCLIE	ADRECLIE	CODPCLIE	VILLCLIE	TELECLIE	NUMEREPR	NOMRREPR	PRENREPR	ADREREPR	CODP REPR	VILLREPR
100001	Digit Equipement	25, rue des Pins	06200	Valbonne	0493564578	2	Dupont	Eric	45, Av du Collège	06000	Nice
100002	F.D.T	10, Impasse de la Poste	06000	Nice	0493564545	2	Dupont	Eric	45, Av du Collège	06000	Nice
100010	Ecole Pigier	10, av J Médecin	06200	Nice	0493656562	2	Dupont	Eric	45, Av du Collège	06000	Nice
100011	Forum Immobilier	5, Av de Grasse	06500	Valbonne	0493252525	2	Dupont	Eric	45, Av du Collège	06000	Nice
100020	Habitable	2, av de Valbonne	06200	Nice	0493565656	2	Dupont	Eric	45, Av du Collège	06000	Nice
200005	Lycée Victor Hugo	Bd Desplaces	13001	Marseille	0491525269	3	Richard	Bernard	12, Av de l'Ecole	13001	Marseille
300010	A.R.T	20, rue de la mer	83000	Toulon	0494562154	1	Durant	Patrick	56, Av du Lycée	83000	Fréjus

1.2. Opérateurs ensemblistes

Les opérateurs ensemblistes <u>ne peuvent s'appliquer que sur des tables à structure identique.</u>

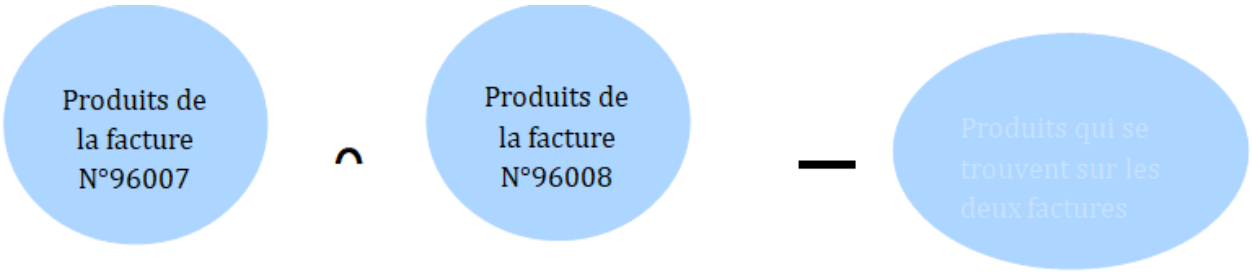
- **Union.**

L'union permet de faire la somme de deux ou plusieurs relations.



- **Intersection.**

L'intersection permet de déterminer les occurrences communes à deux relations.



- **Différence**

Faire la différence entre une relation R1 et une relation R2 consiste à prélever de la relation R1 les occurrences identiques à celles de R2 afin de ne conserver que les occurrences spécifiques à R1.

2. Le processus de construction d'une requête d'interrogation.

1) Observer le schéma relationnel et repérer les liens éventuels entre les tables relationnelles (clé étrangère et clé primaire).

2) Analyser la requête et déterminer les opérations algébriques à réaliser (sélection, projection, jointure)

ATTENTION : Si N tables sont nécessaires pour réaliser la requête, il y aura N-1 jointures.
Pensez à minimiser le nombre de tables dans la réalisation de la requête.

3) Ecrire la requête dans le langage demandé.

4) Exécuter et vérifier la cohérence du résultat obtenu

3. Les outils MySQL

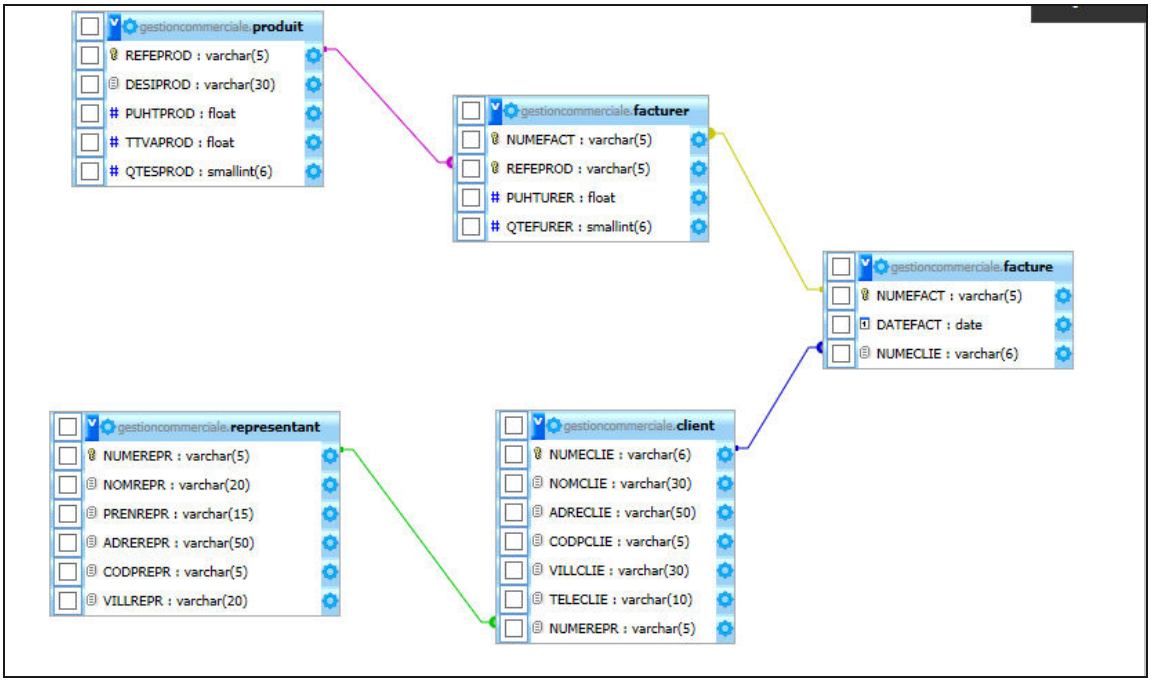


Figure 1 : Schéma relationnel sous MYSQL

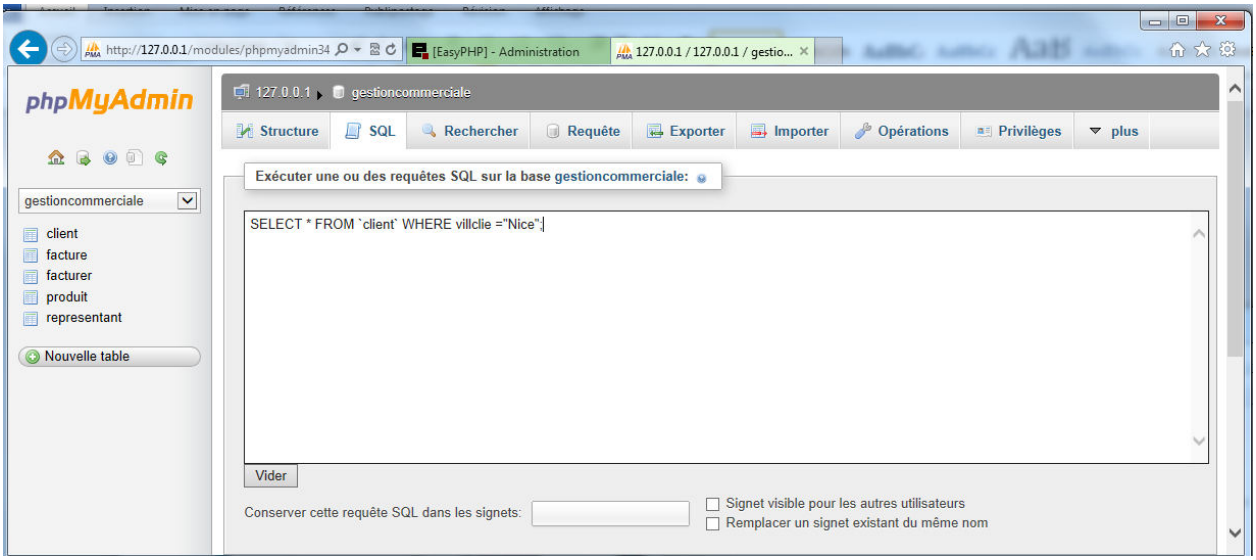


Figure 2 : Requête dans MySQL

4. Le S.Q.L (StructuredQueryLanguage).

4.1. Comment questionner la base de données ?

Exemple :*Liste des clients (numéro, nom, ville) domiciliés à Nice et nom de leur représentant*.

Quelles sont des données à afficher (projection) ?	
Dans quelle(s) table(s) se trouvent ces données ?	
Y-a-t-il des critères de sélection à respecter ?	

Une requête SQL se construit de la façon suivante : PROJECTION, (JOINTURES), (SELECTION)

La requête précédente s'écrira :

Ou

NB : Le langage S.Q.L permet d'interroger la base de données à partir de requêtes élémentaires mais également de gérer complètement la base(création, modification, mise à jour…).

4.2. Structures de base d'une requête d'interrogation

- <u>Forme générale d'une requête SQL</u>

- **SELECT** Colonne1 [AS Alias1], [Colonne2 [AS Alias2],……
- **FROM** Table1 **INNER JOIN** Table2 **ON** Table1.Colonne3 = Table2.Colonne3 [**INNER JOIN** Table3 **ON** Table3.Colonne4 = Table2.Colonne4…]

OU

FROM Table1,Table2, Table3

WHERE Table1.Colonne3 = Table2.Colonne3

AND Table3.Colonne4 = Table2.Colonne4…

- **[WHERE Condition1][AND Condition2 | OR Condition3]**
- **[ORDER BY Colonne5 [<u>ASC</u>|DESC], Colonne6 [<u>ASC</u>|DESC]…];**

4.2.1. Opérateurs relationnels.

4.2.1.1. La Projection.

Afficher la liste des clients avec le numéro, le nom et la ville.

4.2.1.2. La Sélection.

Il existe plusieurs façons d'exprimer une condition dans le cadre d'une sélection (restriction)

- **Avec le "="**

Afficher le numéro, le nom et le numéro de téléphone des clients de Nice

- **Avec" <>", "<", ">",>=, <=**

Afficher les produits (référence et désignation) dont le prix ht dépasse 150 €

Remarque : si le résultat d'une requête renvoie toutes les colonnes d'une table, il faudra utiliser l'étoile après le SELECT

Exemple : *Afficher toutes les colonnes de la table produit pour tous ceux dont le prix ht est supérieur à 150€*

4.2.1.3. La Jointure.

Afficher les clients (numéro, nom, ville) de Nice avec le numéro et le nom de leur représentant.

Il y a 2 façons de formuler une jointure entre 2 tables.

- Dans le « FROM …….WHERE….. [AND]"

- Dans le "FROM… …..INNER JOIN…….ON …….."

ATTENTION : Le nom de colonne(champ) doit être préfixé par le nom de table lorsque ce champ appartient à plusieurs tables.

4.2.2. Les opérateurs ensemblistes.

4.2.2.1. L'Union.

Qui sont les clients habitant de Nice ou St Raphael ?

- Avec "UNION".

- Avec l'opérateur "OR"

- Avec l’opérateur “IN”

4.2.2.2. L'intersection.

Quelles sont les factures éditées en 2018 ?

- Avec "INTERSECT".

- Avec l'opérateur "AND".

- Le “BETWEEN …… AND…….”.

4.2.2.3. La différence.

Qui sont les clients qui n’habitent pas Nice ou St Raphael ?

- Avec "EXCEPT".

- Avec "NOT IN".

4.3. Les spécificités de SQL.

4.3.1. Spécificités de certains prédicats SQL.

4.3.1.1. Le surnom de table.

Il permet abréger le nom d'une table en une seule lettre et ainsi de raccourcir l'écriture d'une requête.

Il suffit de préciser ce surnom dans la clause FROM après le nom de la table.

Ce surnom sera utilisé à la place du nom de la table, dans les requêtes ou la précision du nom de la table est obligatoire.

Afficher les clients (numéro, nom, ville) de Nice avec le nom de leur représentant.

```
SELECT C.NumeClie, C.NomClie, C.VillClie, R.NomRepr
FROM REPRESENTANT R, CLIENT C
WHERE R.NumeRepr = C.NumeRepr
AND C.VillClie='Nice';
```

4.3.1.2. Surnom ou alias de champs avec "AS".

Il s'agit de renommer certains champs afin de modifier le titre de la colonne lors de l'affichage du résultat de la requête.

```
SELECT NomClie as Nom, NomRepr as Représentant
FROM REPRESENTANT, CLIENT
WHEREREPRESENTANT.NumeRepr = CLIENT.NumeRepr
AND CLIENT.VillClie='Nice'
```

Le surnom est souvent utilisé pour nommer un champ calculé.

Afficher pour chaque produit, sa désignation, son prix HT, et son prix TTC

```
SELECT NomProd as Désignation,pxuht as Prix_HT ,(pxuht *(1+txtva)) as Prix_TTC
FROM PRODUIT ;
```

Remarque : un alias ne peut pas contenir d'espace. Si l'on veut utiliser un nom composé il faut écrire l'alias à l'aide d'un _.

Certains SGBDR acceptent l'écriture d'un alias avec des espaces à condition que le nom composé soit écrit a l'intérieur de [ ].

Exemple :

```
SELECT NomProd as Désignation, (pxuht *(1+txtva)) as [PrixTTC]
FROM PRODUIT ;
```

4.3.2. Le tri : ORDER BY

Permet d'afficher le résultat d'une requête selon un ordre précis comme par exemple l'ordre alphabétique, du plus grand au plus petit, …..

Le tri peut s'appliquer sur un champ de type texte ou numérique.

Exemple : *Afficher les clients et leur représentant respectif dans l'ordre alphabétique des clients.*

```
SELECT NumeClie, NomClie, VillClie, NomRepr
FROM REPRESENTANT INNER JOIN CLIENT ON REPRESENTANT.NumeRepr = CLIENT.NumeRepr
WHERE CLIENT.VillClie='Nice'
ORDER BY NomClie;
```

Remarque :

Par défaut, un tri s'effectue par ordre croissant. Pour trier dans un ordre décroissant, il faut préciser DESC après le champ concerné.

Exemple : ORDER BY NomeClie DESC

4.3.3. "Like" ou "Not Like".

Donner le numéro, le nom, la ville des clients domiciliés dans le Var

% (ou * selon les SGBDR) remplace toute chaîne comprenant entre zéro et un nombre quelconque de caractères.

_ (caractère de soulignement) remplace n'importe quel caractère.

Donner la liste des noms de représentants dont le nom contient un « u » en deuxième position.

4.3.4. Le "isNull".

Il indique l'absence de valeur d'une colonne de table autrement dit un champ non renseigné.

Qui sont les clients qui ne nous ont pas communiqué leur numéro de téléphone ?

4.3.5. "Distinct".

Le résultat de certaines requêtes affiche plusieurs fois les mêmes données sur des lignes différentes.

Pour éviter cet affichage multiple et inutile, il convient de rajouter DISTINCT après le SELECT.

Exemple : ***Lister les produits qui ont été facturés***

4.3.6. Les regroupements et les fonctions associées.

4.3.6.1. Les fonctions de regroupement

Les fonctions de regroupement permettent d'effectuer des calculs portant sur un ou des ensembles de données. Il existe cinq principales fonctions de regroupements : MAX, MIN, COUNTpour déterminer le nombre, AVG pour calculer la moyenne et SUM pour calculer unesomme.

Exemple1 :Combien y- a- t'il de clients dans la base ?

Exemple 2 : Quel est le prix moyen d'un produit ?

ATTENTION	**Les fonctions de regroupement COUNT, MAX et MIN peuvent être utilisées sur tout type de champ.** **Les fonctions de regroupement AVG et SUMne peuvent porter que sur des champs numériques**

4.3.6.2. Le "GROUP BY".

La clause *Group by* permet d'effectuer des calculs, non pas sur toute la table, mais sur des sous ensembles de table.

Exemple 1 : Donner pour chacune des factures, le nombre de lignes de facture.

Exemple 2 : Donner le nombre de clients habitant dans chacune des différentes villes.

Le ou les champs qui apparaissent au niveau du 'Select' et qui ne sont pas utilisés dans une fonction de regroupement doivent impérativement se retrouver au niveau du GROUP BY.

4.3.6.3. Le "HAVING".

La clause *HAVING* agit comme le filtre *WHERE* mais permet de filtrer les résultats calculés dans les opérations de regroupement.

Exemple 1 : Donner les numéros de factures qui ont un nombre de lignes supérieur à 2.

```
SELECT NumeFact, COUNT(*)AS NbreLigneFacture
FROM FACTURER
GROUP BY NumeFact
```

Exemple 2 : Donner les villes dans lesquelles il y a plus de 100 clients

```
SELECT VillClie, COUNT(*) AS NbreClient
FROM CLIENT
GROUP BY VillClie
```

4.3.6.4. Forme générale d'une requête.

```
SELECT [DISTINCT|ALL] Colonne1 [AS Alias1], [DISTINCT|ALL]
Colonne2 [AS Alias2,…],[FonctionDeRegroupement(Colonne|*)]

FROM Table1 [Alias1] [, Table2 [Alias2]…]| Table1 [Alias1] INNER
JOIN Table2 [Alias2] ON Table1.Colonne3 = Table2.Colonne3 [INNER
JOIN Table3 ON Table3.Colonne4 = Table2.Colonne4…]

[WHERE Condition1][AND Condition2 | OR Condition3]

GROUP BY Colonne1 [, Colonne2]…

[HAVING Condition1][AND Condition2 | OR Condition3]

[ORDER BY Colonne5 [ASC|DESC], Colonne6 [ASC|DESC]…];
```

4.3.7. Les Sous-requêtes.

Une sous-requête est une requête qui est imbriquée dans autre une requête, ou dans une autre sous-requête. On utilise une sous interrogation à chaque fois qu'un attribut (un champ) de la requête principale doit récupérer sa valeur dans une liste définie par un autre bloc SELECT…FROM…WHERE.

4.3.7.1. Sous requêtes non corrélées.

Sous requêtes de base retournant une ou plusieurs valeurs

- **Avec le signe« = ».**

Quel est le produit le plus cher ?

```
SELECT Refeprod, desiprod
from produit
WHERE puhtprod
```

Avec le signe« = »,le résultat de la sous-requête ne doit renvoyerqu'une seule ligne.

- **Avec le « IN » ou « NOT IN ».**

Donner l'ensemble des informations des représentants qui suivent des clients qui habitent la ville de "Nice".

```
SELECT *
FROM REPRESENTANT
WHERE NumeRepr
```

Donner les numéros de factures pour lesquelles les produits facturés ont une quantité en stock >5.

```
SELECT DISTINCT (FACTURER.NumeFact)
FROM FACTURER
WHERE FACTURER.RefeProd
```

<u>Sous requêtes renvoyant une table (dans le « FROM »)</u>.

Donner le montant moyen d'une facture parmi l'ensemble des factures existantes

+ Options

	NUMEFACT	REFEPROD	PUHTURER	QTEFURER
Éditer Copier Supprimer	96001	10001	152.45	1
Éditer Copier Supprimer	96001	10003	91.47	2
Éditer Copier Supprimer	96002	20005	1372.04	1
Éditer Copier Supprimer	96003	10003	91.47	2
Éditer Copier Supprimer	96003	20002	609.8	2
Éditer Copier Supprimer	96003	30006	2439.18	2
Éditer Copier Supprimer	96004	30007	2850.8	1
Éditer Copier Supprimer	96005	10001	121.96	5
Éditer Copier Supprimer	96005	30003	1219.59	5
Éditer Copier Supprimer	96006	10001	121.96	15
Éditer Copier Supprimer	96007	30009	4116.12	5
Éditer Copier Supprimer	96008	20005	1372.04	3
Éditer Copier Supprimer	96008	20010	914.69	3
Éditer Copier Supprimer	96008	30003	1219.59	3
Éditer Copier Supprimer	96008	30009	3811.23	3
Éditer Copier Supprimer	96009	20010	914.69	5
Éditer Copier Supprimer	96010	10001	121.96	1
Supprimer	96010	20002	609.8	1

Console de requêtes SQL

La requête va se faire à partir de la table FACTURER

Il va falloir calculer le montant de <u>chaque facture</u> et ensuite calculer la moyenne de ces montants.

On va décomposer le travail.

1. on calcule dans une première requête, le montant de chaque facture (MtFact) et on nomme cette requête TOTALFACTURES

```
SELECT NumeFact, SUM(QteFurer *
PuhtUrer) as Mtfact

FROM FACTURER

GROUP BY NumeFact
```

+ Options

NUMEFACT	MtFact
96001	335.38999938964844
96002	1372.0400390625
96003	6280.899841308594
96004	2850.800048828125
96005	6707.749824523926
96006	1829.3999862670898
96007	20580.6005859375
96008	21952.649963378906
96009	4573.450012207031
96010	731.7599868774414

2. On calcule le montant moyen d'une facture

SELECT AVG(MtFact) As Moyenne
FROM TOTALFACTURES

+ Options

Moyenne
6721.474028778076

SELECT ROUND(AVG(MtFact),2) As moyenne FROM TOTALFACTURES

moyenne
6721.47

En une seule requête, le tout s'écrira :

SELECTAVG(MtFact) As Moyenne FROM (SELECTSUM(QteFurer * PuhtUrer) as Mtfact FROM FACTURER GROUP BY NumeFact)TOTALFACTURES	+ Options Moyenne 6721.474028778076
SELECTROUND(AVG(MtFact),2) As MoyenneFROM (SELECTSUM(QteFurer * PuhtUrer) as Mtfact FROM FACTURER GROUP BY NumeFact)TOTALFACTURES	moyenne 6721.47

<u>Sous requêtes renvoyant une donnée calculée</u>

- **Dans le « SELECT ».**

Donner les numéros de facture, leurs montants HT respectifs et le montant global HT de toutes les factures.

Dans la requête, on écrira une sous requête qui permettra de calculer le montant global de toutes les factures

SELECTNumeFact,SUM(PuhtUrer* QtefUrer) as MontantFacture, (SELECTSUM(PuhtUrer*QtefUrer)from FACTURER) As MontantTotalFactures FROM FACTURER GROUP BY NumeFact	+ Options NumeFact / MontantFacture / MontantTotalFactures 96001 / 335.38999938964844 / 67214.74028778076 96002 / 1372.0400390625 / 67214.74028778076 96003 / 6280.899841308594 / 67214.74028778076 96004 / 2850.800048828125 / 67214.74028778076 96005 / 6707.749824523926 / 67214.74028778076 96006 / 1829.3999862670898 / 67214.74028778076 96007 / 20580.6005859375 / 67214.74028778076 96008 / 21952.649963378906 / 67214.74028778076 96009 / 4573.450012207031 / 67214.74028778076 96010 / 731.7599868774414 / 67214.74028778076

- **Dans le « WHERE ».**

Donner les numéros et dates de facture dont les quantités de produitsfacturés sont supérieures à la moyenne des quantités facturées de l'ensemble des Factures.

`SELECT DISTINCT FACTURE.NumeFact, FACTURE.DateFact`
`from FACTURE INNER JOIN FACTURER ON FACTURE.NumeFact=FACTURER.NumeFact`
`WHERE QtefUrer>(SELECT AVG(QteFurer) from facturer)`
	+ Options
	NumeFact / DateFact
	96005 / 2019-10-01
	96006 / 2019-01-20
	96007 / 2019-01-22
	96009 / 2019-01-25

- **Dans le « HAVING ».**

Donner les factures (numéros ,montant) dont le montant est supérieur à la moyenne du montant desfactures.

```
SELECT Numefact, sum(QtefUrer*PuhtUrer)
FROM FACTURER
GROUP BY NumeFact
HAVING sum(QtefUrer*PuhtUrer)>(SELECT AVG(MtFact)AS Moyenne from  (Select SUM(FACTURER.QteFurer* FACTURER.PuhtUrer) as MtFact from FACTURER group by NumeFact)TOTALFACTURE)
```

+ Options

Numefact	sum(QtefUrer*PuhtUrer)
96007	20580.6005859375
96008	21952.649963378906

4.4. Les Vues.

Une vue est une table virtuelle. Elle peut donc être utilisée pratiquement comme une table. Elle ne stocke aucune donnée mais uniquement le résultat d'une requête.

4.4.1. Création d'une vue.

Forme générale.

```
CREATE VIEW NomVue AS OrdreSelect
```

L'Ordre Select est une requête qui permet la création de la vue

Exemple : Créer une vue affichant les clients et le nom de leur représentant

```
CREATE VIEW ClientsEtRepresentants as
        Select NumeClie, NomeClie, NomeRepr
        from Client,Representant
        WhereClient.Numerepr=Representant.NumeRepr;
```

4.4.2. Suppression d'une vue.

- **Forme générale.**

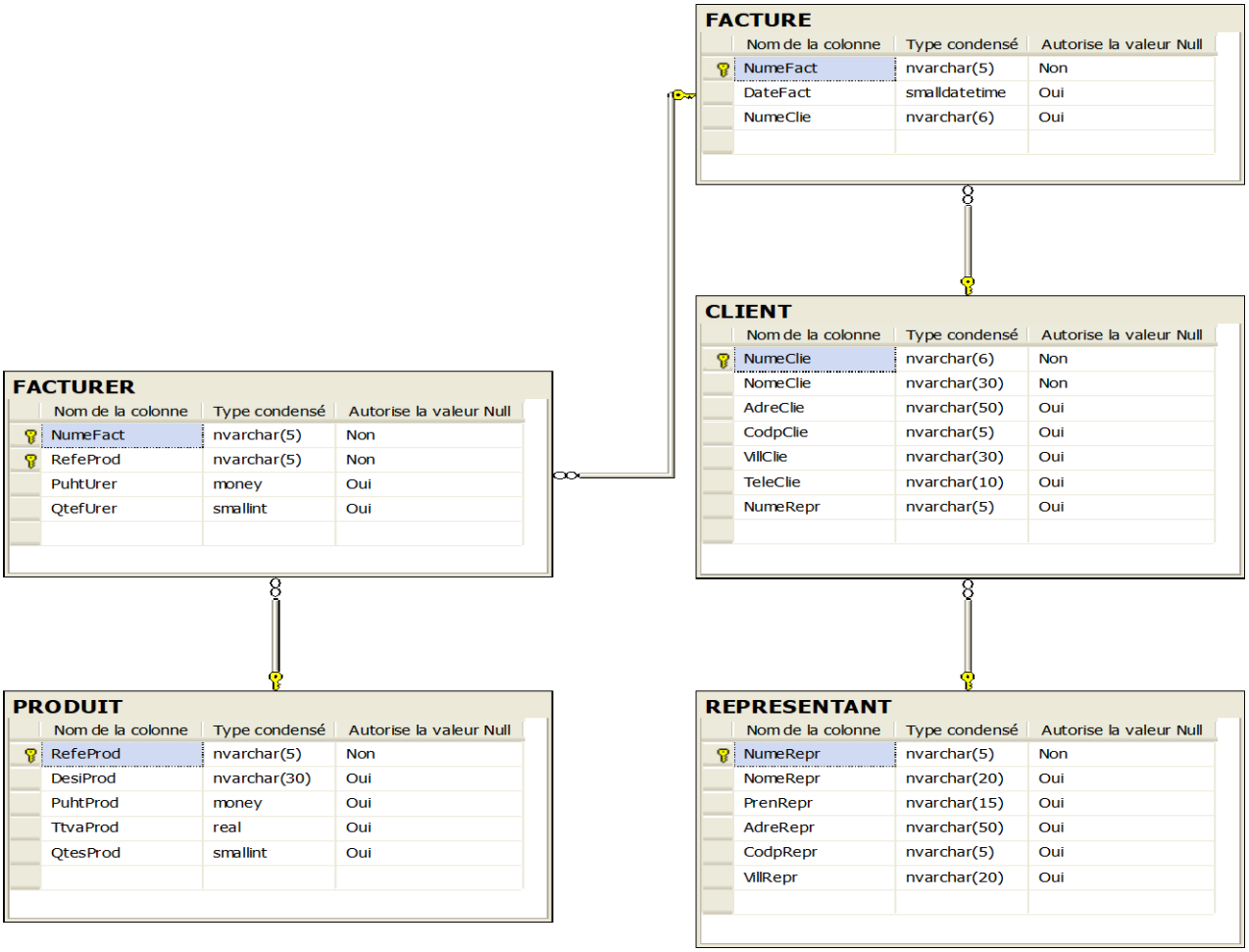
```
DROP VIEW NomVue
```

- **Exemple.**

```
DROP VIEW ClientsEtRepresentants
```

5. Applications

EXERCICE 1 : Interrogation de la BD-R « Gestion Commerciale ».



<u>Travail à faire</u>:

Créer les requêtes d'interrogation ci-dessous

1. Donner le numéro, le nom, et le numéro de téléphone de l'ensemble des clients.
2. Donner la référence et la désignation de l'ensemble des produits.
3. Donner le numéro de facture et la date de facture de l'ensemble des factures triées par ordre croissant de date de facture.

---

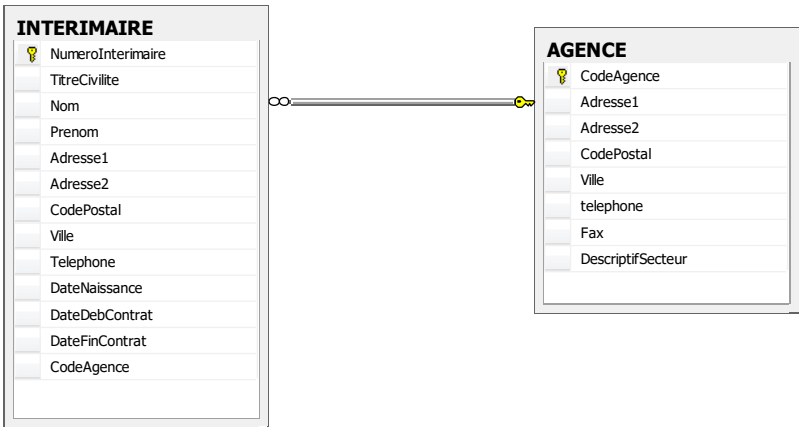
4. Donner la référence, désignation, PU H.T., taux de T.V.A. et Quantité en stock des PRODUITS dont la quantité en stock est inférieure ou égale à 10.
5. Donner le numéro, le nom, l'adresse, le code postal, la ville, le téléphone et le numéro de représentant des clients qui habitent la ville de Nice.
6. Donner le numéro, le nom, l'adresse, le code postal et la ville des représentants qui ont comme code postal 83600.
7. Donner la référence, désignation, PU H.T., taux de T.V.A. et la Quantité en stock des produits qui ont un prix unitaire H.T. supérieur ou égal à 1600€.
8. Donner le numéro de facture, la date de facture et le numéro de client des factures émises depuis le 21/01/2019, par ordre croissant de date de facture.
9. Donner le numéro de facture et les quantités facturées du PRODUIT 10003, par ordre croissant de numéro de facture.
10. Donner la référence, désignation, PU H.T., taux de T.V.A. et la quantité en stock des produits dont la quantité en stock est inférieure ou égale à 10 et dont le PU H.T. est supérieur à 1600€.

11. Donner le numéro, le nom, l'adresse, le code postal, la ville, le téléphone, le numéro de représentant des CLIENTS qui habitent la ville de Nice ou de Valbonne.
12. Donner les numéros des factures correspondant à l'article 30009.
13. Donner les références et les quantités en stock des PRODUITS dont la quantité en stock est inférieure ou égale à 5 triés par ordre décroissant de quantité en stock
14. Donner les numéros des factures correspondant à l'article 30009 ou 10003.
15. Donner le numéro, le nom, l'adresse, le code postal, la ville, le téléphone, le numéro de représentant, le nom de représentant et la ville du représentant de l'ensemble des clients.
16. Donner les numéros de facture, les dates de factures et les noms des clients pour l'ensemble des factures émises à ce jour.
17. Donner les dates de facture, les numéros de client et les références de l'ensemble des produits facturés.
18. Donner les références et le PU H.T. des PRODUITS vendus à plus de 4 exemplaires.
19. Donner le numéro de client et le nom de client, des clients du représentant "Dupont" par ordre croissant sur le nom du client.

---

20. Donner pour chacune des factures le montant total H.T.
21. Donner le nom, le prénom du représentant qui suit les clients qui ont été facturés le 01/05/2019.
22. Donner les références, les désignations des produits facturés au client 100011.
23. Donner les numéros des clients à qui on a facturéun produit dont le prix unitaire est supérieur à 1500 €.
24. Donner le nom, la ville des clients, le numéro et la date de facturation des factures contenant l'article 10001.
25. Donner pour chacun des articles facturés, sa référence et sa quantité moyenne facturée.
26. Donner pour chacun des représentants le nombre de clients qu'il suit.
27. Donner le nom des clients qui n'ont pas de numéro de téléphone.
28. Donner les clients avec lesquels on a réalisé un chiffre d'affaire supérieur à 5000€.
29. Donner les références des factures dont le total H.T. est supérieur à 500€.
30. Donner les références des factures pour lesquelles le produit 10001 a été facturé pour une quantité supérieure à 2.

EXERCICE 2 : Interrogation de la base de données « SRTT ».



Travail à faire :

Créer les requêtes d'interrogation de la base de données SRTT ci-dessous :

1.Donner le Code de l'Agence, Ville, Téléphone des agences.

2. Donner le Titre de Civilité, Nom, Prénom, Date de Naissance des intérimaires.
3. Donner N°, Nom, Date de Naissance des intérimaires classés par ordre croissant de date de naissance.

---

4. Donner l'ensemble des informations de l'agence qui a comme N° de Téléphone le 02/45/12/22/25
5. Donner le N°, Nom, Prénom des intérimaires qui sont nées avant le 01/01/1975.
6. Donner le N°, Nom, Date de Naissance des INTERIMAIRE qui ont comme Prénom «Jean».
7. Donner le N°, Nom, Prénom des intérimaires qui habitent dans la Ville de «Rennes».
8. Donner le N°, Nom, Prénom des intérimaires de Sexe Féminin.
9. Donner N°, Nom, Prénom, date de naissance des intérimaires qui sont nées dans les années 70. Afficher cette liste du plus jeune au plus âgé.
10. Donner N°, Nom, Prénom des intérimaires qui sont rattachés à l'agence ayant pour code VN et qui sont nés après le 01/01/1975.
11. Donner le N°, Nom, Prénom, Date de Naissance et ville des intérimaires qui habitent la ville de « Brest » ou de « Vannes » et qui sont nés entre 1975 et 1980.
12. Donner le N°, Nom, Prénom des intérimaires qui habitent dans la Ville de «Rennes» ou «Brest» ou «Vannes»
13. Donner le N°, Nom, ville et date de naissance des intérimaires qui n'habitent pas à Brest et qui sont nées avant le 01/12/80.

---

14. Donner N°, Nom, Prénom, Date de Naissance, Ville d'Agence, N° de Téléphone d'Agence de l'ensemble des intérimaires.
15. Donner le N°, Nom, Date de Naissance, Code d'Agence des intérimaires qui sont dans le secteur «Morbihan».
16. Donner le Code, Adresse des agences dont le n° Téléphone commence par 0245.
17. Donner le N°, Nom, Prénom des intérimaires dont l'agence a un N° de Téléphone commençant par 0245 ou 0244.

---

18. Donner le nombre d'intérimaires qui habitent la ville de «Brest».
19. Donner pour chacune des agences le nombre d'intérimaires.
20. Donner le nombre d'intérimaires habitant dans chacune des différentes villes.
21. Dans quelle(s) ville(s) y- a- t-_il au moins 2 intérimaires ?

EXERCICE N°3 : Requêtes « Cité nouvelle ».

Cité Nouvelle est une société anonyme d'HLM (Habitations à Loyer Modéré), présente dans les départements de la Loire et du Rhône. Outre son activité principale de bailleur social (mise à disposition d'habitations donnant au locataire la jouissance de son bien contre un loyer), elle conduit de nombreuses opérations d'aménagement et de restructuration urbaine. Elle est l'un des plus importants organismes de logements sociaux de la Loire. La société envisage de mettre en place une demande de logement par l'intermédiaire d'un formulaire complété en ligne par le candidat. Cette solution permettra d'activer plus rapidement la recherche d'un logement car les données saisies par l'internaute seront contrôlées et transmises automatiquement au Centre d'études technique de l'équipement. Elles seront également enregistrées dans la base de données de Cité Nouvelle.

Pour traiter les demandes de logement formulées par internet, M. Nguyen a été conduit à apporter quelques modifications à la base de données de l'entreprise. Il vous propose le schéma relationnel ci-dessous.

DEMANDEUR (id,nom, nomMarital, prenom, titre, dateNaissance, adresse, codePostal, ville, portable, adresseMel, telephoneDomicile, telephoneTravail, revenu)

Clé primaire : id

MOTIFDEMANDE (id, libelle)

Clé primaire : id

SECTEURDEMANDE (id, nom)

Clé primaire : id

DEMANDELOGEMENT (idDemandeur, dateDemande, idSecteurDemande, idMotifDemande, nombrePersonnes, nombreEnfants, dateDisponibilite, commentaire)

Clé primaire : idDemandeur, dateDemande

Clés étrangères : idDemandeur en référence à id de Demandeur,

IdSecteurDemande en référence à id de SecteurDemande,

Idmotifdemande en référence à id de MotifDemande

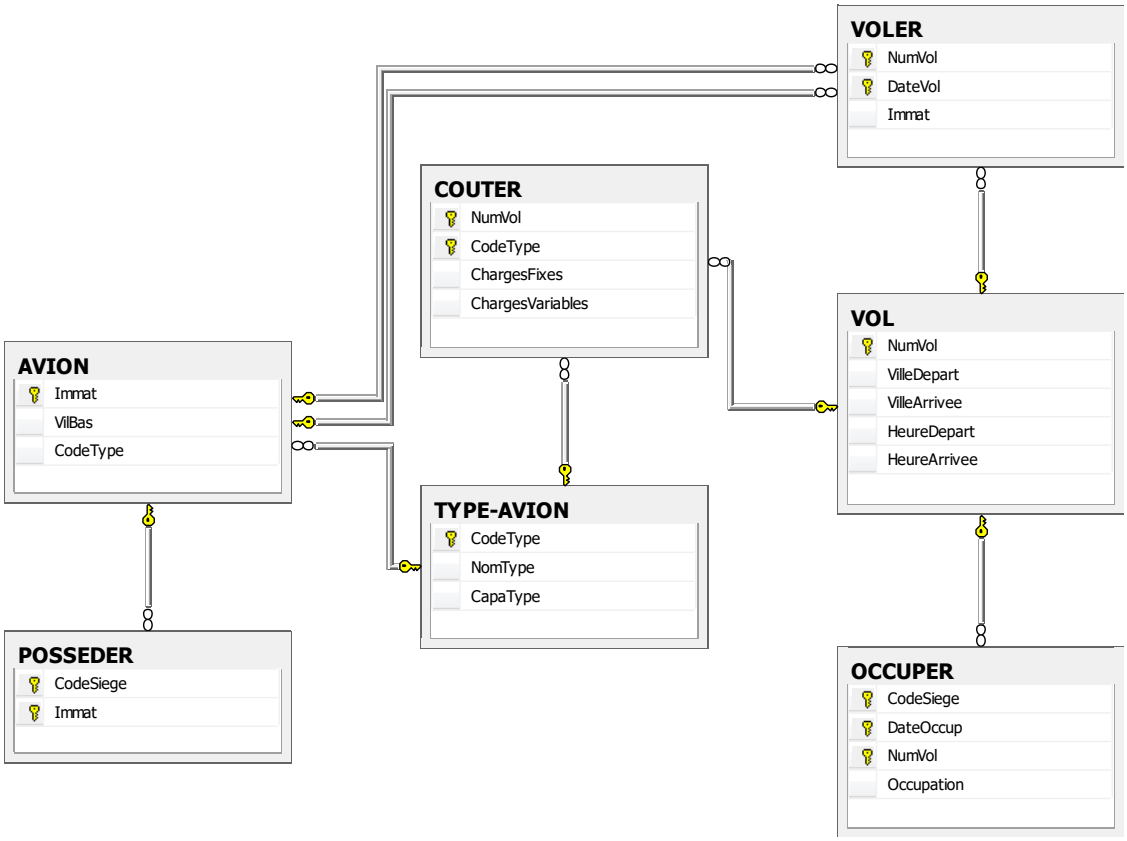
Un logement vient de se libérer dans le secteur Loire-Sud. L'attachée commerciale veut donc savoir quelles personnes (nom du demandeur et les dates des demandes) ont formulé une demande de logement dans ce secteur. Par ailleurs, pour choisir les lieux d'implantation des nouveaux projets de construction, l'attachée commerciale désire connaître les secteurs (id du secteur, nom du secteur et nombre de demandes) qui présentent plus de 150 demandes, c'est à dire les plus attractifs.

<u>Travail à faire</u> : Rédiger en langage SQL, les deux requêtes permettant d'obtenir ces informations.

EXERCICE N°4 : Interrogation de la base de données « Air Provence ».

La compagnie aérienne AIR PROVENCE basé dans le sud de la France, souhaite mettre en place un système d'information lui permettant de connaître notamment les taux réels d'occupation de ses vols et par donc par voie de conséquence la rentabilité de ses vols.

Un schéma relationnel représenté ci-dessous a été établi :



<u>Travail à faire</u> :

Créer les requêtes d'interrogation de la base de données *Air-Provence* ci-dessous :

1. Quels sont les avions (Immatriculation et ville de base) qui ont une capacité strictement supérieure à 250 personnes ?
2. Combien de places ont été occupées dans le vol AF3006 du 19/01/2018 ?
3. Que représente, en termes de capacité, l'ensemble des avions de la flotte de type « A320 » ?
4. Quelle est la capacité moyenne de l'ensemble des avions tous types confondus ?
5. Quel est le total des charges fixes des vols qui sont partis de la ville de Nice à destination de Paris ?
6. Quel est le coût minimum des vols de la compagnie ?
7. Quel est le coût maximum des vols de la compagnie qui partent de Nice en direction de Lyon ?
8. Combien de sièges n'ont pas été occupés le 26/01/2018 ?
9. Combien y a-t-il d'avions dans la flotte pour chacun des différents types d'avions ?
10. Quels sont les moyennes des charges de chacun des différents types d'avion ?

11. **Combien y a-t-il eu de places occupées dans chacun des vols ?**
12. **Combien y a-t-il de types d'avion qui ont plus de 3 avions ?**
13. **Quels sont les vols dont les charges fixes n'ont pu être déterminées ?**

EXERCICE N°5 : Requêtes sur la base de données « STAT ».

La société ÉDIRVA dont le siège social est à Niort (79) est une coopérative à capital variable spécialisée dans l'échange de documents informatisés (ÉDI). Dans le domaine des assurances, ÉDIRVA met en relation les assureurs, les experts, les carrossiers et les dépanneurs, grâce à son service FLÈCHEAUTO.

La société KIMASSUR pratique la variabilité des cotisations. À partir de différents indices statistiques et de multiples simulations, le service "Contrôle de gestion" fixe chaque année les nouveaux tarifs à appliquer. En raison du statut juridique de KIMASSUR, ces nouveaux tarifs doivent être calculés de manière à réaliser l'équilibre entre :

les charges constituées par le coût des sinistres et les frais de fonctionnement,
les ressources constituées par les cotisations et les revenus des placements.

La base de données relationnelle STAT fournit des valeurs statistiques sur les différents contrats automobiles. Un extrait du schéma relationnel de cette base est fourni ci-dessous :

MODELEAUTO (CodeModèle, LibelléModèle, CodeMarque, LibelléMarque)
Clé Primaire : CodeModèle.

STATISTIQUEMODELE (CodeModèle, Année, Mois, NbSinistres, CumulCoûtSinistre, MttTotalCotisation)
Clés Primaires : CodeModèle, Année, Mois.
Clé Etrangère : CodeModèle en référence aCodeModèle de MODELEAUTO

<u>Travail à faire :</u>

1. Formuler en SQL la requête permettant d'obtenir les informations suivantes : nombre annuel de sinistres, cumul annuel des coûts des sinistres et montant total annuel des cotisations, par modèle et par année. La requête affichera le code et le libellé du modèle, le libellé de la marque, l'année et les trois nombres recherchés.

2. Formuler en SQL une requête fournissant la liste des modèles dont le ratio « de couverture » (=Total des cotisations / Coût total des sinistres) a, pour l'année 2019, une valeur supérieure à 4. La requête affichera le libellé du modèle et la valeur du ratio de couverture.

EXERCICE N°6 : Requêtes sur la « Gestion des Escales ».

Le port de Caen-Ouistreham accueille de nombreux navires qui transportent des marchandises importées ou exportées. Les activités du port sont gérées par la Capitainerie dirigée par un commandant. La Capitainerie s'occupe de la circulation des navires (entrées et sorties), supervise les opérations de manutention touchant aux cargaisons (chargement et déchargement du fret) et gère les infrastructures d'accostage et de stockage (grues, quais, hangars, etc.).

Pour automatiser les traitements que nécessite l'enregistrement du séjour d'un navire, le Commandant a fait appel à une société de service. Une base de données relationnelle a été conçue dont le schéma relationnel est fourni ci-dessous :

EMPLACEMENT (NumEmplac, NomEmplac, LongueurEmplac, TirantEauMax, TarifJournalier, CodeTypeFret)
Clé Primaire : NumEmplac.
Clé Etrangère : CodeTypeFret en référence à CodeTypeFret de TYPE_FRET

BATEAU (NumBateau, NomBateau, Tonnage, LongueurBateau, TirantEau)
Clé Primaire : NumBateau.

AGENT (CodeAgent, NomAgent, AdresseAgent, TélAgent, FaxAgent)
Clé Primaire : CodeAgent.

SÉJOUR (NumSéjour, DateArrivée, DuréeSéjour, NumEmplac, NumBateau, CodeAgent)
Clé Primaire : NumSéjour.
Clés Etrangères : NumEmplac en référence à NumEmplac d'EMPLACEMENT,
NumBateau en référence à NumBateau de BATEAU,
CodeAgent en référence à CodeAgent d'AGENT.

TRANSPORTER (NumBateau, CodeTypeFret, QtéFret)
Clé Primaire : NumBateau, CodeTypeFret
Clés Etrangères : NumBateau en référence à NumBateau de BATEAU,
CodeTypeFret en référence à CodeTypeFret de TYPE_FRET.

TYPE_FRET (CodeTypeFret, LibelléFret)
Clé Primaire : CodeTypeFret

<u>Travail à faire :</u>

À partir du schéma relationnel défini ci-dessus, formuler en SQL les requêtes qui permettront au capitaine du port :

1) **De connaître la liste des emplacements disposant d'un tirant d'eau suffisant pour accueillir le bateau nommé NORSTONE (numéro de l'emplacement, longueur, tirant d'eau maximum et code du type de fret) ;**

2) **D'afficher le nombre de séjours ayant débuté durant l'année 2020 pour chaque agent consignataire (nom de l'agent, nombre de séjours) : la liste sera triée par ordre décroissant du nombre de séjours.**

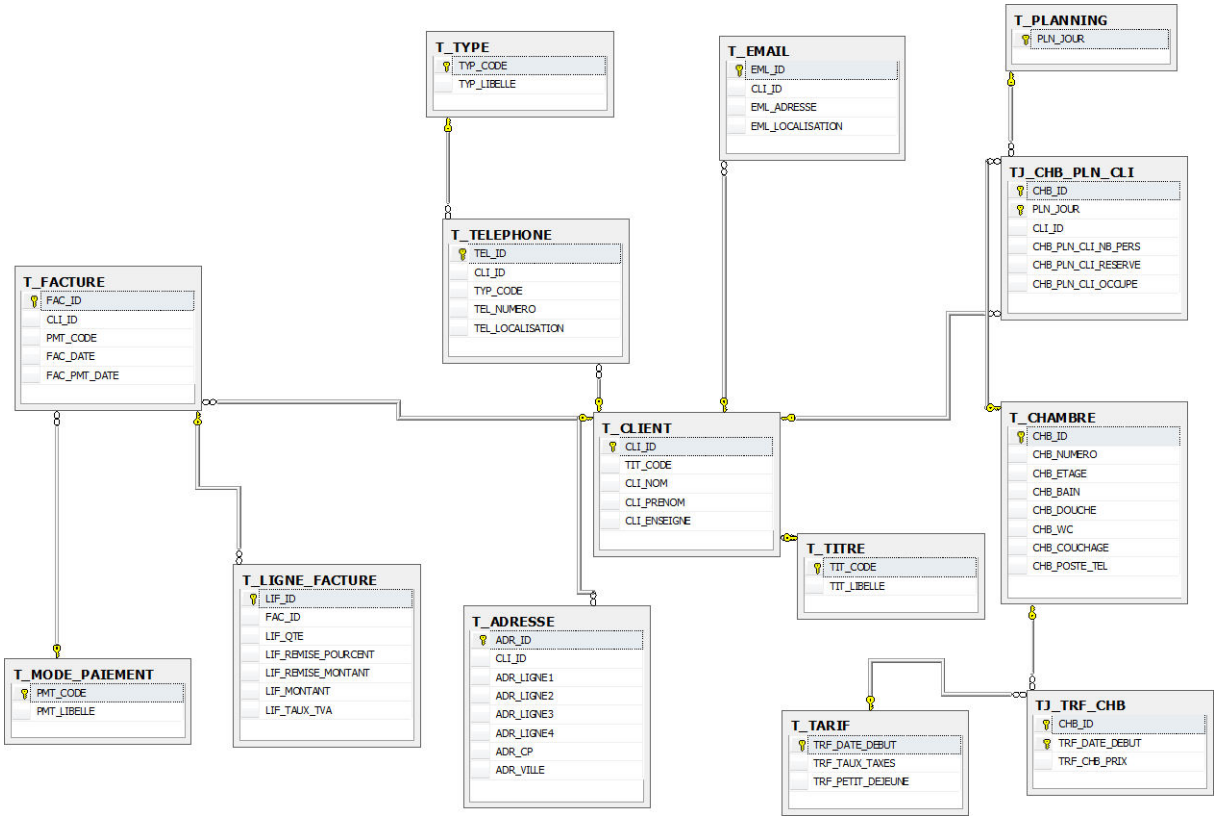
EXERCICE N°7 : Requêtes sur la base de données « Hôtel ».

L'Hôtel du Golfe situé à St Tropez utilise une base de données pour gérer ses chambres, les réservations et la facturation des séjours de ses clients.

Le modèle relationnel vous est présenté ci-dessous.

A l'aide de ce modèle et des informations complémentaires, vous devrez traiter un certain nombre de requête.

Annexe A : Modèle relationnel



Annexe B : Informations relatives à certains champs de la base.

La tableTJ_CHB_PLN_CLI permet de connaître la réservation ou l'occupation d'une chambre.

Une chambre peut avoir été réservée mais pas occupée, c'est pourquoi cette table possède les attributs CHB_PLN_CLI NB_PERS (nombre de personnes : entier)

CHB_PLN_CLI RESERVE (réservée : booléen)

CHB_PLN_CLI OCCUPE (occupe : booléen).

Une chambre, à une date donnée, ne peut être occupée que par un seul client. Mais un client peut occuper plusieurs chambres à la même date ou la même chambre à différentes dates, voire même plusieurs chambres à plusieurs dates.

Pour la table T_CLIENT : un client peut avoir plusieurs adresses, plusieurs numéros de téléphone et plusieurs e-mail. Pour le téléphone, comme pour l'e-mail, l'attribut 'localisation' permet de savoir si le téléphone est situé au domicile, à l'entreprise, etc...

Pour la table T_TITRE permet de donner un titre à une personne, parmi les valeurs 'M.' (monsieur), 'Mme.' (madame) et 'Melle.' (mademoiselle)

Pour la table T_TYPE permet de connaître le type de téléphone, parmi les valeurs 'TEL' (téléphone), 'FAX' (télécopie) et 'GSM' (portable).

Pour la table T_MODE_PAIEMENT permet de connaître le genre de paiement, parmi les valeurs 'ESP' (espèces), 'CHQ' (chèque), 'CB' (carte bancaire).

La table T_FACTURE « Payée » intègre la date du paiement d'une facture.

TRAVAIL A FAIRE :

A l'aide du schéma relationnel fourni ci-dessous, vous traiterez les requêtes suivantes :

1) Quels sont les clients qui ont une adresse email chez le FAI Free ?

2) Quels sont les clients qui n'ont pas d'enseigne ?

3) Quels sont les noms des clients, classés par ordre alphabétique, à qui on a déjà adressé une facture ?

4) Afficher les 10 villes d'où provient le plus grand nombre de clients ?

Pour cette requête, il conviendra de calculer le nombre de clients provenant de chaque ville, d'afficher ce nombre par ordre décroissant et de ne conserver, à l'affichage, que les 10 premières. Pour cela, vous terminerez la dernière instruction de la requête par » LIMIT 10 »

5) Quel est le montant moyen hors taxe des factures des clients de l'hôtel ?

6) Quelles sont les possibilités totales de couchage de l'hôtel ?

7) Quels sont les 2 modes de paiement les plus utilisés ?

8) Quels sont les clients qui n'ont pas d'adresse email ?

9) Quel est le chiffre d'affaires brut hors taxes pour le mois de juillet 2013 ?

10) Quel a été le taux d'occupation de la chambre n°2 ?

11) Quel est le délai moyen, en jours, de règlement des factures ?