

Exercice 1

Voici les tables qui composent la base de données **Immeubles** :

- **Appartements** (idAppartement, #idImmeuble, Numero, Etage, Superficie)
- **Immeubles** (idImmeuble, Nom, Adresse, NbEtages, AnneeConstruction, #idPersonne)
- **Occupants** (#idImmeuble, #idAppartement, #idPersonne, AnneeArrivee)
- **Personnes** (idPersonne, Nom, Age, Profession)

La clé étrangère idPersonne dans la table Immeubles représente la notion de gérant d'un immeuble.

Cette base de données est accessible sur www.sn-i.fr → Rubrique « Ressources pédagogiques ».

Écrivez les requêtes suivantes en SQL.

1. Nom des immeubles ayant strictement plus de 10 étages.

```
SELECT Nom
FROM Immeubles
WHERE NbEtages>10
```

2. Nom des personnes ayant emménagé avant 1994.

```
SELECT Nom
FROM Personnes
JOIN Occupants USING(idPersonne)
WHERE AnneeArrivee<1994
```

3. Qui habite le Koudalou ?

```
SELECT Personnes.Nom
FROM Personnes
JOIN Occupants ON Personnes.idPersonne=Occupants.idPersonne
JOIN Immeubles ON Occupants.idImmeuble=Immeubles.idImmeuble
WHERE Immeubles.Nom='Koudalou'
```

4. Nom des informaticiens de plus de 25 ans.

```
SELECT Nom
FROM Personnes
WHERE Profession=""
```

5. Nom des immeubles ayant un appartement de plus de 150 m².

```
SELECT DISTINCT Nom
FROM Immeubles AS i
JOIN Appartements USING(idImmeuble)
WHERE Superficie>150
```

6. Qui gère l'appartement où habite Rachel ?

```
SELECT gerant.Nom
FROM Personnes AS locataires
JOIN Occupants USING(idPersonne)
JOIN Immeubles USING(idImmeuble)
JOIN Personnes AS gerants ON gerants.idPersonne=Immeubles.idPersonne
```

Bases de données

WHERE locataires.Nom='Rachel'

7. Dans quel immeuble habite un acteur ?

```
SELECT Immeubles.Nom  
FROM Personnes  
JOIN Occupants USING(idPersonne)  
JOIN Immeubles USING(idImmeuble)  
WHERE Profession='acteur'
```

8. Qui habite un appartement de moins de 70 m2.

```
SELECT Nom  
FROM Appartements  
JOIN Occupants USING(idAppartement)  
JOIN Personnes USING(idPersonne)  
WHERE Superficie<70
```

9. Nom des personnes qui habitent au dernier étage de leur immeuble.

```
SELECT Personnes.Nom  
FROM Immeubles  
JOIN Appartements ON Immeubles.idImmeuble=Appartements.idImmeuble  
JOIN Occupants ON Occupants.idAppartement=Appartements.idAppartement  
JOIN Personnes ON Occupants.idPersonne=Personnes.idPersonne  
WHERE Etage=NbEtages
```

10. Qui a emménagé au moins 20 ans après la construction de son immeuble ?

```
SELECT Personnes.Nom  
FROM Immeubles  
JOIN Occupants ON Occupants.idImmeuble=Immeubles.idImmeuble  
JOIN Personnes ON Personnes.idPersonne=Occupants.idPersonne  
WHERE AnneeArrivee>AnneeConstruction+20
```

11. Profession du gérant du Barabas ?

```
SELECT Profession  
FROM Immeubles  
JOIN Personnes ON Immeubles.idPersonne=Personnes.idPersonne  
WHERE Immeubles.Nom='Barabas'
```

12. Couples de personnes ayant emménagé dans le même immeuble la même année.

```
SELECT p1.Nom,p2.Nom  
FROM Personnes AS p1  
JOIN Occupants AS o1 ON p1.idPersonne=o1.idPersonne  
JOIN Occupants AS o2 ON o1.idImmeuble=o2.idImmeuble  
JOIN Personnes AS p2 ON o2.idPersonne=p2.idPersonne  
WHERE o1.AnneeArrivee=o2.AnneeArrivee  
AND p1.idPersonne<>p2.idPersonne
```

13. Age et profession des occupants de l'immeuble géré par Ross ?

```
SELECT p1.Age,p1.Profession  
FROM Personnes AS p1  
JOIN Occupants ON p1.idPersonne=Occupants.idPersonne  
JOIN Immeubles ON Occupants.idImmeuble=Immeubles.idImmeuble  
JOIN Personnes AS p2 ON Immeubles.idPersonne=P2.idPersonne  
WHERE p2.Nom='Ross'
```

Bases de données

14. Qui habite, dans un immeuble de plus de 10 étages, un appartement de plus de 100 m2.

```
SELECT Personnes.Nom
FROM Immeubles
JOIN Appartements ON Immeubles.idImmeuble=Appartements.idImmeuble
JOIN Occupants ON Appartements.idAppartement=Occupants.idAppartement
JOIN Personnes ON Occupants.idPersonne=Personnes.idPersonne
WHERE NbEtages>10 AND Superficie>100
```

15. Couples de personnes habitant, dans le même immeuble, un appartement de même superficie.

```
SELECT p1.Nom,p2.Nom
FROM Personnes AS p1
JOIN Occupants AS o1 ON p1.idPersonne=o1.idPersonne
JOIN Appartements AS a1 ON a1.idAppartement=o1.idAppartement
JOIN Appartements AS a2 ON a2.idImmeuble=o1.idImmeuble
JOIN Occupants AS o2 ON a2.idImmeuble=o2.idImmeuble
JOIN Personnes AS p2 ON o2.idPersonne=p2.idPersonne
WHERE a1.Superficie=a2.Superficie
      AND p1.idPersonne<>p2.idPersonne
```

16. Qui n'habite pas un appartement géré par Ross ?

```
SELECT p1.Personnes
FROM Personnes AS p1
JOIN Occupants ON p1.idPersonne=Occupants.idPersonne
JOIN Immeubles ON Occupants.idImmeuble=Immeubles.idImmeuble
JOIN Personnes AS p2 ON Immeubles.idPersonne=p2.idPersonne
WHERE p2.Nom<>'Ross'
```

17. Qui n'habite pas un appartement qu'il gère lui-même ?

```
SELECT p1.Personnes
FROM Personnes AS p1
JOIN Occupants ON p1.idPersonne=Occupants.idPersonne
JOIN Immeubles ON Occupants.idImmeuble=Immeubles.idImmeuble
JOIN Personnes AS p2 ON Immeubles.idPersonne=p2.idPersonne
WHERE p2.idPersonne<>p1.idPersonne
```

18. Quels sont les immeubles où personne n'a emménagé en 1996 ?

```
SELECT Nom
FROM Immeubles
WHERE idImmeuble NOT IN
(
    SELECT DISTINCT idImmeuble
    FROM Occupants
    WHERE AnneeArrivee=1996
)
```

19. Quels sont les immeubles où tout le monde a emménagé en 1994 ?

```
SELECT Nom
FROM Immeubles
WHERE idImmeuble NOT IN
(
    SELECT DISTINCT idImmeuble
    FROM Occupants
    WHERE AnneeArrivee<>1994
)
```

Exercice 2

Voici les tables qui composent la base de données **Médiathèque** :

- **Acteur** (CodeActeur, NomActeur, PrenomActeur)
- **Emprunt** (#CodeFilm, NumExemplaire, #CodePers, DatePret, DateRetour)
- **Emprunteur** (CodePers, NomPres, PrenomPers, AdrPers, TelPers)
- **Etat** (CodeEtat, DesignationEtat)
- **Exemplaire** (#CodeFilm, NumExemplaire, #CodeEtat, #CodeSupport)
- **Film** (CodeFilm, TitreFilm, Annee, Duree, Resume, #CodeGenre)
- **Genre** (CodeGenre, IntGenre)
- **Jouer_un_role** (#CodeFilm, #CodeActeur)
- **Support** (CodeSupport, IntSupport)

Cette base de données est accessible sur www.sn-i.fr → Rubrique « Ressources pédagogiques ».

Écrivez les requêtes suivantes en SQL.

1. Liste des films et de leurs acteurs (champs à afficher : TitreFilm, NomActeur, PrenomActeur et IntSupport)

```
SELECT TitreFilm, NomActeur, PrenomActeur, IntSupport
FROM Film
JOIN Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
JOIN Film_Support USING(CodeFilm)
JOIN Support USING(CodeSupport)
```

2. Liste des films et de leurs acteurs (champs à afficher : TitreFilm, NomActeur, PrenomActeur, IntSupport et IntGenre)

```
SELECT TitreFilm, NomActeur, PrenomActeur, IntSupport, IntGenre
FROM Film
JOIN Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
JOIN Film_Support USING(CodeFilm)
JOIN Support USING(CodeSupport)
JOIN Genre USING(CodeGenre)
```

3. Liste des films prêtés (sortis de la vidéothèque) (champ à afficher : TitreFilm)

```
SELECT TitreFilm
FROM Emprunt
JOIN Film USING(CodeFilm)
WHERE DateRetour IS NULL
```

4. Liste des films dans lesquels joue Sylvester Stallone (champs à afficher : TitreFilm, Annee, Duree et IntGenre)

```
SELECT TitreFilm, Annee, Duree, IntGenre
FROM Film
JOIN Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
JOIN Genre USING(Codegenre)
WHERE NomActeur='Stallone' AND PrenomActeur='Sylvester'
```

5. Liste des films dans lesquels jouent Gérard Depardieu et Christian Clavier (champs à afficher : TitreFilm, Annee, Duree et IntGenre)

```
SELECT TitreFilm, Annee, Duree, IntGenre
FROM (
SELECT Film.*
FROM Film
JOIN Jouer_un_role USING(CodeFilm)
```

Bases de données

```
JOIN Acteur USING(CodeActeur)
WHERE NomActeur='Clavier' AND PrenomActeur='Christian'
) A Film_Clavier
JOIN Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
JOIN Genre USING(Codegenre)
WHERE NomActeur='Depardieu' AND PrenomActeur='Gérard'
```

6. Liste des films dans lesquels joue Gérard Depardieu ou Christian Clavier (champs à afficher : TitreFilm, Année, Durée et IntGenre)

```
SELECT TitreFilm,Année,Durée,IntGenre
FROM Film
JOIN Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
JOIN Genre USING(Codegenre)
WHERE (NomActeur='Depardieu' AND PrenomActeur='Gérard')
OR (NomActeur='Clavier' AND PrenomActeur='Christian')
```

7. Liste des personnes ayant déjà au moins une fois emprunté « Rambo », « Astérix et Obélix contre César » ou « The Faculty » (champs à afficher : NomPers, PrenomPers, AdrPers et TelPers)

```
SELECT DISTINCT NomPers,PrenomPers,AdressePers,TelPers
FROM Films
JOIN Emprunt USING(CodeFilm)
JOIN Emprunteur USING(CodePers)
WHERE TitreFilm='Rambo' OR TitreFilm='Asterix' OR TitreFilm='The Faculty'
```

8. Liste des films qui ont déjà été empruntés par tous les emprunteurs (champ à afficher : TitreFilm)

```
SELECT TitreFilm
FROM Emprunt
JOIN Films USING(CodeFilm)
GROUP BY TitreFilm
HAVING COUNT(DISTINCT CodePers)=(
SELECT COUNT(*) FROM Emprunteur
)
```

9. Liste des films qui n'ont jamais été empruntés (champ à afficher : TitreFilm)

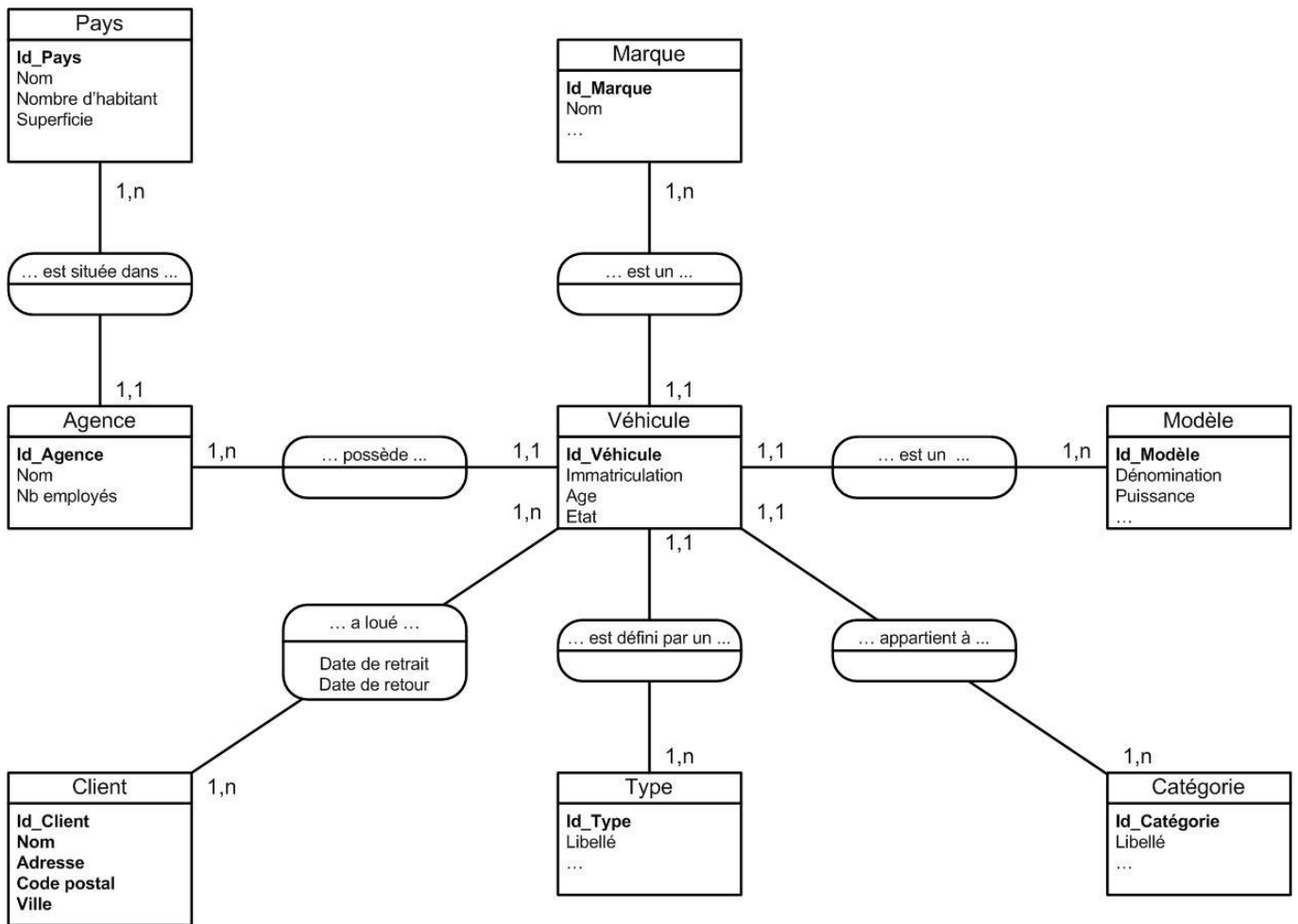
```
SELECT TitreFilm
FROM Film
WHERE CodeFilm NOT IN (
SELECT DISTINCT CodeFilm FROM Emprunt
)
```

10. Liste de tous les acteurs et des codes films dans lesquels ils jouent (champ à afficher : NomActeur, PrenomActeur et CodeFilm)

```
SELECT CodeFilm,NomActeur,PrenomActeur
FROM Jouer_un_role USING(CodeFilm)
JOIN Acteur USING(CodeActeur)
```

Exercice 3

Voici le MCD du logiciel d'une agence de location de véhicules.



1. Produisez le MLD.

Pays(*Id_Pays*, *Nom*, *Nombre_habitants*, *Superficie*)
Agence(*Id_Agence*, *Nom*, *Nb_employés*, #*Id_Pays*)
Marque(*Id_Marque*, *Nom*)
Type(*Id_Type*, *Libellé*)
Catégorie(*Id_Catégorie*, *Libellé*)
Modèle(*Id_Modèle*, *Dénomination*, *Puissance*)
Client(*Id_Client*, *Nom*, *Adresse*, *Code_postal*, *Ville*)
A_loué(#*Id_Client*, #*Id_Vehicule*, *Date_de_retrait*, *Date_de_retour*)
Véhicule(*Id_Véhicule*, *Immatriculation*, *Age*, *Etat*, #*Id_Marque*, #*Id_Modèle*, #*Id_Agence*, #*Id_Catégorie*, #*Id_Type*)

2. En supposant que le MLD est implémenté sur une base de données SQL, réalisez les requêtes suivantes :

- Obtenir la liste des véhicules empruntés et rendu le même jour ainsi que l'agence de rattachement

```

SELECT Agence.Nom, Immatriculation
FROM Agence
JOIN Véhicule
USING (Id_Agence)
JOIN A_loué
USING (Id_Véhicule)
WHERE Date_de_retrait=Date_de_retour
    
```

Bases de données

- Obtenir le nombre véhicules pour chaque marque

```
SELECT Marque.Nom, COUNT(*)
FROM Véhicule
JOIN Marque
USING (Id_Marque)
GROUP BY Marque.Nom
```

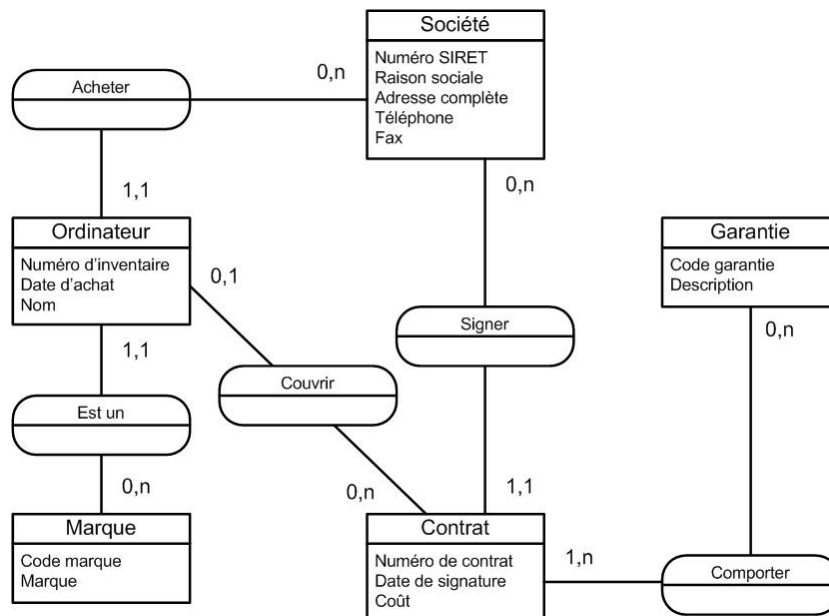
- Obtenir les noms des clients qui ont loué plus de 10 véhicules de marque « Renault »

```
SELECT Client.Nom
FROM Client
JOIN A_loué
USING (Id_Client)
JOIN Véhicule
USING (Id_Véhicule)
JOIN Marque
USING (Id_Marque)
WHERE Marque.Nom="Renault"
GROUP BY Client.Nom
HAVING COUNT(Id_Véhicule)>10
```

- Obtenir le nombre d'agences et d'employés par pays.

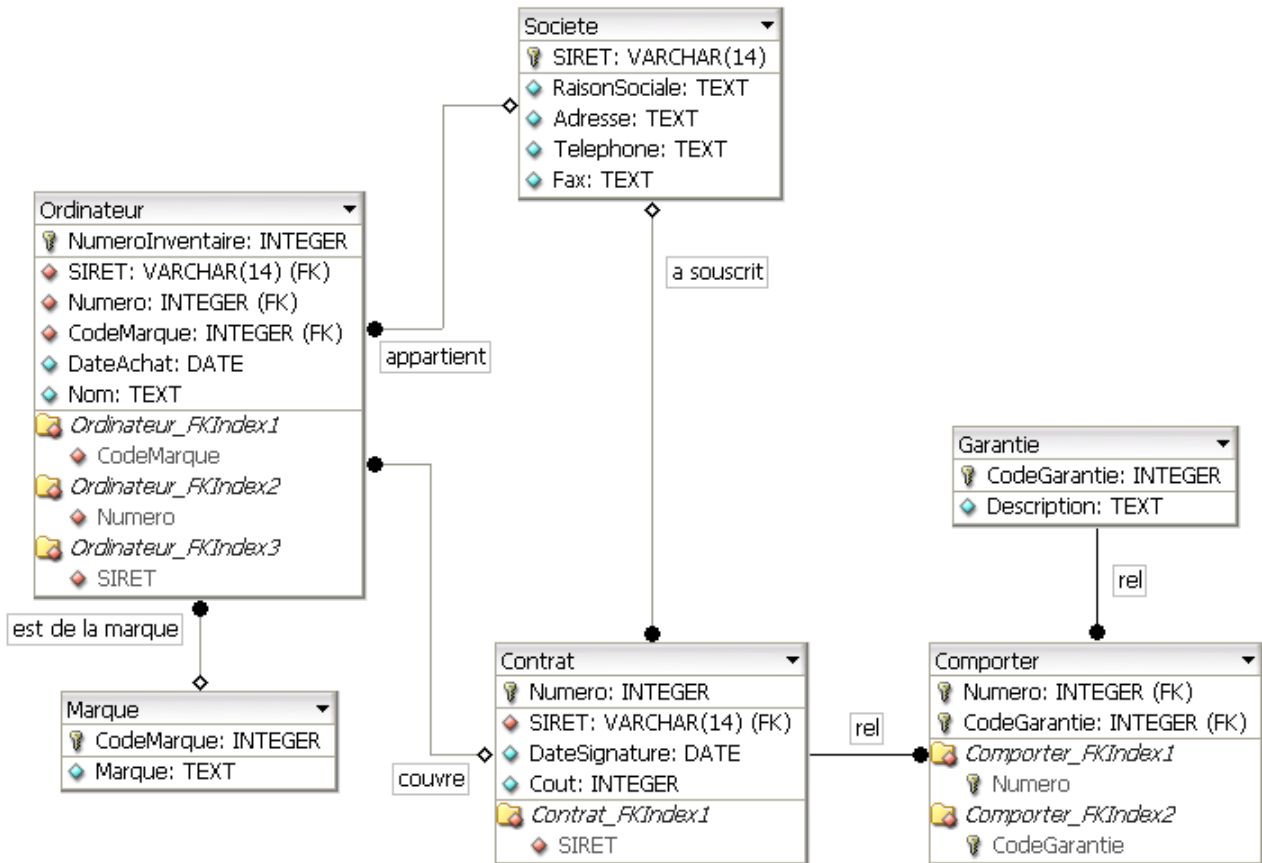
```
SELECT COUNT(Id_Agence), SUM(Nb_employés), Pays.Nom
FROM Agence
JOIN Pays
USING (Id_Pays)
GROUP BY Pays.Nom
```

Exercice 4



Bases de données

1. Passez ce MCD en MLD.



2. Ecrivez en SQL les requêtes suivantes.

a. Obtenir la liste des contrats rattachés à la société « sni ».

```
SELECT Numero FROM Contrat
JOIN Societe ON Contrat.SIRET = Societe.SIRET
WHERE RaisonSociale = 'sni'
```

b. Obtenir pour chaque client le montant global de ses contrats ainsi que le coût moyen d'un contrat.

```
SELECT SUM(Cout), AVG(Cout), RaisonSociale, Contrat.SIRET FROM Contrat
JOIN Societe ON Contrat.SIRET = Societe.SIRET
GROUP BY SIRET, RaisonSociale
```

c. Obtenir la liste des ordinateurs de marque « Dell » couverts par un contrat de la société « sni ».

```
SELECT * FROM Ordinateur AS O
JOIN Contrat AS C USING(Numero)
JOIN Societe AS S ON S.SIRET = C.SIRET
JOIN Marque AS M ON M.CodeMarque = O.CodeMarque
WHERE S.RaisonSociale = 'sni'
AND M.Marque = 'Dell'
```

d. Obtenir la liste des contrats qui ne couvrent que des ordinateurs de la marque « Dell ».

```
SELECT * FROM Contrat AS C
WHERE C.Numero NOT IN
(SELECT C.Numero FROM Contrat
JOIN Ordinateur AS O USING(Numero)
JOIN Marque AS M USING(CodeMarque)
WHERE M.Marque <> 'Dell')
```


Bases de données

e. Obtenir la liste des ordinateurs garantis contre les incendies et les inondations.

```
SELECT O.NumeroInventaire FROM Ordinateur AS O
JOIN Contrat AS C USING(Numero)
JOIN Comporter AS Co USING(Numero)
JOIN Garantie As G USING(CodeGarantie)
WHERE G.Description = 'incendies'
INTERSECT
SELECT O.NumeroInventaire FROM Ordinateur AS O
JOIN Contrat AS C USING(Numero)
JOIN Comporter AS Co USING(Numero)
JOIN Garantie As G USING(CodeGarantie)
WHERE G.Description = 'inondations'
```

f. Obtenir le nombre d'ordinateurs achetés chaque jour du mois de novembre 2006.

```
SELECT count(NumeroInventaire), dateAchat FROM Ordinateur
WHERE dateAchat >= '2006-11-01' AND dateAchat <= '2006-11-30'
GROUP BY dateAchat
```

g. Obtenir le nombre de contrats offrant des garanties sur des ordinateurs vieux de plus de 2 ans.

```
SELECT COUNT(Contrat.Numero) AS nb FROM Contrat
JOIN Ordinateur USING (Numero)
WHERE DateAchat<CURRENT_DATE-365*2
```

h. Obtenir la liste des contrats couvrant les ordinateurs de la société « sni ».

```
SELECT Numero,DateSignature,Cout
FROM Contrat
JOIN Ordinateur USING(Numero)
JOIN Societe USING(SIRET)
WHERE RaisonSociale='sni'
```

Exercice 5

Soit le modèle relationnel suivant :

Salaries(idSalarie,Nom,Prenom,Salaire,#idSalarieChef)

Seuls les dirigeants n'ont pas de chef (champ idSalarieChef qui vaut NULL).

1. Ecrire une requête SQL qui permet d'afficher la liste des salariés avec le nom de leur chef si possible.

```
SELECT S1.Nom,S1.Prenom,S2.Nom AS NomChef,S2.Prenom AS PrenomChef
FROM Salaries AS S1
JOIN Salaries AS S2 ON S1.idSalarieChef=S2.idSalarie
```

2. Ecrire une requête SQL qui permet d'afficher la liste des chefs avec le nombre de salariés encadrés.

```
SELECT S2.idSalarie,S2.Nom,S2.Prenom,COUNT(*) AS NbSalaries
FROM Salaries AS S1
JOIN Salaries AS S2 ON S1.idSalarieChef=S2.idSalarie
GROUP By S2.idSalarie,S2.Nom,S2.Prenom
```

Exercice 6

Soit le modèle relationnel suivant relatif à la gestion des notes annuelles d'une promotion d'étudiants :

- **Etudiant** (idEtudiant, Nom, Prenom)
- **Matiere** (idMatiere, LibelleMat, CoeffMat)
- **Evaluer** (#idEtudiant, #idMatiere, Date, Note)

Réalisez les requêtes SQL suivantes...

1. Quel est le nombre total d'étudiants ?

```
SELECT COUNT(*) FROM Etudiant
```

2. Quelles sont, parmi l'ensemble des notes, la note la plus haute et la note la plus basse ?

```
SELECT MIN(Note),MAX(Note)  
FROM Evaluer
```

3. Quelles sont les moyennes de chaque étudiant dans chacune des matières ?

```
SELECT Nom,Prenom,LibelleMat,AVG(Note)  
FROM Etudiant  
JOIN Evaluer USING(N°Etudiant)  
JOIN Matiere USING(CodeMat)  
GROUP BY N°Etudiant,Nom,Prenom,CodeMat,LibelleMat
```

4. Quelles sont les moyennes par matière ?

```
SELECT LibelleMat,AVG(Note)  
FROM Evaluer  
JOIN Matiere USING(CodeMat)  
GROUP BY CodeMat,LibelleMat
```

5. Quelle est la moyenne générale de chaque étudiant ?

```
SELECT Nom,Prenom,SUM(NoteC)/(SELECT SUM(CoeffMat) FROM Matiere)  
FROM (  
SELECT Nom,Prenom,CoeffMat*AVG(Note) AS NoteC  
FROM Etudiant  
JOIN Evaluer USING(N°Etudiant)  
JOIN Matiere USING(CodeMat)  
GROUP BY N°Etudiant,Nom,Prenom,CodeMat,LibelleMat,CoeffMat  
) AS n  
GROUP BY Nom,Prenom
```

6. Quels sont les étudiants qui ont une moyenne générale supérieure ou égale à la moyenne générale de la promotion ?

```
SELECT Nom,Prenom  
FROM (  
SELECT Nom,Prenom,SUM(NoteC)/(SELECT SUM(CoeffMat) FROM Matiere) AS Moy  
FROM (  
SELECT Nom,Prenom,CoeffMat*AVG(Note) AS NoteC  
FROM Etudiant  
JOIN Evaluer USING(N°Etudiant)  
JOIN Matiere USING(CodeMat)  
GROUP BY N°Etudiant,Nom,Prenom,CodeMat,LibelleMat,CoeffMat  
) AS n  
GROUP BY Nom,Prenom  
)  
WHERE Moy > (  
SELECT AVG(Moy)  
FROM (
```

Bases de données

```
SELECT Nom,Prenom,SUM(NoteC)/(SELECT SUM(CoefMat) FROM Matiere) AS Moy
FROM (
SELECT Nom,Prenom,CoefMat*AVG(Note) AS NoteC
FROM Etudiant
JOIN Evaluer USING(N°Etudiant)
JOIN Matiere USING(CodeMat)
GROUP BY N°Etudiant,Nom,Prenom,CodeMat,LibelleMat,CoefMat
) AS n
GROUP BY Nom,Prenom
```

)
)