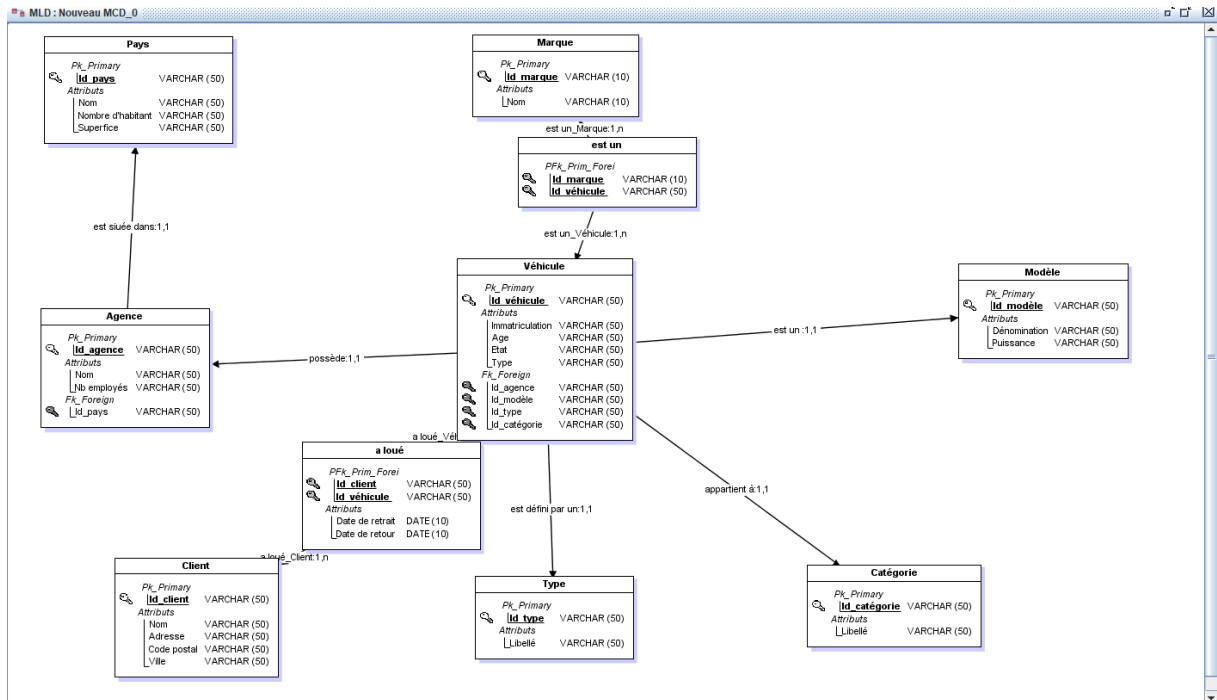
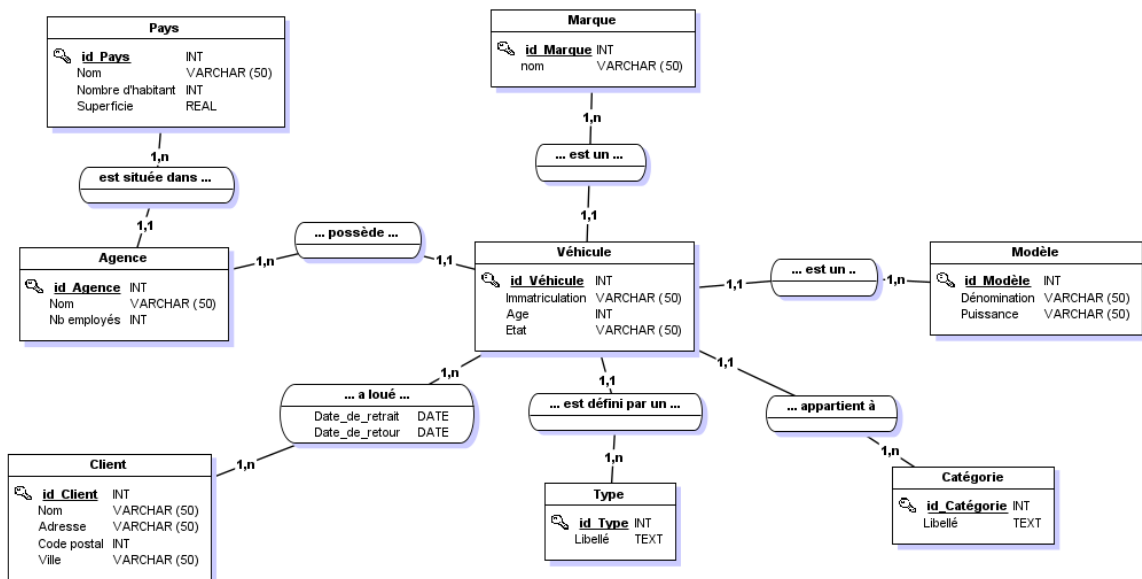


Questionnaire PPE TP 3

MLD :



MCD :



Traduction du MLD Graphique en une représentation textuelle simplifiée d'une base de données :

- Véhicule (**id_Vehicule**, immatriculation, age, etat, #id_Agence, #id_Marque, #id_Type, #id_categorie, #id_modele)
- Pays (**id_pays**, nom, nombre_habitant, superficie)
- Agence (**id_Agence**, nom, nb_employes, #id_pays)
- a_loue (**id_client**, **id_vehicule**, date_de_retrait, date_de_retour)
- Client (**id_client**, nom, adresse, code_postal, ville)
- Marque (**id_marque**, nom)
- Type (**id_type**, libelle)
- Catégorie (**id_categorie**, libelle)
- Modèle (**id_modele**, denomination, puissance)

a. Afficher toutes les informations sur les véhicules

loués par le Client n° T122

```
SELECT c . *  
FROM `a_loue` al  
INNER JOIN client c ON al.id_client = c.id_client  
WHERE al.Id_client =5  
LIMIT 0 , 30
```

b. Afficher toutes les locations réalisées par le client n° T122

```
SELECT c . *  
FROM `a_loue` al  
INNER JOIN client c ON al.id_client = c.id_client  
WHERE al.id_vehicule =5  
LIMIT 0 , 30
```

c. Afficher l'immatriculation, l'âge et l'état de tous les véhicules.

```
SELECT `Immatriculation` , `Age` , `Etat`  
FROM `vehicule`  
WHERE 1  
LIMIT 0 , 30
```

d. Afficher les noms des clients et les adresses, des clients qui habitent à << Nice >>.

```
SELECT `Nom` , `Adresse`  
FROM `client`  
WHERE `Ville` = 'Nice'  
LIMIT 0 , 30
```

e. Affiche la liste des clients par ordre alphabétique croissant des noms

```
SELECT *
FROM `client`
ORDER BY `client`.`Nom` ASC
LIMIT 0 , 30
```

f. Ajouter l'attribut kilométrage et Afficher la liste des voitures par ordre décroissant des compteurs (kilométrage)

```
SELECT *
FROM `vehicule`
ORDER BY `kilometre` DESC
LIMIT 0 , 30
```

g. Afficher les informations sur les clients qui ont loué la voiture EW 2EW

```
SELECT *
FROM `client` c
INNER JOIN a_loue al ON c.id_client = al.id_client
INNER JOIN vehicule v ON al.id_vehicule = v.id_vehicule
WHERE v.immatriculation = 'AAA-111'
LIMIT 0 , 30
```

h. Afficher toutes les voitures noires :)

```
SELECT *
FROM `vehicule`
WHERE `couleur` = 'noir'
LIMIT 0 , 30
```

i. Afficher toutes les voitures ayant un kilométrage <10000 km

```
SELECT *
FROM `vehicule`
WHERE `kilometre` <10000
LIMIT 0 , 30
```

j. Afficher toutes les informations sur les locations réalisées avant 2018

```
SELECT *
FROM `a_loue`
WHERE `Date_de_retrait` >01 /01 /2018 ;
LIMIT 0 , 30
```

k. Afficher la moyenne des kilométrages de tous les véhicules du parc.

```
SELECT AVG( `kilometre` )  
FROM `vehicule`  
WHERE 1
```

L. Afficher toutes les locations réalisées en 2018

```
SELECT *  
FROM `a_loue`  
WHERE `Date_de_retrait`  
BETWEEN '01-01-2018'  
AND '31-12-2018'  
LIMIT 0 , 30
```

M. Afficher le nombre de voitures ayant un kilométrage <10 000 kilomètres

```
SELECT COUNT(*)  
FROM `vehicule`  
WHERE `kilometre` <1000  
LIMIT 0 , 30
```

Partie 2 :

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

```
SELECT `v.id_vehicule`, `al.date_de_retrait`, `a.nom`, `al.date_de_retour`  
FROM `vehicule`  
INNER JOIN agence a ON a.id_agence = v.id_agence  
INNER JOIN a_loue al ON v.id_vehicule = al.id_vehicule  
WHERE al.date_de_retrait = al.date_de_retour
```

- Obtenir le nombre véhicules pour chaque marque

```
SELECT `m.nom`, COUNT(*)  
FROM `vehicule`  
INNER JOIN marque m ON v.id_marque = m.id_marque  
GROUP BY m.nom
```

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

```
SELECT `nom`, COUNT( * )
FROM `vehicule`
INNER JOIN marque m ON v.id_marque = m.id_marque
INNER JOIN a_loue al ON v.id_vehicule = al.id_vehicule
INNER JOIN client c ON al.id_client = c.id_client
WHERE m.nom = "Audi"
GROUP BY m.nom
HAVING COUNT( * ) > 10
```

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

```
SELECT `p.nom`, `nombre_d'employes`, COUNT( `a.id_agence` ) AS "nombre_d'agence"
FROM `agence`
INNER JOIN pays p ON a.id_pays = p.id_pays
GROUP BY p.nom
```

Exercice 2

1. Les informations relatives aux étudiants (Code, Nom et Date de naissance) selon l'ordre alphabétique croissant du nom

```
SELECT `NomEt`, `DatnEt`, `CodeEt`
FROM `ETUDIANT`
ORDER BY `NomEt` ASC;
```

2. Les noms et les grades des enseignants de la matière dont le nom est 'BD'.

```
SELECT `NomEns`, `GradeEns`
FROM `ENSEIGNANT`
WHERE CodeMat = (
SELECT `CodeMat`
```

```
FROM `MATIERE`  
WHERE NomMat = 'BD');
```

3. La liste distincte formée des noms et les coefficients des différentes matières qui sont enseignées par des enseignants de grade 'Grd3'.

```
SELECT DISTINCT `NomMat`, `CoefMat`  
FROM `MATIERE`  
WHERE CodeMat  
IN (  
SELECT `CodeMat`  
FROM `ENSEIGNANT`  
WHERE GradeEns = 'Grd3');
```

4. La liste des matières (Nom et Coefficient) qui sont suivies par l'étudiant de code 'Et321'.

```
SELECT `NomMat`, `CoefMat`  
FROM `MATIERE`  
WHERE CodeMat  
IN (  
SELECT `CodeMat`  
FROM `note`  
WHERE CodeEt = 'Et321');
```

5. Le nombre d'enseignants de la matière dont le nom est 'Informatique'

```
SELECT COUNT( * )  
FROM `ENSEIGNANT`  
WHERE CodeMat = (  
SELECT `CodeMat`  
FROM `MATIERE`  
WHERE NomMat = 'informatique' );
```

Exercice 3

1. Quelle est la composition de l'équipe Festina (Numéro, nom et pays des coureurs) ?

```
SELECT `C.NumeroCoureur`, `C.NomCoureur`, `PAYS.NomPays`  
FROM `coureur` C  
INNER JOIN EQUIPE E ON C.CodeEquipe = E.CodeEquipe  
INNER JOIN PAYS ON C.CodePays = PAYS.CodePays  
WHERE E.NomEquipe = 'Festina'
```

2. Quel est le nombre de kilomètres total du Tour de France 97 ?

```
SELECT SUM( NbKm ) AS "Nombre kilometre total"  
FROM etape
```

3. Quel est le nombre de kilomètres total des étapes de type "Haute Montagne" ?

4. Quels sont les noms des coureurs qui n'ont pas obtenu de bonifications ?

```
SELECT `C.NomCoureur`  
FROM `coureur` C  
WHERE `C.NumeroCoureur` NOT  
IN (  
SELECT `a.NumeroCoureur`  
FROM `attribuer_bonification` a )
```

5. Quels sont les noms des coureurs qui ont participé à toutes les étapes ?

```
SELECT `c.NomCoureur`  
FROM `coureur` c  
WHERE NOT  
EXISTS (  
SELECT `e.NumeroEtap`  
FROM `etape` e  
WHERE e.NumeroEtap NOT  
IN (  
SELECT `p.NumeroEtap`  
FROM `participer` p  
WHERE p.NumeroCoureur = c.NumeroCoureur ))
```

6. Quel est le classement général des coureurs (nom, code équipe, code pays et temps des coureurs) à l'issue des 13 premières étapes sachant que les bonifications ont été intégrées dans les temps réalisés à chaque étape ?

7. Quel est le classement par équipe à l'issue des 13 premières étapes (nom et temps des équipes) ?

Exercice 4

1. la liste des clients de Marrakech.

```
SELECT `Nomcli`, `Prenomcli`, `adressecli`  
FROM `Client`  
WHERE adressecli = 'Marrakech'
```

2. la liste des produits (Numprod, désignation, prix) classés de plus cher au moins cher.

```
SELECT `Numprod`, `designation`, `prix`  
FROM `Produit`  
ORDER BY prix DESC
```

3. noms et adresses des vendeurs dont le nom commence par la lettre 'M'.

```
SELECT `Nomvendeur`, `adresse_vend`  
FROM `Vendeur`  
WHERE Nomvendeur LIKE 'M%'
```

4. la liste des commandes effectuées par le vendeur "Mohammed" entre le 1er et 30 janvier 2020.

```
SELECT c.*  
FROM `Commande` c  
INNER JOIN Vendeur v ON c.Idvendeur = v.Idvendeur  
WHERE Nomvendeur = 'Mohammed'  
AND date_com  
BETWEEN '2020-01-01'  
AND '2020-01-30'
```

5. le nombre des commandes contenant le produit n° 365.

```
SELECT COUNT(*) AS "Nombre de commande pour le produit n°365"  
FROM `Commande`  
WHERE Numprod = 365
```

Exercice 5

1. La liste de tous les étudiants.

```
SELECT `nom`, `prenom`  
FROM `etudiant`
```

2. Nom et coefficient des matières.

```
SELECT `nom_matiere`, `coefficient`  
FROM `matiere`
```

3. Les numéros des cartes d'identité des étudiants dont la moyenne entre 7 et 12.

4. La liste des étudiants dont le nom commence par 'ben'.

```
SELECT `nom`, `prenom`, `numero_carte_etudiant`
FROM `ETUDIANT`
WHERE nom LIKE 'ben%'
```

5. Le nombre des étudiants qui ont comme matière '12518'.

```
SELECT COUNT( * ) AS "nombre des étudiants qui ont comme matière 12518"
FROM (
SELECT DISTINCT numero_carte_etudiant
FROM NOTE
WHERE code_matiere = '12518'
) AS students
```

6. La somme des coefficients des matières.

```
SELECT SUM( coefficient ) AS "total coefficient"
FROM `MATIERE`
```

7. Les noms des étudiants qui une note_examen >10.

```
SELECT `nom`, `prenom`
FROM `etudiant` e
INNER JOIN note n ON e.numero_carte_etudiant = n.numero_carte_etudiant
WHERE note_examen >10
```

8- Afficher les noms et les coefficients des matières étudiées par l'étudiant "01234568".

```
SELECT `nom_matiere`, `coefficient`
FROM `matiere` m
INNER JOIN note n ON n.code_matiere = m.code_matiere
INNER JOIN etudiant e ON n.numero_carte_etudiant = e.numero_carte_etudiant
WHERE e.numero_carte_etudiant = '01234568'
```