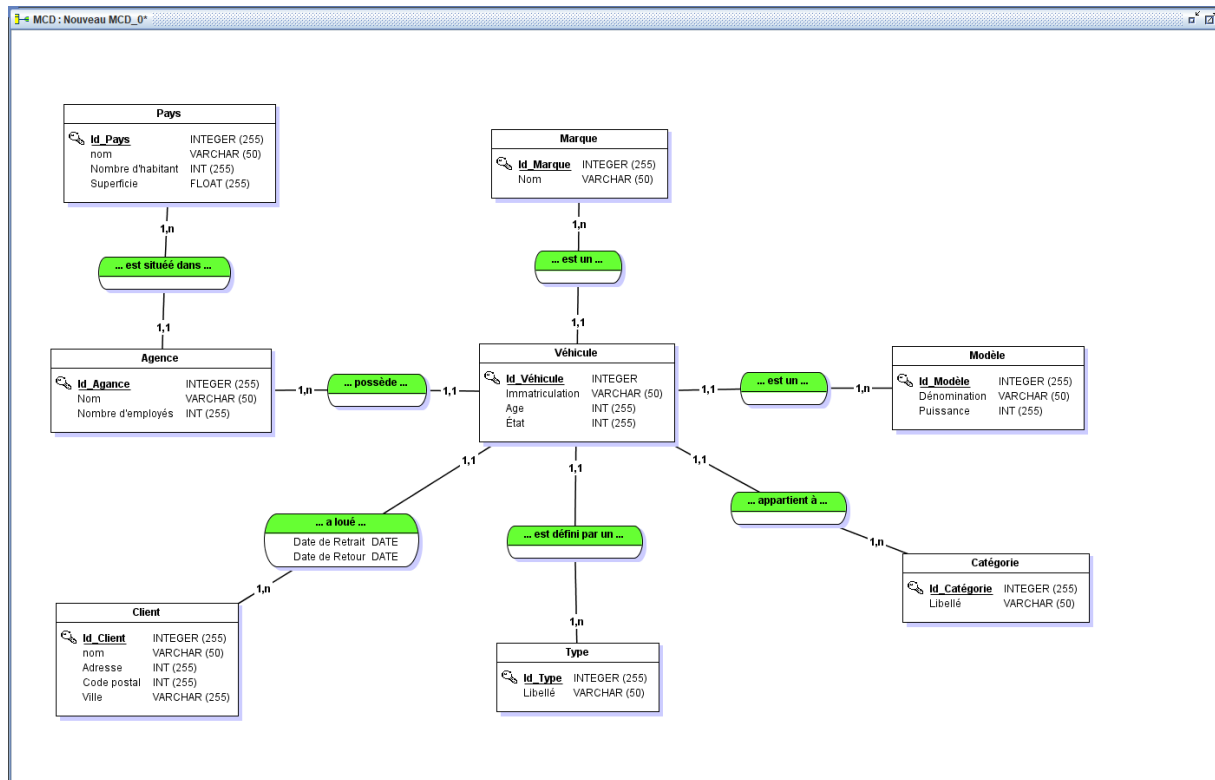


## MISSION 3



### Exercice 1 :

- a. `SELECT * FROM véhicule BY client n°T122 ;`
- b. `SELECT * FROM locations BY n°T122 ;`
- c. `SELECT Immatriculation, Age, État FROM véhicule ;`
- d. `SELECT nom, Adresse FROM client WHERE 'Nice' ;`
- e. `SELECT nom FROM Client ORDER BY ASC ;`
- f. `ALTER TABLE voiture`  
`ADD COLUMN kilometrage INTERGER`  
`SELECT *`  
`FROM véhicule`  
`ORDER BY DESC ;`
- g. `SELECT nom, Adresse Code postal, Ville FROM Client WERE véhicule = 'EW 25' ;`
- h. `SELECT * FROM véhicule WHERE couleur 'noir' ;`
- i. `SELECT * FROM véhicule WHERE kilometrage < 10 000 ;`
- j. `SELECT * FROM locations WHERE YEAR (< 2018) ;`
- k. `SELECT AVG(kilometrage) as moyenne_kilometrage FROM voiture ;`
- l. `SELECT * FROM locations WHERE YEAR (2018) ;`
- m. `SELECT COUNT(*) FROM véhicule WHERE kilometrage < 10 000 ;`

`SELECT emprunter et rendu FROM véhicule WHERE DATE(day) ;`

`SELECT marque count(*) FROM véhicule GROUP BY marque ;`

SELECT nom FROM Client HAVING COUNT(véhicule) > 10 ;

SELECT agent, employé FROM pays ;

Exercice 2 :

- a. SELECT \* FROM étudiant ORDER BY NomEt ASC;
- b. SELECT NomEns, GradEns FROM enseignant INNER JOIN matiere on CodeEns = CodeMat WHERE NomMat = 'BD' ;
- c. SELECT CoefMat, NomMat FROM matiere INNER JOIN enseignant ON CodeMat = CodeEns WHERE GradeEns = 'Grd3' ;
- d. SELECT NomMat, CoefMat FROM matiere INNER JOIN enseignant ON CodeMat = CodeEns INNER JOIN enseignant ON CodeEns = CodeEt WHERE CodeEt = 321 ;
- e. SELECT COUNT(NomEns) From matiere INNER JOIN enseignant ON CodeEns = CodeMat WHERE NomMat = 'Informatique' ;

Exercice 3 :

a.

SELECT NumCoureur, NomCoureur, CodePays FROM Coureur INNER JOIN equipe on NumCoureur = CodeEquipe WHERE NomEquipe = 'Festina' ;

b.

SELECT COUNT(km) FROM Attribuer\_Bonification ;

c.

```
SELECT SUM(NbKm) AS TotalKmHauteMontagne FROM ETAPE WHERE CodeType = ( SELECT CodeType FROM TYPE_ETAPE WHERE LibelleType = 'Haute Montagne' );
```

d.

```
SELECT DISTINCT NomCoureur FROM COUREUR WHERE NuméroCoureur NOT IN ( SELECT DISTINCT NuméroCoureur FROM ATTRIBUER_BONIFICATION );
```

e.

```
SELECT NomCoureur FROM COUREUR WHERE NOT EXISTS ( SELECT etape.NuméroEtap FROM ETAPE WHERE NOT EXISTS ( SELECT * FROM PARTICIPER WHERE PARTICIPER.NuméroEtap = ETAPE.NuméroEtap AND PARTICIPER.NuméroCoureur = COUREUR.NuméroCoureur ) );
```

f.

```
SELECT COUREUR.NomCoureur, EQUIPE.CodeEquipe, COUREUR.CodePays, SUM(TIME_TO_SEC(PARTICIPER.TempsRealisé) + ATTRIBUER_BONIFICATION.NbSecondes) AS TempsTotal FROM COUREUR JOIN PARTICIPER ON COUREUR.NuméroCoureur = PARTICIPER.NuméroCoureur JOIN ETAPE ON PARTICIPER.NuméroEtap = ETAPE.NuméroEtap JOIN EQUIPE ON COUREUR.CodeEquipe = EQUIPE.CodeEquipe LEFT JOIN ATTRIBUER_BONIFICATION ON PARTICIPER.NuméroEtap = ATTRIBUER_BONIFICATION.NuméroEtap AND PARTICIPER.NuméroCoureur = ATTRIBUER_BONIFICATION.NuméroCoureur WHERE ETAPE.NuméroEtap <= 13 GROUP BY COUREUR.NuméroCoureur ORDER BY TempsTotal;
```

g.

```
SELECT EQUIPE.NomEquipe, SUM(TIME_TO_SEC(PARTICIPER.TempsRealisé) + ATTRIBUER_BONIFICATION.NbSecondes) AS TempsTotal
FROM EQUIPE JOIN COUREUR ON EQUIPE.CodeEquipe = COUREUR.CodeEquipe JOIN PARTICIPER ON COUREUR.NuméroCoureur =
PARTICIPER.NuméroCoureur JOIN ETAPE ON PARTICIPER.NuméroEtape = ETAPE.NuméroEtap LEFT JOIN ATTRIBUER_BONIFICATION ON
PARTICIPER.NuméroEtape = ATTRIBUER_BONIFICATION.NuméroEtape AND PARTICIPER.NuméroCoureur =
ATTRIBUER_BONIFICATION.NuméroCoureur WHERE ETAPE.NuméroEtap <= 13 GROUP BY EQUIPE.CodeEquipe ORDER BY TempsTotal;
```

Exercice 4 :

a.

SELECT \* FROM Clients WHERE 'Marrakech';

b.

SELECT Numprod, désignation, prix FROM Produit ORDER BY prix DESC;

c.

SELECT NomVendeur, AdresseVendeur FROM vendeur WHERE letter 'M';

d.

SELECT \* FROM Commande INNER JOIN Venduer ON Commande.IdVendeur = Vendeur.IdVendeur  
WHERE NomVendeur = 'Mohammed' AND date\_com BETWEEN '2020-01-01' AND '2020-01-30' ;

e.

SELECT COUNT(\*) FROM CommandeWhere NumProd = 365;

Exercice 5:

a.

SELECT \* FROM Etudiant;

b.

SELECT nom\_matiere, coefficient FROM matiere;

c.

SELECT Numéro\_carte\_étudiant FROM Note GROUP BY Numéro\_carte\_étudiant HAVING  
AVG(note)\_exam) >= 7 AND AVG(note-examen) <= 12;

d.

SELECT \* FROM Etudiant WHERE Nom LIKE 'ben';

e.

SELECT COUNT(DISTINCT numéro\_carte\_étudiant) FROM Note WHERE code\_matière = 12518 ;

f.

SELECT SUM(Coefficient) FROM matiere ;

g.

```
SELECT Nom, Prénom FROM Etudiant Where numéro_carte_étudiant IN ( SELECT  
numéro_carte_étudiant FROM Note WHERE note_examen > 10 ;
```

h.

```
SELECT Nommatiere, coefficient FROM matiere INNER JOIN Note ON Matiere.code_matière =  
Note.code_matière WHERE Note.numéro_carte_étudiant = 01234568 ;
```