

Correction

Rédiger les requêtes SQL qui permettent d'obtenir les résultats suivants :

1- Nom et prénom des étudiants habitant Paris

```
SELECT Nom, Prenom
FROM ETUDIANT
WHERE Loc='PARIS';
```

2- Numéro et libellé des entreprises ayant accueilli des stagiaires

```
SELECT DISTINCT Noentr, Libelle
FROM AFFECTATION JOIN ENTREPRISE
ON Noetab=Noentr;
```

3- Nom des maîtres de stage

```
SELECT DISTINCT Maitre
FROM AFFECTATION;
```

4- Nom et adresse de l'entreprise dans laquelle travaille Mr MORICET

```
SELECT DISTINCT Noentr, Libelle, Ville, Dpt
FROM ENTREPRISE JOIN AFFECTATION
ON Noentr = Noetab
WHERE Maitre = 'MORICET';
```

5- Nom des étudiants qui ont une entreprise dans leur ville de résidence principale.

```
SELECT DISTINCT Nom
FROM ETUDIANT JOIN ENTREPRISE
ON Loc = Ville;
```

6- Dates de début et de fin de stage qui se déroulent en 2022

```
SELECT Datdeb, Datfin
FROM STAGE
WHERE YEAR(Datdeb)=2022;
```

7- Nom des entreprises de la région Aquitaine susceptibles de prendre des stagiaires. Cette liste sera triée par ordre alphabétique des villes où elles se trouvent. Les départements de la région Aquitaine sont : Dordogne(24), Gironde(33), Landes(40), Lot-Et-Garonne(47) et Pyrénées Atlantique (64)

```
SELECT Libelle, Ville
FROM ENTREPRISE
WHERE Dpt IN (24,33,40,47,64)
ORDER BY 2;
```

ou

```
SELECT Libelle, Ville
FROM ENTREPRISE
WHERE Dpt =24 OR Dpt =33 OR Dpt =64 OR Dpt =40 OR Dpt =47
ORDER BY Ville;
```

8- Nom et prénom des étudiants qui habitent la même ville que l'étudiant Bernard LAMOUREUX.

```
SELECT Nom, Prenom
FROM ETUDIANT
WHERE Nom<>'LAMOUREUX'
AND Loc = (SELECT Loc FROM ETUDIANT
            WHERE Nom='LAMOUREUX'
            AND Prenom ='Bernard');
```

Ou avec une auto-jointure

```
SELECT ER.Nom, ER.Prenom
FROM ETUDIANT AS ER JOIN ETUDIANT AS EM
WHERE ER.Nom<>'LAMOUREUX'
And EM.Loc=ER.Loc
And EM.Nom='LAMOUREUX';
```

9- Résidence principale des étudiants ayant effectué un stage

```
SELECT DISTINCT Loc
FROM ETUDIANT E JOIN AFFECTATION A
ON E.Noetud = A.Noetud;
```

10-Nombre d'entreprises commençant par SA

```
SELECT COUNT(*) AS NbEntreprise
FROM ENTREPRISE
WHERE Libelle LIKE 'SA%';
```

11-Nom et localité des étudiants qui ne sont pas joignables par téléphone

```
SELECT Nom, Loc
FROM ETUDIANT
WHERE Tel =''
```

12-Nom des entreprises ayant reçu des stagiaires et implantés dans un département dont le numéro est compris entre 30 et 40.

```
SELECT Libelle
FROM ENTREPRISE
WHERE Noentr IN (SELECT DISTINCT NoEtab
                  FROM AFFECTATION)
AND Dpt BETWEEN 30 AND 40
```

13-Nombre de prénoms différents parmi les étudiants

```
SELECT COUNT(DISTINCT Prenom) AS `Nb prenom`
FROM ETUDIANT;
```

14-Numéro de département et nombre d'entreprises de chaque département.

```
SELECT Dpt, COUNT(Noentr) as nbEntreprise
FROM ENTREPRISE
GROUP BY Dpt
```

15- Numéro, libellé et nombre de stages de chaque entreprise qui a accueilli des stagiaires

```
SELECT noentr, Libelle, COUNT(noetud) as nbStages
FROM entreprise JOIN affectation
ON noetab = noentr
GROUP BY noentr, Libelle
```

16-Le nom des 3 maitres de stage qui ont suivi le plus de stages, ainsi que le nombre de stages suivis.

```
SELECT Maitre, COUNT(*) as nbStages
FROM affectation
GROUP BY Maitre
ORDER BY nbStages DESC
LIMIT 3
```

17-Nombre de stages dans les entreprises situées dans les départements compris entre 70 et 80.

```
SELECT noentr, Libelle, COUNT(noetud) as nbStages
FROM entreprise JOIN affectation
ON noetab = noentr
WHERE Dpt BETWEEN 70 AND 80
GROUP BY noentr, Libelle
```

18-Nom des étudiants qui ont au moins 2 entreprises dans le département de leur résidence principale.

```
SELECT Nom
FROM ETUDIANT
WHERE CP/1000 IN (SELECT Dpt
                  FROM ENTREPRISE
                  GROUP BY Dpt
                  HAVING COUNT(Noentr) >=2);
```

19-Numéro de la période de stage qui a la plus longue durée.

```
SELECT Periode
FROM STAGE
WHERE Datfin - Datdeb = ( SELECT MAX( Datfin – Datdeb )
                        FROM STAGE );
```

20-Numéro du département qui possède le plus d'entreprises d'accueil pour les stagiaires

```
SELECT Dpt
FROM ENTREPRISE
GROUP BY Dpt
HAVING COUNT(*)=( SELECT MAX(nbentreprises) as nbMaxEntreprises
                  FROM (SELECT COUNT(*) as nbentreprises
                        FROM ENTREPRISE
                        GROUP BY Dpt) as TMaxNbentreprises
                  )
```

Etape : Récupération du nombre maximal d'entreprises par département

```
SELECT MAX(nbentreprises) as nbMaxEntreprises
FROM ( SELECT COUNT(*) as nbentreprises
      FROM ENTREPRISE
      GROUP BY Dpt) as TMaxNbentreprises
```