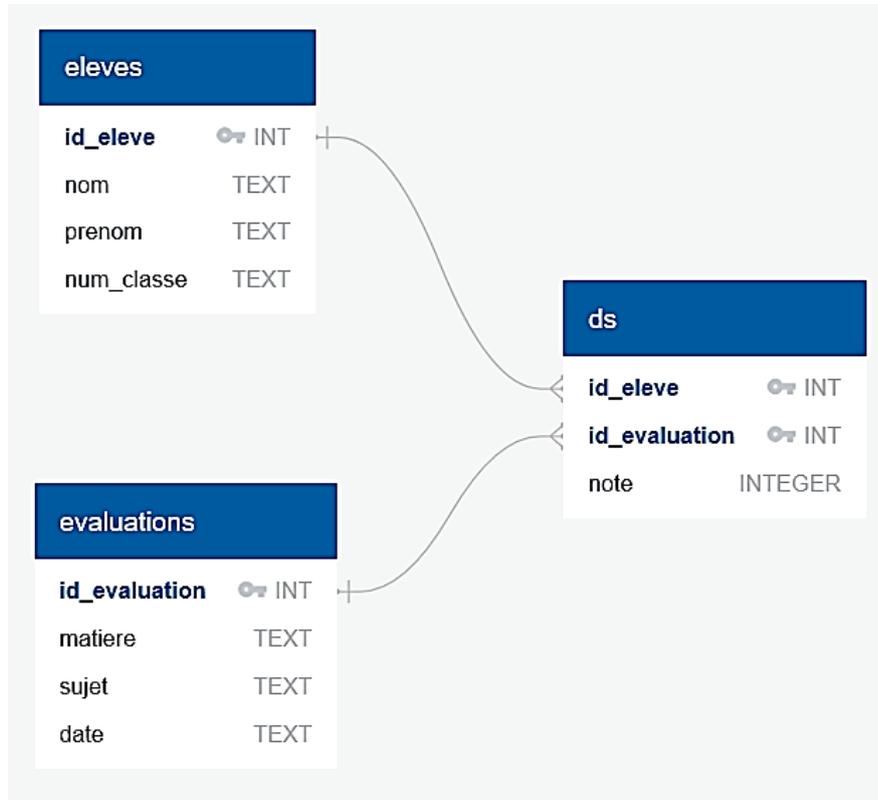


On a constitué une bdd constituée des tables suivantes pour gérer les notes des élèves d'un lycée :



Cette bdd est créée et remplie partiellement dans le fichier *tpNotes.py* à télécharger sur nsibranly.fr et à placer dans votre répertoire de travail.

On demande de compléter ce fichier en créant les fonctions *rechercherNotes()* et *saisirNotes()* décrites ci-dessous :

1- FONCTION RECHERCHENOTES() :

Cette fonction n'a pas de paramètre. A l'exécution de cette fonction, le script demande à l'utilisateur de saisir dans la console le nom d'un élève. Cette fonction recherche alors dans la bdd les notes qui correspondent à cet élève et les affiche ensuite dans la console. L'utilisateur a la possibilité de saisir partiellement l'orthographe du nom. Si cette saisie partielle correspond à plusieurs élèves de la bdd, les notes de chacun d'entre eux sont affichées. On donne ci-dessous 3 exemples d'exécution :

```
>>> rechercheNotes()
Entre un nom d'élèves : anGo
Les notes de ANGO sont :
'maths' : 17/20
'nsi' : 12/20
```

```
>>> rechercheNotes()
Entre un nom d'élèves : ba
Les notes de BALANDIER sont :
'maths' : 9/20
'nsi' : 14/20
Les notes de BAUDIN sont :
'maths' : 19/20
'nsi' : 2/20
```

```
>>> rechercheNotes()
Entre un nom d'élèves : OUSTRIC
Aucune note pour OUSTRIC dans la bdd
```

Le début du code de cette fonction peut être le suivant :

```
def rechercheNotes() :
    saisie = input("Entre un nom d'élèves : ")
    saisie = saisie.upper()
    saisie2 = '%' +saisie+'%'
    # Exécution des requêtes de sélection
```

CORRIGE

```
def rechercheNotes() :
    saisie = input("Entre un nom d'élèves : ")
    saisie = saisie.upper()
    saisie2 = '%' +saisie+'%'
    # Exécution des requêtes de sélection
    curseur.execute("""
        SELECT eleves.nom , evaluations.matiere, ds.note FROM ds
        JOIN eleves ON ds.id_eleve = eleves.id_eleve
        JOIN evaluations ON ds.id_evaluation = evaluations.id_evaluation
        WHERE eleves.nom LIKE ? ;
    """, [saisie2])
    resultat = curseur.fetchall()
    n = len(resultat)
    if n == 0 :
        print(f"Aucune note pour {saisie} dans la bdd")
    else :
        nom = ""
        for e in resultat :
            if e[0] != nom :
                print(f"Les notes de {e[0]} sont :")
                nom = e[0]
            print(f" '{e[1]}' : {e[2]}/20 ")
```

2- FONCTION SAISIRNOTES():

Cette fonction n'a pas de paramètre. A l'exécution de cette fonction, le script demande à l'utilisateur de saisir dans la console la note d'un élève particulier. Cette fonction recherche alors dans la bdd l'existence de cet élève. S'il n'existe pas, le script s'arrête. S'il existe, le script demande à l'utilisateur de saisir l'id de l'évaluation. Si l'id saisi n'existe pas dans la bdd, le script s'arrête. S'il existe, le script demande de saisir la note. On donne ci-dessous 3 exemples d'exécution :

```
>>> saisirNotes()
Entre un nom d'élèves : THETCHAROEN
Entre l'id de l'évaluation : 199
Entre la note : 18
Une note de 18/20 a été insérée pour Matisse THETCHAROEN en eps
```

```
>>> saisirNotes()
Entre un nom d'élèves : BADA
BADA n'est pas inscrit à ce lycée
```

```
>>> saisirNotes()
Entre un nom d'élèves : PICHON
Entre l'id de l'évaluation : 5
5 n'est pas un id enregistré dans la bdd
```

```
def saisirNotes() :
    nom = input("Entrez un nom d'élèves : ")
    curseur.execute("""
        SELECT id_eleve, prenom FROM eleves
        WHERE nom = ? ;
    """, [nom])
    resultat = curseur.fetchall()
    if resultat == [] :
        print(f"{nom} n'est pas inscrit à ce lycée")
    else :
        id_eleve = int(resultat[0][0])
        prenom = resultat[0][1]
        id_evaluation = int(input("Entrez l'id de l'évaluation : "))
        curseur.execute("""
            SELECT matiere FROM evaluations
            WHERE id_evaluation = ? ;
        """, [id_evaluation])
        resultat = curseur.fetchall()
        if resultat == [] :
            print(f"{id_evaluation} n'est pas un id enregistré dans la bdd")
        else :
            matiere = resultat[0][0]
            note = int(input("Entrez la note : "))
            curseur.execute("""
                INSERT INTO ds VALUES (?, ?, ?) """, (id_eleve, id_evaluation, note))
            print(f"Une note de {note}/20 a été insérée pour {prenom} {nom} en {matiere}")
            connexion.commit()
```