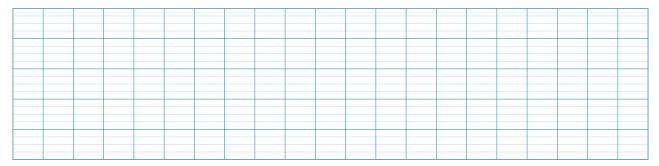
Exercice 1: BDD

Afin de lancer un nouveau service de streaming de musique, vous devez construire une base de données pour les morceaux de votre catalogue. Pour l'instant vous disposez d'une seule table avec les informations des morceaux. Voici **un extrait** de cette table :

Titre	Durée	Artiste	Album	Piste	CD	Année
Astronomy	384	Blue Öyster Cult	Secret Treaties	8	1	1974
Stone Cold Crazy	136	Queen	Sheer Heart attack	8	1	1974
Under Pressure	242	Queen and David Bowie	Hot Space	11	2	1982
The Outlaw Torn	589	Metallica	Load	14	1	1996
Fuel	270	Metallica	Reload	1	1	1997
The Memory Remains	279	Metallica and Marianne Faithfull	Reload	2	1	1997
Astronomy	398	Metallica	Garage Inc.	8	1	1998
Stone Cold Crazy	139	Metallica	Garage Inc.	11	2	1998
Fuel	276	Metallica and the San Francisco Symphony	S&M	6	1	1999
The Outlaw Torn	599	Metallica and the San Francisco Symphony	S&M	6	2	1999

Cette table ne convient pas vraiment pour faire une base de données.

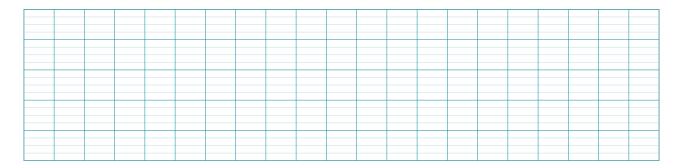
1. Expliquer pourquoi aucune des colonnes ne peut pas servir de clef primaire.



2. Pourquoi est-ce que cette table est problématique si on veut rajouter des informations sur les artistes, comme leur nationalité?



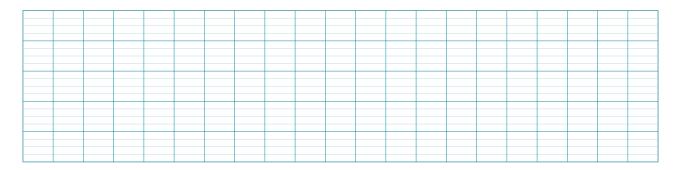
3. Quel est le problème si on souhaite chercher les morceaux d'un artiste? Vous pourrez prendre l'exemple de Metallica.



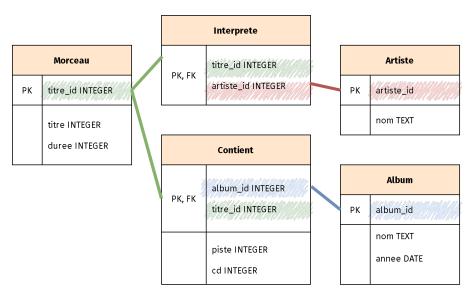
Un ami vous suggère d'utiliser le schéma suivant :

Morceau(<u>titre_id</u>, titre, duree, <u>artiste_id</u>, album, piste, cd, annee) Artiste(<u>artiste_id</u>, nom)

4. Expliquer pourquoi cette représentation ne permet toujours pas de gérer les morceaux fait par deux artistes différents.



Finalement, vous arrivez au schéma suivant :



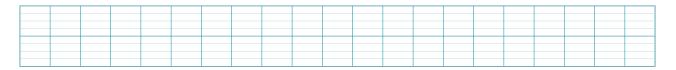
5. Compléter les tables à l'aide des informations déjà disponibles. Un des morceaux n'a pas été intégré, inutile de l'y remettre. Si les noms dépassent, mettre uniquement le début.

titre_id	titre	duree	
519	Astronomy		
1219	Astronomy		
316	Stone Cold Crazy	136	
1319	Stone Cold Crazy	139	
1298		270	
1570			
401			
1125	The Outlaw Torn	589	
599 The Outlaw Torn		599	

titre_id	artiste_id
519	25
1219	154
1319	
1298	154
1570	154
1570	318
1125	154
1591	154
1591	318
316	79
401	79
401	108

nom
Metallica

6. Comment appelle-t-on les clefs primaires de certaines tables apparaissant dans certaines tables, comme dans Interprete?



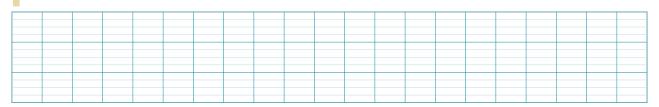
7. Expliquer pourquoi le couple (titre_id, artiste_id) peut servir de clef primaire à Interprete.



8. Traduire en langage naturel les requêtes suivantes :

SQL

SELECT titre, duree FROM Morceau WHERE duree > 600 ORDER BY duree DESC;

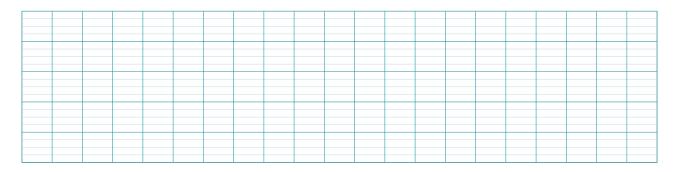


SQL

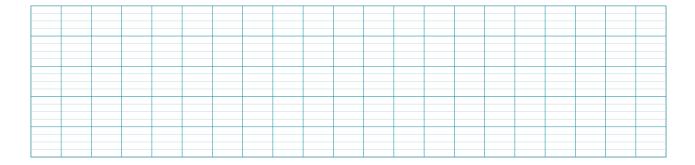
SELECT cd, piste, titre FROM Morceau
JOIN Contient ON Contient.titre_id = Morceau.titre_id
JOIN Album ON Contient.album_id = Album.album_id
WHERE nom = "Garage Inc."
ORDER BY cd, piste;



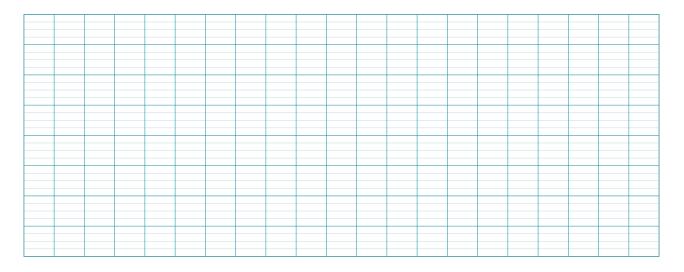
9. Donner la requête SQL permettant d'obtenir le nom de l'artiste dont l'identifiant est 200.



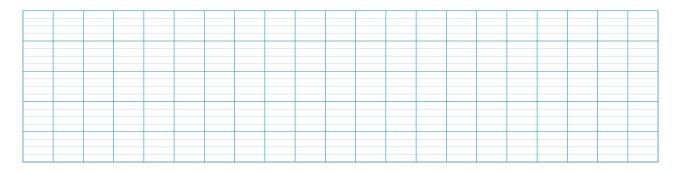
10. Donner la requête SQL permettant d'obtenir le nom de tous les albums sortis entre 1999 et 2010.



11. Donner la requête SQL permettant d'obtenir le titre et la durée de tous les morceaux, triés par ordre décroissant de durée, de tous les morceaux de l'artiste dont l'identifiant est 200.



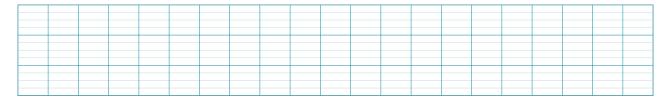
12. Les stars étant capricieuses, certaines veulent changer de nom. Donner la requête permettant à 'Maître Gims' de devenir 'Gims' dans la table des artistes.



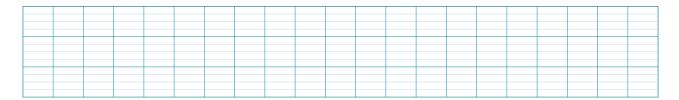
On rajoute maintenant les tables pour les utilisateurs :

Utilisateur(util_id INTEGER, nom TEXT, e-mail TEXT, adresse TEXT)
Ecoute(id_ecoute INTEGER, titre_id TEXT, util_id INTEGER, date DATE)

13. Expliquer pourquoi le couple (titre_id ,util_id) ne peut pas être une clef primaire.



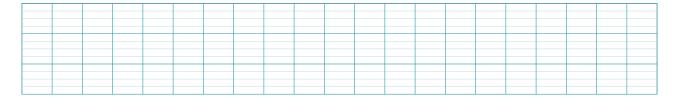
14. Donner la requête SQL permettant d'ajouter l'utilisateur numéro 2179, qui s'appelle Bob VHS, dont l'email est bob.vhs@hotmail.com et qui habite à New York.



15. Traduire le requête suivante en langage naturel :

SQL

```
SELECT COUNT(DISTINCT titre) FROM Morceau
JOIN Ecoute ON Morceau.titre_id = Ecoute.titre_id
WHERE date = "2020-12-12";
```



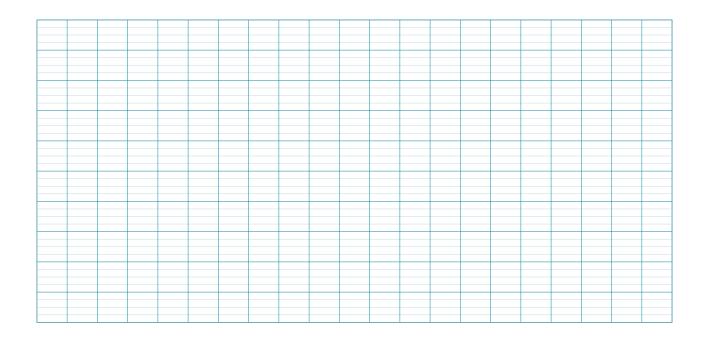
Exercice 2 : renverser une file avec une pile

Écrire en Рүтном une fonction renverse_file qui

- en entrée prend une file;
- ne renvoie rien mais **utilise une pile** pour renverser la file.

Exemple d'utilisation

```
>>> print(F)
2 -> 3 -> 5 -> 1
>>> renverse_file(F)
>>> print(F)
1 -> 5 -> 3 -> 2
```



Exercice 3: fonction mystere

Pour désigner une pile on donnera ses éléments en partant du sommet vers le fond. Ainsi 2 : 3 représentera la pile

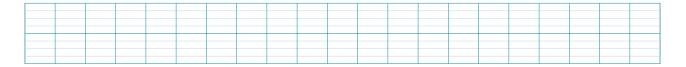


On considère la fonction suivante :

Python

```
def mystere(pile1, pile2):
    if est_vide(pile1):
        return pile2
    else:
        empiler(pile2, depiler(pile1))
        return mystere(pile1, pile2)
```

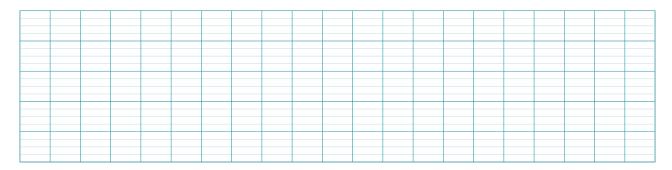
1. Dans cette question on a p = 30 : 20 : 10 et q = 40 : 50 : 60. Que renvoie l'appel mystere(p,q)?



2. Expliquer en une phrase la fonction mystere.



3. Dans cette question on a p = 10 : 20. Que se passe t-il lors de l'appel mystere(p, p)?



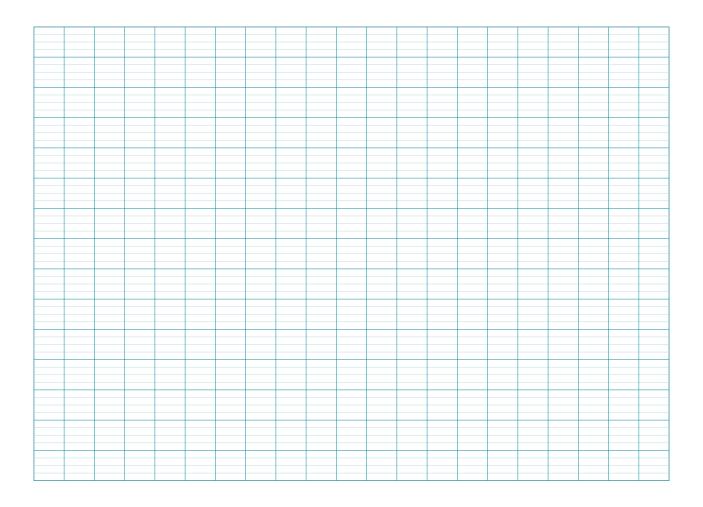
Exercice 4: maximum d'une file

Écrire en PYTHON une fonction max_file qui

- en entrée prend une file non vide composée d'int positifs;
- renvoie le maximum de cette file. Attention la file doit être remise dans l'état initial et aucune autre structure de données (pile, file, liste) ne doit être utilisée.

Exemple d'utilisation

```
>>> print(F)
>>> 2 -> 3 -> 5 -> 1
>>> max_file(F)
>>> 5
>>> print(F)
>>> 2 -> 3 -> 5 -> 1
```



Exercice 4: analyse de code et POO

Tu viens d'être nommé développeur dans une entreprise qui gère les systèmes d'alarmes pour les maisons des particuliers. Ton prédécesseur a commencé à écrire une classe **Alarme**, qui doit normalement répondre aux exigences suivantes :

- l'alarme peut être activée et désactivée;
- chaque intrusion détectée doit être systématiquement consignée dans un journal;
- en cas d'intrusion, si l'alarme est activée, un SMS doit être envoyé au centre de télésurveillance.

Son code et la documentation sont donnés en annexe.

1. On exécute le script suivant :

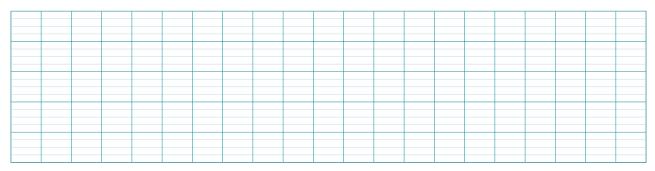
Python

```
from alarme import Alarme

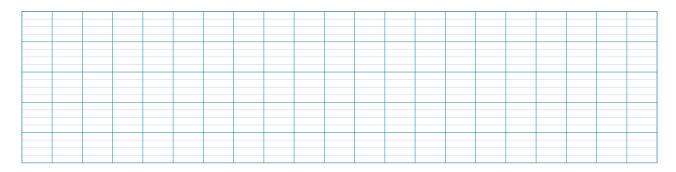
alarme1 = Alarme("Loritz", "971971971", False)
alarme2 = Alarme("Poincaré", "971971971", True)
alarme1.intrusion()
alarme1.activer()
```

alarme1.intrusion()
alarme1.desactiver()
alarme2.intrusion()

Donner les contenus des SMS envoyés.

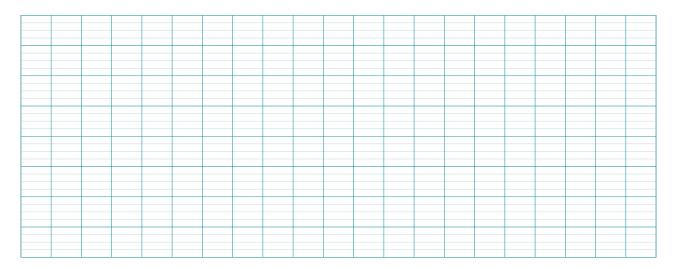


2. Que contient alarme1.journal à la fin du script?



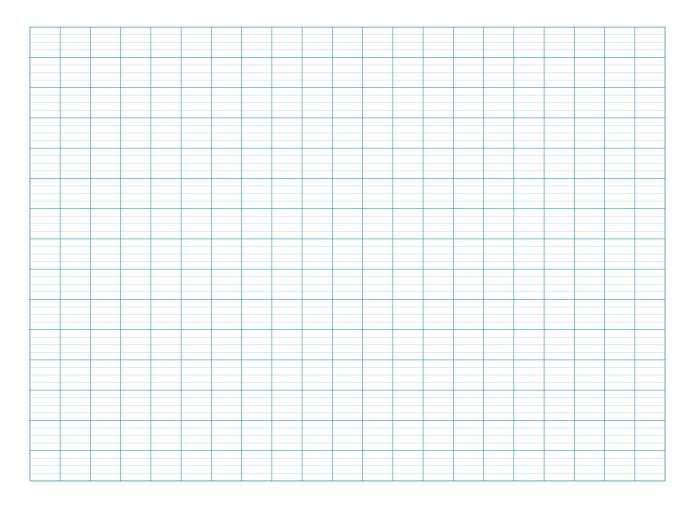
3. En testant le code, on constate que lorsque l'alarme est désactivée les intrusions ne sont pas enregistrées dans le journal.

D'où vient l'erreur? Proposer une correction.



4. On constate que le journal de bord consigne parfois des envois de SMS que le centre n'a jamais reçus : le système a bien tenté de les envoyé mais l'envoi a échoué.

Proposer une correction du code telle que si l'envoi d'un SMS échoue, cela soit consigné dans le journal.



5. Écrire une méthode d'instance efface_journal qui efface le journal de l'instance.

