

## Epreuve Regroupée

Janvier 2025

**BRANCHE : ALGORITHMES ET LANGAGE DE  
PROGRAMMATION OBJET (ALP)**

**CLASSE : ESIG1 Classes A et B  
(Groupes 1, 2 et 3)**

**DATE : 14 janvier 2025**

**NOM : .....**

**PRÉNOM : .....**

**Enseignants : C. VOGT / J. TRAVNJAK / M. STEINLE**

**N° du poste de travail : ESIG-PB.....**

**N° de clé USB : .....**

## Modalités

- Durée : 180 minutes.
- Travail individuel.
- Documentation personnelle (livres, papiers) : Autorisée.
- Documentation électronique (clé USB, ...) : Autorisée si recopiée avant le début de l'épreuve.
- Tout partage de ressources de votre poste de travail avec le réseau ou toute autre tentative de communication (que ce soit avec un humain ou un outil d'intelligence artificielle) seront considérés comme de la fraude et sanctionnés par la note minimale.  
Cela comprend la présence d'un téléphone portable, montre connectée, lunettes connectées, clé USB ou tout autre dispositif de stockage ou de communication à proximité immédiate de votre place de travail : tout ceci doit être posé à l'endroit indiqué par le/la surveillant-e.

## Pour démarrer

- **Avant** le début de l'épreuve, votre documentation sur disque externe, clé USB, ... doit être recopiée sur **C:\ESIGUsers\Doc**. Pour cela, connectez-vous au réseau sur le poste de travail qui vous a été attribué.
- **Une fois l'épreuve commencée**, copiez dans **C:\ESIGUsers** le contenu du dossier réseau de votre groupe, qui vous sera indiqué au début de l'épreuve de façon à avoir un répertoire **C:\ESIGUsers\Eléments ER ALP 14-01-25**.
- Ouvrez ce dossier **C:\ESIGUsers\Eléments ER ALP 14-01-25**.
- Ouvrez Thonny, puis ouvrez les fichiers indiqués dans l'énoncé. **N'oubliez pas d'indiquer vos nom et prénom à l'intérieur des fichiers modifiés ainsi que dans le nom du fichier.**

## Consignes générales

### Sur l'organisation

- Lisez tous les documents fournis.
- Complétez les procédures ou fonctions demandées de manière à ce qu'elles répondent aux spécifications de l'énoncé.
- Vous rendrez les fichiers correspondants sur la clé USB qui vous sera remise quand vous serez prêt à rendre.

### Sur la programmation en Python

- **Interdiction** d'utiliser des fonctions prédéfinies comme `max`, `min`, `index`, `count`, `sum`, `mean` ou équivalent dans des modules fournis par python.
- Les éléments suivants sont explicitement autorisés : la fonction `len`, les slices, le mot clé `in` dans toutes ses formes, les « compréhensions ». En cas de doute, demandez au responsable d'énoncé.
- Pas de résultats ou de données codés « en dur » : votre code doit pouvoir s'adapter à d'autres jeux de données.
- Les seules variables globales autorisées sont les constantes.

C'est à vous de vérifier que les fichiers enregistrés et rendus contiennent bien la dernière version de votre travail

# 2024 Wrapped

## Contexte

Ça y est, 2024 s'est terminée en beauté et vous recevez sur votre application musicale préférée la revue de vos goûts musicaux pour l'année écoulée.

Vous devez réaliser le programme qui s'occupe de créer cette revue en fonction d'une liste de chansons écoutées et de leur durée d'écoute mensuelle.

L'objectif de ce programme est de vous présenter les “stories” comme :

- La liste des musiques écoutées durant l'année
- Le nombre de morceaux par genre
- Votre morceau préféré
- Etc.

## Organisation des données

Un fichier à part, nommé `music_data.py`, vous est fourni avec les données suivantes :

- `LST_MORCEAUX` : cette liste contient tous les morceaux écoutés de cette année pour une personne. Elle a été réduite pour les besoins de cet examen. En effet, dans la réalité on écoute bien plus de 20 morceaux en une année ;).
- `TAB_MINUTES` : ce tableau 2d contient le nombre de minutes pour chaque morceau et chaque mois de l'année.  
Chaque ligne représente un morceau, chaque colonne représente un mois. Les lignes sont donc parallèles à la liste `LST_MORCEAUX`, et les colonnes sont parallèles à la liste `LST_MOIS`
- `DCT_MORCEAU_GENRE` : ce dictionnaire permet de connaître le genre musical d'un morceau de musique, le nom du morceau étant la clé, et la valeur étant le genre correspondant
- `DCT_MORCEAU_ARTISTE` : ce dictionnaire permet de connaître l'artiste qui interprète un morceau. Le morceau étant la clé, et la valeur étant l'artiste qui interprète la chanson.

## Questions

Votre travail s'effectuera dans le fichier `musique_votrenom.py`

Une partie du code est déjà présente, vous devez compléter les fonctions et les procédures tel que demandé dans chaque question et utiliser leur(s) paramètre(s) pour accéder aux différentes structures de données. Vous êtes libres d'ajouter des fonctions et des procédures selon vos besoins et votre manière de décomposer les problèmes.

Cependant, **vous ne devez pas** modifier le code existant (`def main()`, nom et paramètres des procédures/fonctions à compléter, ...).

**NB :** Pour alléger les aperçus de la console, nous vous montrons une partie de l'output, vous trouverez cependant en annexe (*output.txt*) la totalité des outputs de chaque question pour comparaison complète.

Bonne chance !

## Question 1 : Récap général

Dans cette question, votre objectif est de faire une revue générale des écoutes musicales de l'année 2024.

a) Complétez la procédure `afficher_recap_general(dico_morceaux_artistes)`, qui affiche un récapitulatif général des morceaux écoutés durant l'année ainsi que de l'artiste associé.

b) Complétez la fonction `creer_dico_genres(dico_morceau_genres)` pour qu'elle **retourne** un nouveau dictionnaire, ayant pour clé un genre, et pour valeur le nombre de morceaux écoutés de ce genre.

c) Créez ensuite une fonction qui trouve le genre le moins écouté de l'année (vous pouvez ré-utiliser le dictionnaire créé ci-dessus).

d) Enfin, ajoutez le code nécessaire dans la procédure `afficher_recap_genres(dico_genres_ecoutes)`, pour appeler les fonctions complétées/créées précédemment et afficher la répartition des écoutes par genre ainsi que le genre le moins écouté de l'année.

Ci-dessous un aperçu du résultat attendu :

Question 1 :
<pre>Voici ton récap annuel général :  How I could Just Kill a man: Rage against the machine Tadow: Masego From Gaza, with love: Saint Levant Hind's hall: Macklemore (.....)  Voici la répartition de tes écoutes par genre :  Funk metal: 2 Trip hop: 2 Hip hop: 3 Old school hip-hop: 3 Rap :6 (.....)  Durant cette année, le genre que tu as le moins écouté est : Musique du monde</pre>

Figure 1 : Extrait de la console pour la question 1

## Question 2 : Statistiques de base

a) Complétez la fonction `calculer_minutes_tot(tab2d_minutes)` afin qu'elle retourne le nombre de minutes total passé à écouter de la musique durant toute l'année. Pour connaître le nombre de minutes totales, vous devez additionner toutes les minutes du tableau 2d fourni. L'affichage du résultat doit être géré dans la procédure `afficher_minutes_tot(tab2d_minutes)`

b) Complétez la procédure `afficher_morceau_maximum(tab2d_minutes, lst_morceaux)` afin qu'elle affiche le nom du morceau que la personne a écouté le plus durant toute l'année.

Il faudra également afficher le nombre total de minutes passées à écouter le morceau.

**Conseil :** Vous êtes libres de décomposer ce problème en autant de fonctions que vous le souhaitez. Vous pouvez également faire tout le traitement dans la procédure d'affichage. A vous de choisir l'approche avec laquelle vous êtes le plus à l'aise.

Ci-dessous un aperçu du résultat attendu :

Question 2 :
Durant cette année, tu as passé au total 61380 minutes à écouter tes morceaux préférés.
Ton morceau préféré pour cette année est Hind's hall. Tu as passé 3563 minutes à l'écouter, tu es un fan dédié !

Figure 2 : Extrait de la console pour la question 2

## Question 3 : Statistiques avancées

Pour cette dernière question, on vous demande de faire un récapitulatif d'écoute par artiste. L'objectif étant de savoir combien de minutes annuelles ont été consacrés à chaque artiste.

Vous aurez remarqué que dans la liste des 20 morceaux, il y a parfois des morceaux de mêmes artistes. Il va donc falloir trouver une manière d'additionner tout ça. Vous êtes libres d'organiser votre code comme vous le souhaitez pour ce dernier challenge, mais vous devez compléter la procédure d'affichage suivante :

```
def afficher_minutes_artiste(tab2d_minutes, lst_morceaux,
dct_morceaux_artistes)
```

Attention, les artistes ne doivent apparaître qu'une seule fois dans votre récapitulatif. Par exemple pour Gaël Faye, même s'il y a 3 morceaux, il vous faudra afficher Gaël Faye qu'une fois avec les minutes totales couvrant tous les morceaux.

En plus d'afficher le nombre de minutes passés pour chaque artiste, vous allez déterminer un statut de « fan » pour l'utilisateur. Voici les cas possibles :

- Écoute de plus de 6000 minutes : *Tu es méga fan*
- Écoute entre 4000 et 6000 minutes : *Tu es super fan*
- Écoute entre 2000 et 4000 minutes : *Tu es amateur*
- Écoute inférieure à 2000 : *Tu peux mieux faire*

Ci-dessous un aperçu du résultat attendu :

Question 3 :
<p>Découvre maintenant les artistes qui t'ont accompagné tout au long de 2024</p> <p>Rage against the machine : 5637 minutes: Tu es super fan !</p> <p>Masego : 2766 minutes: Tu es amateur !</p> <p>Saint Levant : 3424 minutes: Tu es amateur !</p> <p>Macklemore : 3563 minutes: Tu es amateur !</p> <p>(...)</p>

Figure 3 : Extrait de la console pour la question 3

**Annexe** : output.txt -> fichier contenant la totalité de l'output console

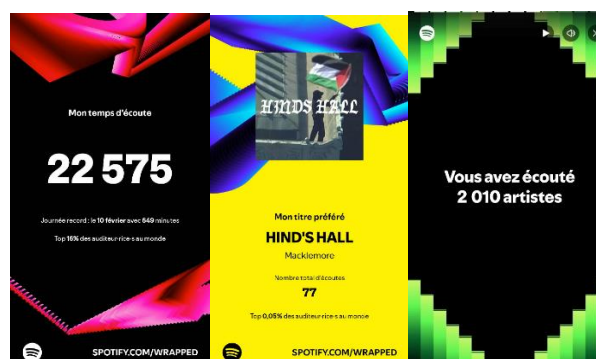


Figure 4 : Exemple Spotify