

Langage de programmation 1 : Projet

Comme dit tout au long de ce cours, pratiquer est la meilleure façon d'apprendre efficacement à programmer. Vous aurez l'occasion, à travers ce projet, de valider vos connaissances acquises durant ce module de programmation Python, mais aussi d'approfondir les notions vues et découvrir de nouvelles pratiques.

Consignes générales

- Ce projet sera à réaliser par groupe de **4 élèves**
- Vous devez choisir un sujet parmi les deux proposés. La note finale de ce projet sera calculée sur la base du livrable mais aussi de la qualité du passage à l'oral et des réponses aux questions.
- Votre rendu devra être mis dans un **dossier zippé** nommé :

`sujet_<1 ou 2>_NOMELEVE1_NOMELEVE2_NOMELEVE3_NOMELEVE4.zip`


- Le livrable devra être composé des éléments suivants:
 1. Les données sources utilisées rassemblées dans un dossier
 2. Vos scripts Python regroupés dans un dossier
 3. Un rapport au format .pdf structuré qui doit illustrer votre réflexion pour la réalisation du projet. Il doit comporter à minima les éléments suivants:
 - Le sujet choisi
 - Analyse : Restitution des analyses demandées en première partie du sujet
 - Méthodologie : Description des étapes de développement, des outils, bibliothèques utilisées etc.
Répartition du travail dans le groupe
 - Conception : Architecture du programme (mettre un schéma montrant sa structure, comment les scripts interagissent entre eux) et de l'interface (avec des captures d'écran du rendu visuel)
 - Implémentation : Fonctionnalités développées, exemples de code pertinents et explications des choix effectués
 - Tests et validation : Méthodes de test appliquées au projet, i.e. qu'avez vous fait pour vous assurer que le programme fonctionne bien ?
 - Perspectives :
 - Si le projet était à refaire qu'auriez-vous fait différemment ?
 - Qu'avez vous appris via ce projet ? Difficultés rencontrées ? Pour cette partie, une réponse de chaque membre du groupe est attendue
 - Annexe :
 - Les instructions pour exécuter votre programme
 - Bibliographie : liens vers les ressources utilisées

- L'ensemble du livrable devra être envoyé à mon adresse e-mail (imane.loukah@univ-paris1.fr) pour le samedi 23 novembre 23:59 dernier délai. **Tout retard sera sanctionné.**
- La soutenance devra reprendre les points principaux de votre réalisation : Bibliothèques utilisées, structure du programme que vous avez créé, principales fonctionnalités, démonstration de son fonctionnement et décrire brièvement la partie gestion de projet. Vous devrez préparer une présentation PowerPoint pour un passage à l'oral de 10 minutes. Les oraux seront réalisés sur la dernière séance du cours (le 26 novembre 2024).

L'utilisation d'IA Générative (type ChatGPT) n'est pas autorisée.

Toute tentative de plagiat ou utilisation d'IA Générative sera sanctionnée par la note de 0.

Remarques :

- La qualité et la clarté de vos codes seront pris en considération dans la note du projet. Il sera important de bien organiser vos codes sous forme de fonctions, de bien commenter vos scripts pour que je puisse comprendre votre démarche en les lisant.
-  Vous devrez prendre en compte dans votre code le fait qu'il sera exécuté sur mon PC. Je ne dois pas avoir à modifier à la main tous les chemins que vous écrivez dans vos scripts.
- Attention à la casse pour les différents sujets (plus spécifiquement pour les parties 2 & 3) : si un utilisateur écrit "TOTO", est ce qu'il arrivera à récupérer le résultat "Toto" présent dans les bases de données ?
- Il n'est pas demandé d'ajouter les résultats de l'analyse de la partie 1 dans l'interface graphique à construire en partie 3
- Toute prise d'initiative supplémentaire sera prise en compte et valorisée

Sujet 1 : The Movie DataBase & Netflix

L'objectif de ce sujet est d'exploiter des données provenant de The Movie DataBase et le catalogue de Netflix afin de construire un moteur de recommandation « simple » de films (appelé simple ici car ne fait pas appel à du Machine Learning). Ce moteur de recherche/recommandation permettra aux utilisateurs de découvrir des films qui pourraient les intéresser en fonction de leurs préférences.

Les données à votre disposition proviennent de Kaggle et vous pouvez directement les télécharger via les liens présentés plus bas.

- **Full TMDb Movies Dataset 2024** : <https://www.kaggle.com/datasets/asaniczka/tmdb-movies-dataset-2023-930k-movies>

Ces données sont extraites de The Movies Database et recensent des informations sur près de 930 000 films, comme par exemple les titres, les genres, année de sortie... Il y a 24 colonnes mais sont présentées ci-dessous uniquement les 10 premières, toutes ne seront pas utilisées :

Libellé de la variable	Définition
id	Identifiant unique pour chaque film
title	Titre du film
vote_average	Note moyenne du film donnée par les spectateurs
vote_count	Nombre de votes reçus
status	Statut du film (ex. : Released, Post Production)
release_date	Date de sortie du film
revenue	Recettes du film
runtime	Durée du film en minutes
adult	Indicateur si le film est destiné à un public adulte
backdrop_path	URL de l'image de fond du film
budget	Budget alloué au film
homepage	URL de la page officielle du film
imdb_id	Identifiant IMDb du film
original_language	Langue originale dans laquelle le film a été produit
original_title	Titre original du film
overview	Description ou résumé du film
popularity	Score de popularité du film
poster_path	URL de l'affiche du film
tagline	Slogan ou phrase mémorable associée au film
Genres	Liste des genres auxquels le film appartient
production_companies	Liste des sociétés de production impliquées dans la production du film
production_countries	Liste des pays participant à la production du film
spoken_languages	Liste des langues parlées dans le film
keywords	Mots clés associés au film

- **Netflix Movies and TV Shows** : <https://www.kaggle.com/datasets/rahulvyasm/netflix-movies-and-tv-shows>

Ce dataset contient des informations sur les films et séries disponibles sur Netflix, incluant des détails tels que les titres, la date de sortie, la note... La liste des données présentes sont détaillées dans le tableau ci-dessous :

Libellé de la variable	Définition
show_id	Identifiant unique pour chaque titre
type	Catégorie du titre, soit 'Film' soit 'Série TV'
title	Nom du film ou de la série TV
director	Réalisateur(s) du film ou de la série TV
cast	Liste des acteurs/actrices principaux dans le titre
country	Pays où le film ou la série TV a été produit
date_added	Date à laquelle le titre a été ajouté à Netflix
release_year	Année de la sortie initiale du film ou de la série
rating	Classification par âge du titre
duration	Durée du titre, en minutes pour les films et en saisons pour les séries TV
listed_in	Genres auxquels le titre appartient
description	Bref résumé du titre

Partie 1 : Analyse descriptive et visualisation des données :

Note: Le code de cette partie peut être réalisé dans un notebook (pas obligatoire toutefois...). Les analyses quant à elles devront figurer dans le rapport

Vous devrez tout d'abord effectuer une description classique des données (ex. nombre d'observations, de variables, type des variables...) puis répondre aux questions suivantes:

1. Quelle est la note moyenne par genre des films ? Représenter graphiquement
2. Quelle est la répartition des années de sortie des films disponibles sur TMDB par rapport aux films ajoutés sur Netflix ? Représenter graphiquement
3. Analyse des facteurs influant sur la note : Analyse des facteurs influant sur la popularité : Quels critères influent sur la note d'un film ? Utiliser des graphiques pour illustrer ces relations, et mettre en avant les tendances importantes identifiées

Partie 2 : Moteur de recherche/recommandations

Développer un moteur de recommandation qui suggère des films à partir de données saisies par l'utilisateur :

- Si un utilisateur choisit un ou plusieurs genres et une année, recommander des films correspondant aux critères. Afficher les résultats suivant leur note et leur popularité, en n'affichant que les noms de films, genre, date de sortie, synopsis, langue, note. Si plusieurs films sont ex-aequo, afficher en premier celui qui a reçu le plus de votes
- Demander des informations sur ses genres favoris, pays de production, année, catégorie d'âge, durée du film etc... Le programme doit laisser la possibilité à l'utilisateur de ne pas sélectionner toutes les options. Pour cette suggestion, vous pouvez par exemple donner les 5 films les plus récents, ainsi que les 5 films les mieux notés. Afficher les résultats suivant leur note et leur popularité. Si plusieurs films sont ex-aequo, afficher en premier celui qui a reçu le plus de votes
- Si l'utilisateur saisit des titres de films qu'il a appréciés, suggérer des films similaires basées sur des caractéristiques communes, comme par exemple un genre similaire, note égale ou supérieure, langue similaire, keywords similaires... (choisir des critères pertinents qui selon vous peuvent aider à déterminer la similarité entre 2 films) puis mettre en évidence ceux disponibles sur Netflix.

Partie 3 : Création d'interface graphique

Une fois le programme de la partie 2 réalisé, l'idée est de ne pas devoir écrire des lignes de code pour exécuter votre programme une fois le script exécuté. Il faudra donc créer une interface afin que l'utilisateur soit guidé pour l'utilisation de votre programme.

Pour ce faire, vous devrez utiliser la librairie Tkinter¹ pour réaliser une interface graphique. Cette interface devra comporter les éléments suivants:

- Possibilité pour l'utilisateur de saisir les différents critères (selon ce qui sera fait à la partie 2)
- Affichage des résultats conformément aux éléments demandés dans la partie 2

Idée: Vous pouvez améliorer l'interface en y intégrant des liens vers les pages Wikipedia, IMDb ou ce qui vous semble pertinent.

*Note: en cas de difficultés, il est possible de faire une version plus « simple » de l'interface en demandant les infos à l'utilisateur avec des **input** puis en affichant les résultats dans la console avec **print**. La version avec Tkinter étant plus avancée, elle sera bien sûr plus valorisée au niveau du barème que la version simple de l'interface.*

¹ Quelques références pour Tkinter :

- <https://python.doctor/page-tkinter-interface-graphique-python-tutoriel>
- <https://realpython.com/python-gui-tkinter/>
- <https://www.geeksforgeeks.org/python-gui-tkinter/>

Sujet 2 : Logements Airbnb dans les grandes villes européennes

L'objectif de ce sujet est d'exploiter des données provenant de la plateforme de location Airbnb pour plusieurs villes européennes afin de les analyser et proposer un moteur de recherche/recommandation aux utilisateurs. Les données englobent des informations pour les villes suivantes : Amsterdam, Athènes, Barcelone, Berlin, Budapest, Lisbonne, Londres, Paris, Rome, Vienne.

Les fichiers sont disponibles sur le site de Kaggle : <https://www.kaggle.com/datasets/thedevastator/airbnb-prices-in-european-cities/data>

Vous y trouverez pour chaque ville un fichier « weekdays », donnant les caractéristiques de prix pour les logements en semaine, et un fichier « weekend » donnant les prix le weekend.

<nom_de_la_ville>_weekdays.csv : Tous les fichiers de données « weekdays » ont la même structure qui est la suivante :

Libellé de la variable	Définition
id	Identifiant de l'hébergement
realSum	Prix total de l'hébergement pour deux personnes et deux nuits en EURO
room_type	Type d'hébergement
room_shared	Variable binaire indiquant si c'est une chambre partagée
room_private	Variable binaire indiquant si c'est une chambre privée
person_capacity	Nombre maximum de personnes admises dans l'hébergement
host_is_superhost	Variable binaire indiquant si l'hôte est un « Superhôte » sur Airbnb
multi	Variable binaire indiquant si l'annonce appartient à des hôtes ayant 2 à 4 offres
biz	Variable binaire indiquant si l'annonce appartient à des hôtes ayant plus de 4 offres
cleanliness_rating	Note de propreté
guest_satisfaction_overall	Note globale de l'annonce par les clients
bedrooms	Nombre de chambres (0 si c'est un studio)
dist	Distance du centre-ville
metro_dist	Distance de la station de métro la plus proche en km
attr_index	Indice d'attraction de l'emplacement du logement
attr_index_norm	Indice d'attraction normalisé (échelle 0-100)
lng	Longitude de l'emplacement du logement
lat	Latitude de l'emplacement du logement

<nom_de_la_ville>_weekends.csv : Idem pour les fichiers de données « weekend » :

Libellé de la variable	Définition
realSum	Prix total de l'hébergement pour deux personnes et deux nuits en EURO
room_type	Type d'hébergement
room_shared	Variable binaire indiquant si c'est une chambre partagée
room_private	Variable binaire indiquant si c'est une chambre privée
person_capacity	Nombre maximum de personnes admises dans l'hébergement
host_is_superhost	Variable binaire indiquant si l'hôte est un « Superhôte » sur Airbnb
multi	Variable binaire indiquant si l'annonce appartient à des hôtes ayant 2 à 4 offres
biz	Variable binaire indiquant si l'annonce appartient à des hôtes ayant plus de 4 offres
cleanliness_rating	Note de propreté
guest_satisfaction_overall	Note globale de l'annonce par les clients
bedrooms	Nombre de chambres (0 si c'est un studio)
dist	Distance du centre-ville
metro_dist	Distance de la station de métro la plus proche en km
attr_index	Indice d'attraction de l'emplacement du logement
attr_index_norm	Indice d'attraction normalisé (échelle 0-100)
lng	Longitude de l'emplacement du logement
lat	Latitude de l'emplacement du logement

Partie 1 : Etude des caractéristiques influant sur les prix

Note: Le code de cette partie peut être réalisé dans un notebook (pas obligatoire toutefois...). Les analyses quant à elles devront figurer dans le rapport

Vous devrez tout d'abord effectuer une description classique des données (ex. nombre d'observations, de variables, type des variables...) puis répondre aux questions suivantes :

1. Y a-t-il une différence de prix entre les logements en semaine et ceux du weekend pour chaque ville ?
Représenter graphiquement
2. Quelle est la ville où les appartements privés pour au moins 4 personnes sont les plus chers en moyenne ?
Les moins cher ? Représenter graphiquement

3. Regarder comment les éléments géographiques comme l'emplacement et l'accès aux transports en commun, la ville... peuvent affecter la tarification des logements. Utiliser des graphiques pour illustrer ces relations, et mettre en avant les tendances importantes identifiées
4. Etudier l'impact des dynamiques sociales, notamment l'effet des évaluations, popularité et caractéristiques des hôtes (superhôte) sur les prix. Utiliser des graphiques pour illustrer ces relations, et mettre en avant les tendances importantes identifiées

Partie 2 : Moteur de recherche/recommandations

Cette partie a pour but de créer un outil de recommandation qui puisse aider un utilisateur à effectuer une recherche de location suivant certains critères :

- « Où partir en vacances selon mes contraintes ? » : L'utilisateur saisit des caractéristiques sur le type de location, capacité d'accueil semaine/weekend etc (choisir 5 critères à proposer à l'utilisateur). Afficher en sortie du programme les fourchettes de prix pour chaque ville, les moyennes de prix et recommander à l'utilisateur les 3 destinations les moins chères
- L'utilisateur choisit une ville ainsi que les options suivantes à minima (Le programme doit laisser la possibilité à l'utilisateur de ne pas sélectionner toutes les options) :
 - Choix du type de location (appartement entier, chambre...)
 - Capacité d'accueil (nombre de personnes)
 - Distance du centre ville
 - Voyage en semaine ou le weekend
 - Budget

Lorsque les critères choisis par l'utilisateur correspondent à des annonces, afficher pour chacune des villes les résultats en les affichant par pertinence (ex. Prix ou note). Si aucun résultat de correspond aux critères, il faudra inviter l'utilisateur à changer certains critères.

Partie 3 : Interface graphique

Une fois le programme de la partie 2 réalisé, l'idée est de ne pas devoir écrire des lignes de code pour exécuter votre programme une fois le script exécuté. Il faudra donc créer une interface afin que l'utilisateur soit guidé pour l'utilisation de votre programme.

Pour ce faire, vous devrez utiliser la librairie Tkinter² pour réaliser une interface graphique. Cette interface devra comporter les éléments suivants:

- Possibilité pour l'utilisateur de saisir les différents critères (selon ce qui sera fait à la partie 2)

² Quelques références pour Tkinter :

- <https://python.doctor/page-tkinter-interface-graphique-python-tutoriel>
- <https://realpython.com/python-gui-tkinter/>
- <https://www.geeksforgeeks.org/python-gui-tkinter/>

- Affichage des résultats selon ce qui a été demandé dans la partie 2

Idée: Vous pouvez améliorer l'interface en y intégrant des liens vers les pages Wikipedia, des cartes ou ce qui vous semble pertinent.

*Note: en cas de difficultés, il est possible de faire une version plus « simple » de l'interface en demandant les infos à l'utilisateur avec des **input** puis en affichant les résultats dans la console avec **print**. La version avec Tkinter étant plus avancée, elle sera bien sûr plus valorisée au niveau du barème que la version simple de l'interface.*