

TAILLY Emmanuel

EL MAGRI Ali

Projet n°9 : La météo d'une journée donnée



L'objectif de ce projet est de déterminer une météo détaillée d'une journée donnée.

Paramètres :

- la date choisie doit être paramétrable ;
- la zone géographique considérée (station, département, région) doit être paramétrable.

Problématique : *Comment concevoir une solution paramétrable et intuitive qui permet de fournir des prévisions ou des données météorologiques détaillées pour une date et une zone géographique spécifiques ?*

Introduction :

Le projet "La météo d'une journée donnée" vise à offrir une solution simple et paramétrable permettant à tout utilisateur de consulter les données météorologiques détaillées d'une date précise, pour une région ou une zone géographique donnée, en s'appuyant sur des données fiables provenant de services spécialisés et tout sa en utilisant seulement python.

Explication du programme python :

Fonction charger_donnees_meteo(fichiers_csv) :

- Cette fonction charge les données des fichiers CSV listés dans fichiers_csv.
- Chaque ligne des fichiers est transformée en dictionnaire grâce à csv.DictReader
- Les données de tous les fichiers sont regroupées dans une liste (data).
- Si un fichier n'est pas trouvé, un message d'erreur est affiché (messagebox)

Fonction obtenir_meteo(data, ville, journee):

- Cette fonction filtre les données pour obtenir celles correspondant à la ville et à la date fournies par l'utilisateur.
- Si une correspondance est trouvée, les informations météorologiques comme les températures, la force du vent, sont mise dans une chaîne de caractères .
- Si aucune correspondance n'est trouvée, la fonction renvoie un message indiquant que les données sont introuvables.

Fonction afficher_meteo():

- Cette fonction lit la ville et la date saisies par l'utilisateur dans les champs d'entrée, puis utilise la fonction obtenir_meteo pour récupérer les données correspondantes.
- Les résultats sont affichés dans une zone de texte (texte_resultat).

Fonction afficher_graphique():

- Cette fonction extrait les températures minimale et maximale à partir des données obtenues pour tracer un graphique en barres.
- Le graphique est intégré dans l'interface Tkinter à l'aide de FigureCanvasTkAgg.

Les fichiers CSV contenant les données météorologiques sont listés dans

fichiers_csv = []

Ces fichiers contiennent toute les données a traité

Schema d'utilisation :

1. L'utilisateur ouvre l'application.
2. Dans l'onglet "Recherche" :
 - a. Il saisit une ville et une date (au format AAAAMMJJ).
 - b. Il clique sur le bouton "Rechercher" pour afficher les données dans une zone de texte.
3. Dans l'onglet "Graphique" :
 - a. Il clique sur "Afficher Graphique" pour visualiser un graphique des températures minimales et maximales.

4. Si les champs sont vides ou si les données sont introuvables, un message d'avertissement s'affiche.

Problèmes rencontrés :

Dans le cadre de cette SAE, nous avons dû apprendre à utiliser une nouvelle bibliothèque (Principalement grace au site “Tkinter - l’informatique, c’est fantastique” et grace a des videos youtube d’explications), Tkinter , qui est un module standard de Python pour la création d'interfaces graphiques. Grâce à cette bibliothèque, nous avons pu intégrer une interface utilisateur intuitive et interactive à notre programme de traitement des données météorologiques.

L'utilisation de Tkinter nous a permis de concevoir plusieurs fonctionnalités essentielles :

- Des champs de saisie pour que l'utilisateur puisse renseigner la ville et la date souhaitées ;
- Un système d'onglets pour structurer l'application et séparer les différentes fonctionnalités (recherche et visualisation graphique) ;
- Une zone de texte dynamique pour afficher les résultats de la recherche ;
- Une intégration de graphiques interactifs grâce à l'utilisation conjointe de Tkinter et Matplotlib.

Cette démarche nous a permis d'allier l'aspect algorithmique du traitement des données avec une présentation visuelle conviviale, offrant ainsi une expérience utilisateur nettement améliorée. En parallèle, cela nous a permis de développer des compétences pratiques dans la conception d'interfaces graphiques, une compétence précieuse pour tout développement d'application.

Le deuxième problème rencontré a été la lecture simultanée de plusieurs fichiers CSV. Bien que cela puisse sembler simple au premier abord, ce défi s'est avéré plus complexe qu'il n'y paraissait, en grande partie en raison des erreurs d'étourderie et des subtilités liées au format des fichiers. Trouver une manière efficace de lire et combiner les données provenant de plusieurs fichiers tout en assurant leur cohérence et leur intégrité a nécessité plusieurs itérations.

Conclusion :

Ce projet, "La météo d'une journée donnée", nous a permis d'aborder un ensemble de problématiques liées à la gestion, au traitement et à la visualisation de données météorologiques en utilisant exclusivement Python. À travers l'apprentissage de la bibliothèque Tkinter, nous avons pu concevoir une application intuitive et paramétrable, répondant aux besoins d'un utilisateur souhaitant accéder facilement à des informations météorologiques détaillées pour une zone et une date spécifiques.