

Cours – Exploitation de données météo avec Python

Objectifs

- interroger une **API météo** avec Python
- manipuler des **dictionnaires** et des **listes**
- afficher des données météo
- générer un **QR code** à partir d'informations calculées

Tout le travail se fait dans **un seul fichier** : **meteo.py**.

1. Interroger une API météo (météo actuelle)

Fichier : **meteo.py**

```
import requests

URL = "https://api.open-meteo.com/v1/forecast"

# Coordonnées de Rodez
LAT = 44.35258
LON = 2.57338

params = {
    "latitude": LAT,
    "longitude": LON,
    "current_weather": True,
    "timezone": "Europe/Paris"
}

response = requests.get(URL, params=params)
data = response.json()

print("Les données sont : ", data)
print("Type : ", type(data))
```

Travail à faire

- Exécuter le programme
- Observer le résultat affiché dans la console

Questions

- Que contient la variable **data** ?
- De quel **type** est cette variable ?

2. Affichage de la météo actuelle

Ajouter le code suivant **à la suite du programme** :

```
current = data["current_weather"]

temperature = current["temperature"]
vent = current["windspeed"]

print("Météo actuelle à Rodez")
print("Température :", temperature, "°C")
print("Vent :", vent, "km/h")
```

Travail à faire

- Exécuter le programme
- Vérifier que les données affichées sont cohérentes

Questions

- Que représente `current_weather` ?
- Pourquoi utilise-t-on des crochets `[]` pour accéder aux données ?

3. Associer un code météo à une description

Les conditions météo sont représentées par des **codes numériques**. Nous allons utiliser un **dictionnaire** pour les traduire en texte.

Ajouter le code suivant :

```
weather_codes = {
    0: "Ciel dégagé",
    1: "Peu nuageux",
    2: "Nuageux",
    3: "Couvert",
    61: "Pluie"
}

code = current["weathercode"]

description = weather_codes.get(code, "Temps inconnu")

print("Conditions :", description)
```

Travail à faire

- Exécuter le programme
- Observer le texte affiché

4. Découvrir les prévisions horaires (listes)

Nous allons maintenant récupérer les **prévisions horaires de température**.

Ajouter ce nouveau code (toujours dans le même fichier) :

```
params = {
    "latitude": LAT,
    "longitude": LON,
    "hourly": "temperature_2m",
    "forecast_days": 1,
    "timezone": "Europe/Paris"
}

response = requests.get(URL, params=params)
forecast = response.json()

times = forecast["hourly"]["time"]
temps = forecast["hourly"]["temperature_2m"]

print("Prévisions pour les prochaines heures :")

for i in range(3):
    print(times[i], "→", temps[i], "°C")
```

Questions

- Que contiennent les variables `times` et `temps` ?
 - Que fait `range(3)` dans la boucle `for` ?
-

5. Génération d'un QR Code météo

Nous allons maintenant créer un **QR code** contenant les informations météo obtenues précédemment.

Ajouter le code suivant **à la fin du fichier** :

```
import qrcode

text = (
    "Météo à Rodez\n"
    f"Température : {temperature} °C\n"
    f"Vent : {vent} km/h\n"
    f"Temps : {description}\n"
)

img = qrcode.make(text)
img.save("meteo_rodez.png")

print("QR code généré")
```

Travail à faire

- Exécuter le programme
 - Scanner le QR code généré avec un téléphone
-

6. Exercices

1. Modifier la **ville** (changer latitude et longitude)

2. Personnaliser le texte du QR code

3. Ajouter une ligne :

Bonne journée *

4. Afficher **une seule** prévision horaire au lieu de trois
