



TP 4 : Découverte, organisation et analyse d'articles scientifiques avec Python

Objectif du TP

- Manipuler des listes et dictionnaires
- Parcourir des données avec des boucles
- Effectuer des recherches simples
- Organiser des informations scientifiques
- Réaliser une analyse basique de données

Exercice 1 : Découverte et organisation des articles.

Copier, exécuter et analyser le code suivant :

```
# Liste d'articles scientifiques
articles = [
    {"title": "ChatGPT for Natural Language Tasks", "author": "OpenAI",
"year": 2023},
    {"title": "Tesla Autopilot Safety Analysis", "author": "Tesla Inc.",
"year": 2022},
    {"title": "Smart Grid Management using AI", "author": "A. Smith",
"year": 2023},
    {"title": "Siri and Alexa Performance Study", "author": "Apple &
Amazon", "year": 2021},
    {"title": "Google Translate Improvements", "author": "Google",
"year": 2020}
]
# Affichage des articles
for art in articles:
    print("Title:", art["title"])
    print("Author:", art["author"])
    print("Year:", art["year"])
    print("-"*30)
```

- 1) Ajouter une nouvelle information "keyword" (Exemple "AI") pour chaque article.
- 2) Pourquoi utilise-t-on des dictionnaires pour représenter les articles ?
- 3) Ajouter un nouvel article à la liste.
- 4) Afficher uniquement les articles publiés après 2021.

Exercice 2 : Recherche par mot-clé.

Recopier et exécuter le code suivant (*articles* est défini dans l'exercice précédent).

```
keyword = "AI"
results = [art for art in articles if keyword in art["title"]]
print("Articles trouvés avec", keyword, ":")
```

```
for res in results:
    print(res["title"])
```

- 1) Que signifie: *keyword in art["title"]*
- 2) Expliquer la ligne suivante: *results = [art for art in articles if keyword in art["title"]]*
- 3) Modifier le code pour rechercher plusieurs mots-clés. Exemple : "AI" et "ChatGPT".
- 4) Que se passe-t-il si aucun article ne correspond ?
- 5) Afficher aussi l'auteur et l'année.
- 6) Rendez la recherche insensible à la casse (par exemple : « AI », « ai » et « Ai » doivent être considérés comme identiques).
- 7) Pourquoi est-il important de filtrer les articles avant de les lire entièrement ?

Exercice 3 : Résumé d'article.

Recopier et exécuter le code suivant.

```
article_text = """
ChatGPT is a conversational AI that can answer questions, write texts,
and summarize articles.
It is widely used in education and research.
"""
summary = " ChatGPT is an AI that answers questions, writes and
summarizes texts."
print("Summary:", summary)
```

- 1) Quelle est la différence entre *article_text* et *summary* dans ce code ?
- 2) Comment pourrait-on générer un résumé automatique en Python ?
- 3) Pourquoi est-il important de résumer les articles avant de les utiliser ?
- 4) Que se passerait-il si le texte original est très long ?

Exercice 4 : Analyse des mots-clés.

Etant donné le code Python suivant : (Remarque : *articles* est défini dans l'Exercice 1).

```
keywords = ["AI", "Smart Grid", "Translate"]
for kw in keywords:
    count = sum(1 for art in articles if kw in art["title"])
    print(f"Number of articles containing '{kw}':", count)
```

- 1) Que fait l'instruction *sum(1 for art in articles if kw in art["title"])* ?
- 2) Pourquoi le comptage des mots-clés est-il utile dans une revue de littérature ?
- 3) Comment peut-on afficher la liste des articles pour chaque mot-clé ?
- 4) Comment peut-on trier les mots-clés par nombre d'articles en ordre décroissant ?
- 5) Que se passe-t-il si un article contient plusieurs mots-clés à la fois ?