



IUS
INSTITUT
UNIVERSITAIRE
DES SCIENCES

Faculté des Sciences et Technologie
(FST)

Niveau : L3-FST

Cours : Mathématiques pour l'Informatique

Soumis au chargé de cours : Ismaël SAINT AMOUR

Préparé par : Jameson DOMINIQUE

Date : 07 Juin 2025

TD2- Programmation Mathématiques pour la science des données en Python.

Exercice 1 : Saisie des Ventes d'un Magasin et Graphe en Ligne

✦ Objectif :

Demander à l'utilisateur d'entrer les ventes d'un magasin pour 7 jours.

Calculer le total et la moyenne des ventes.

Afficher un graphe en ligne montrant l'évolution des ventes au fil de la semaine.

```
import matplotlib.pyplot as plt

def exercice_ventes():
    # Saisie des ventes
    jours = ["Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi",
            "Dimanche"]
    ventes = []

    for jour in jours:
        vente = float(input(f"Entrez les ventes pour {jour}: "))
        ventes.append(vente)

    # Calculs
    total = sum(ventes)
    moyenne = total / 7

    print(f"\nTotal des ventes sur la semaine: {total:.2f}€")
    print(f"Moyenne quotidienne: {moyenne:.2f}€")

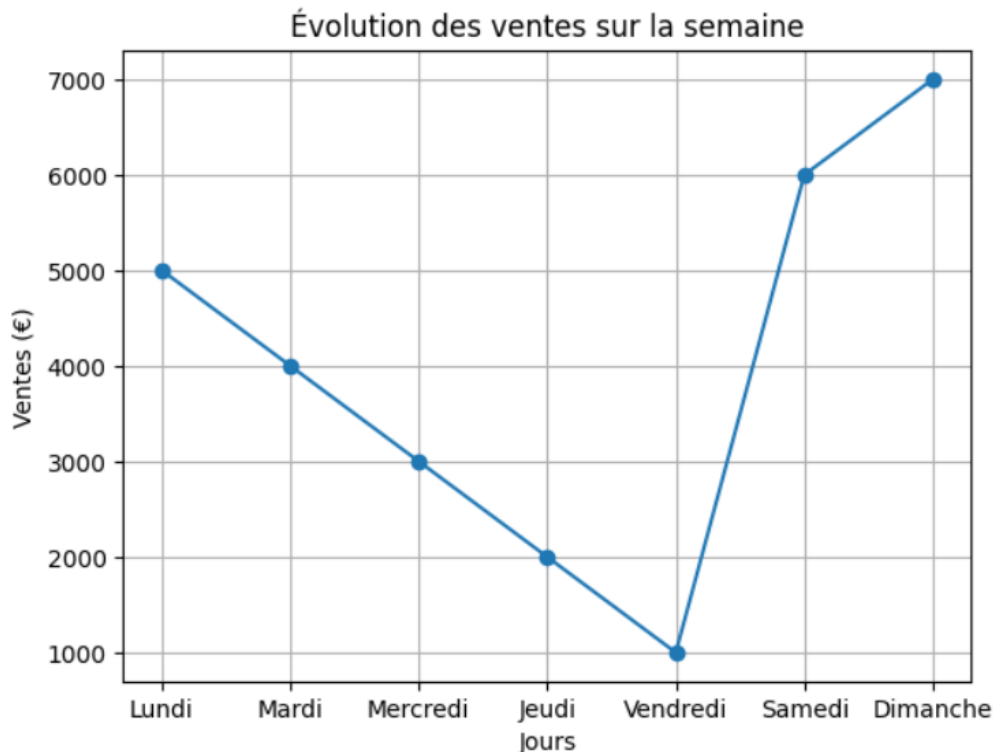
    # Graphique
    plt.plot(jours, ventes, marker='o')
    plt.title("Évolution des ventes sur la semaine")
    plt.xlabel("Jours")
    plt.ylabel("Ventes (€)")
    plt.grid(True)
    plt.show()

exercice_ventes()
```

Entrez les ventes pour Lundi: 5000
Entrez les ventes pour Mardi: 4000
Entrez les ventes pour Mercredi: 3000
Entrez les ventes pour Jeudi: 2000
Entrez les ventes pour Vendredi: 1000
Entrez les ventes pour Samedi: 6000
Entrez les ventes pour Dimanche: 7000

colab.research.google.com –

Total des ventes sur la semaine: 28000.00€
Moyenne quotidienne: 4000.00€



Exercice 2 : Évolution de la Température sur 7 Jours

◆ Objectif :

Demander à l'utilisateur d'entrer les ventes d'un magasin pour 7 jours.

Calculer le total et la moyenne des ventes.

Afficher un graphe en ligne montrant l'évolution des ventes au fil de la semaine.

```
import matplotlib.pyplot as plt

def exercice_temperatures():
    jours = ["Lundi", "Mardi", "Mercredi", "Jeudi", "Vendredi", "Samedi",
            "Dimanche"]
    temperatures = []

    for jour in jours:
        temp = float(input(f"Entrez la température pour {jour} (en °C): "))
        temperatures.append(temp)

    moyenne = sum(temperatures) / 7
```

```
print(f"\nTempérature moyenne de la semaine: {moyenne:.1f}°C")

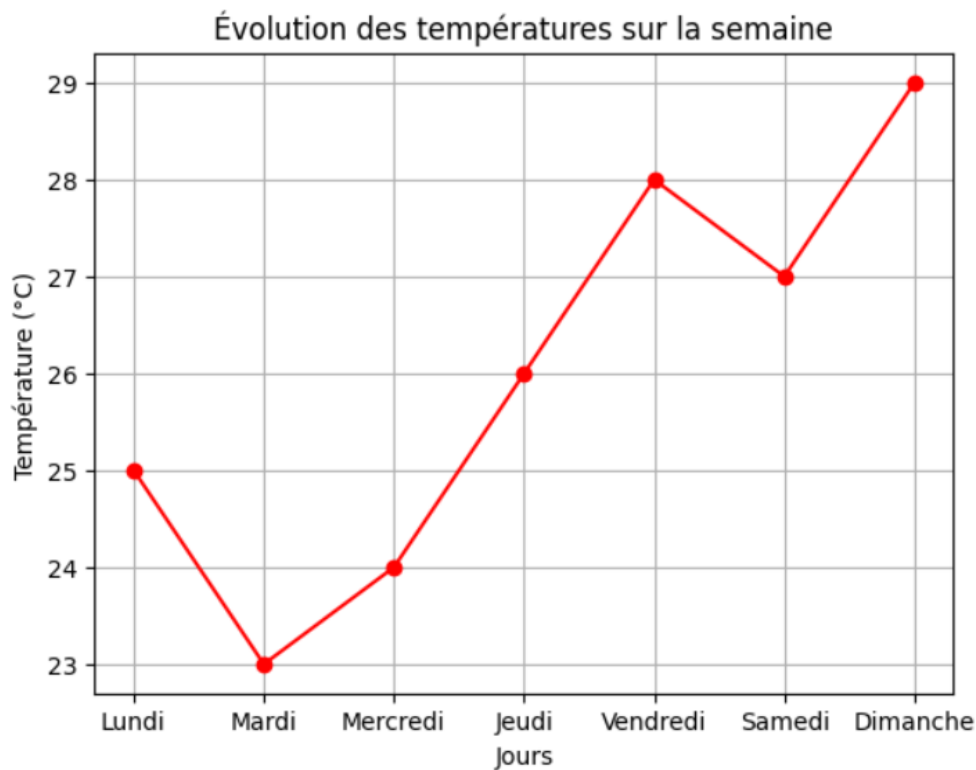
plt.plot(jours, temperatures, marker='o', color='red')
plt.title("Évolution des températures sur la semaine")
plt.xlabel("Jours")
plt.ylabel("Température (°C)")
plt.grid(True)
plt.show()

exercice_temperatures()
```

Entrez la température pour Lundi (en °C): 25
Entrez la température pour Mardi (en °C): 23
Entrez la température pour Mercredi (en °C): 24
Entrez la température pour Jeudi (en °C): 26
Entrez la température pour Vendredi (en °C): 28
Entrez la température pour Samedi (en °C): 27
Entrez la température pour Dimanche (en °C): 29

colab.research.google.com

Température moyenne de la semaine: 26.0°C



Exercice 3 : Simulation d'un Lancer de Pièces et Enregistrement en JSON

◆ Objectif :

Simuler un certain nombre de lancers de pièces (pile ou face).

Enregistrer les résultats dans un fichier JSON.

Afficher le nombre de fois où on obtient "Pile" et "Face".

```
import random
import json

def exercice_pieces():
    nb_lancers = int(input("Combien de fois voulez-vous lancer la pièce? "))

    resultats = {"Pile": 0, "Face": 0}

    for _ in range(nb_lancers):
        resultat = random.choice(["Pile", "Face"])
        resultats[resultat] += 1

    # Enregistrement en JSON
    with open("resultats_pieces.json", "w") as f:
        json.dump(resultats, f)

    print("\nRésultats des lancers:")
    print(f"Pile: {resultats['Pile']} fois")
    print(f"Face: {resultats['Face']} fois")

exercice_pieces()
```

```

import random
import json

def exercice_pieces():
    nb_lancers = int(input("Combien de fois voulez-vous lancer la pièce? "))

    resultats = {"Pile": 0, "Face": 0}

    for _ in range(nb_lancers):
        resultat = random.choice(["Pile", "Face"])
        resultats[resultat] += 1

    # Enregistrement en JSON
    with open("resultats_pieces.json", "w") as f:
        json.dump(resultats, f)

    print("\nRésultats des lancers:")
    print(f"Pile: {resultats['Pile']} fois")
    print(f"Face: {resultats['Face']} fois")

exercice_pieces()

```

Combien de fois voulez-vous lancer la pièce? 4

Résultats des lancers:
Pile: 1 fois
Face: 3 fois

Tableau

Exercice 4 : Calcul et Comparaison des Volumes de Solides

◆ Objectif :

Demander à l'utilisateur de choisir parmi trois solides :

Sphère → $V = \frac{4}{3}\pi r^3$

Prisme Rectangulaire → $V = L \times l \times h$

Cône → $V = \frac{1}{3}\pi r^2 h$

Calculer et afficher le volume du solide choisi.

Afficher un graphe en barres comparant les volumes des solides.

```

import matplotlib.pyplot as plt
import math

def exercice_volumes():
    print("Choisissez un solide:")
    print("1. Sphère")
    print("2. Prisme Rectangulaire")
    print("3. Cône")

    choix = int(input("Votre choix (1-3): "))

    volumes = []
    labels = []

```

```

if choix == 1:
    r = float(input("Rayon de la sphère: "))
    volume = (4/3) * math.pi * r**3
    print(f"Volume de la sphère: {volume:.2f}")
    volumes.append(volume)
    labels.append("Sphère")
elif choix == 2:
    L = float(input("Longueur du prisme: "))
    l = float(input("Largeur du prisme: "))
    h = float(input("Hauteur du prisme: "))
    volume = L * l * h
    print(f"Volume du prisme: {volume:.2f}")
    volumes.append(volume)
    labels.append("Prisme")
elif choix == 3:
    r = float(input("Rayon du cône: "))
    h = float(input("Hauteur du cône: "))
    volume = (1/3) * math.pi * r**2 * h
    print(f"Volume du cône: {volume:.2f}")
    volumes.append(volume)
    labels.append("Cône")

# Comparaison avec les autres solides (valeurs par défaut pour l'exemple)
if choix != 1:
    volumes.insert(0, (4/3)*math.pi*2**3) # Sphère r=2
    labels.insert(0, "Sphère")
if choix != 2:
    volumes.insert(1, 3*4*5) # Prisme 3x4x5
    labels.insert(1, "Prisme")
if choix != 3:
    volumes.insert(2, (1/3)*math.pi*2**2*5) # Cône r=2, h=5
    labels.insert(2, "Cône")

plt.bar(labels, volumes)
plt.title("Comparaison des volumes des solides")
plt.ylabel("Volume")
plt.show()

exercice_volumes()

```

← → ↺ colab.research.google.com/drive/13fLE5oKKt036tiDQerJlq_I98o5dQNL5

Exo 3_TD2.ipynb ☆ ☁

Fichier Modifier Affichage Insérer Exécution Outils Aide

Q Commandes + Code + Texte ▶ Tout exécuter ▼

```
import random
import json

def exercice_pieces():
    nb_lancers = int(input("Combien de fois voulez-vous lancer la pièce? "))
    resultats = {"Pile": 0, "Face": 0}
    for _ in range(nb_lancers):
        resultat = random.choice(["Pile", "Face"])
        resultats[resultat] += 1
    # Enregistrement en JSON
    with open("resultats_pieces.json", "w") as f:
        json.dump(resultats, f)
    print("\nRésultats des lancers:")
    print(f"Pile: {resultats['Pile']} fois")
    print(f"Face: {resultats['Face']} fois")
exercice_pieces()
```

↗ Combien de fois voulez-vous lancer la pièce? 4

Résultats des lancers:
Pile: 1 fois
Face: 3 fois

{ } Variables

Upcoming Earnings Search

Exercice 5 : Périmètre et Surface de Figures Géométriques

🎯 Objectif :

Demander à l'utilisateur d'entrer les températures journalières d'une semaine.

Calculer la température moyenne.

Afficher un graphe en ligne montrant l'évolution des températures.

Simuler un certain nombre de lancers de pièces (pile ou face).

Enregistrer les résultats dans un fichier JSON.

Afficher le nombre de fois où on obtient "Pile" et "Face".

Demander à l'utilisateur de choisir parmi trois solides :

Sphère → $V = \frac{4}{3}\pi r^3$

Prisme Rectangulaire → $V = L \times l \times h$

Cône → $V = \frac{1}{3}\pi r^2 h$

Calculer et afficher le volume du solide choisi.

Afficher un graphe en barres comparant les volumes des solides.

Demander à l'utilisateur de choisir parmi trois figures :

Carré → Périmètre : $P = 4c$, Surface : $S = c^2$

Losange → Périmètre : $P = 4c$, Surface : $S = D \times d / 2$

Trapèze → Périmètre : $P = a + b + c + d$, Surface : $S = (B + b) \times h / 2$

Calculer et afficher le périmètre et la surface de la figure choisie.

Afficher un graphe comparant les surfaces des figures.


```

import matplotlib.pyplot as plt

def exercice_figures():
    print("Choisissez une figure géométrique:")
    print("1. Carré")
    print("2. Losange")
    print("3. Trapèze")

    choix = int(input("Votre choix (1-3): "))

    surfaces = []
    labels = []

    if choix == 1:
        c = float(input("Côté du carré: "))
        perimetre = 4 * c
        surface = c * c
        print(f"Périmètre: {perimetre:.2f}")
        print(f"Surface: {surface:.2f}")
        surfaces.append(surface)
        labels.append("Carré")
    elif choix == 2:
        c = float(input("Côté du losange: "))
        D = float(input("Grande diagonale: "))
        d = float(input("Petite diagonale: "))
        perimetre = 4 * c
        surface = D * d / 2
        print(f"Périmètre: {perimetre:.2f}")
        print(f"Surface: {surface:.2f}")
        surfaces.append(surface)
        labels.append("Losange")
    elif choix == 3:
        a = float(input("Côté a: "))
        b = float(input("Côté b: "))
        c = float(input("Côté c: "))
        d = float(input("Côté d: "))
        B = float(input("Base grande B: "))
        b_petit = float(input("Base petite b: "))
        h = float(input("Hauteur h: "))
        perimetre = a + b + c + d
        surface = (B + b_petit) * h / 2
        print(f"Périmètre: {perimetre:.2f}")
        print(f"Surface: {surface:.2f}")
        surfaces.append(surface)
        labels.append("Trapèze")

    # Comparaison avec les autres figures (valeurs par défaut pour l'exemple)
    if choix != 1:

```

```

surfaces.insert(0, 5*5) # Carré c=5
labels.insert(0, "Carré")
if choix != 2:
    surfaces.insert(1, 6*4/2) # Losange D=6, d=4
    labels.insert(1, "Losange")
if choix != 3:
    surfaces.insert(2, (5+3)*4/2) # Trapèze B=5, b=3, h=4
    labels.insert(2, "Trapèze")

plt.bar(labels, surfaces)
plt.title("Comparaison des surfaces des figures")
plt.ylabel("Surface")
plt.show()

```

exercice_figures()

Choisissez un solide:

1. Sphère
2. Prisme Rectangulaire
3. Cône

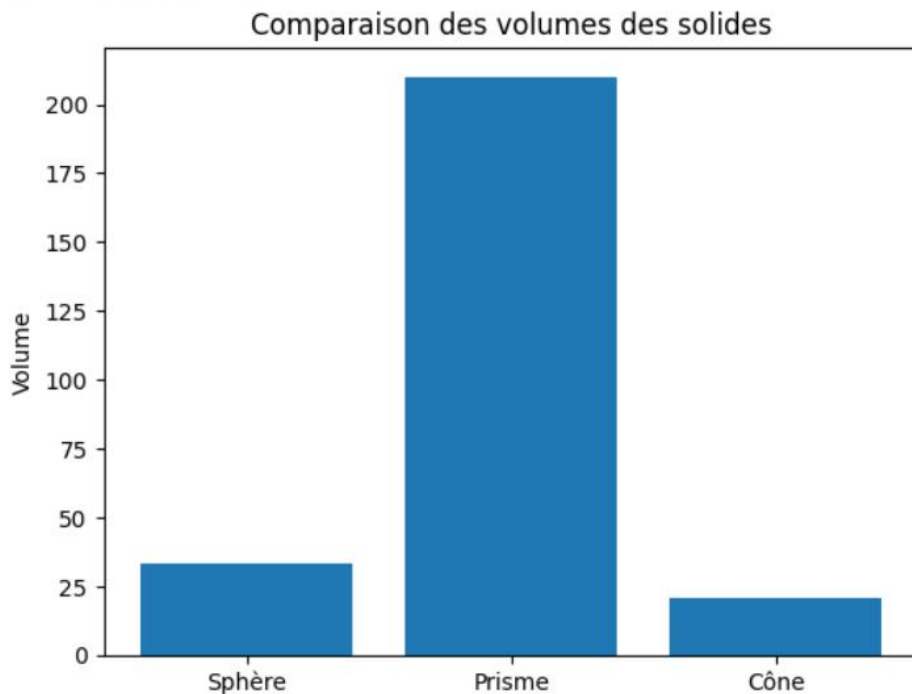
Votre choix (1-3): 2

Longueur du prisme: 5

Largeur du prisme: 6

Hauteur du prisme: 7

Volume du prisme: 210.00



Conclusion

Ce TP m'a permis de développer mes compétences en programmation Python et en analyse de données à travers plusieurs exercices pratiques. Voici un bilan des apprentissages et des résultats obtenus :

Bilan des compétences acquises

Manipulation de données : Saisie, traitement et analyse de données numériques.

Visualisation graphique : Création de graphiques linéaires et diagrammes en barres avec Matplotlib.

Gestion de fichiers : Sauvegarde de résultats dans un fichier JSON.

Calculs mathématiques : Application de formules géométriques et statistiques. 3D, etc.).

Ce TP m'a permis de consolider mes bases en Python et de mieux comprendre l'importance de la visualisation des données pour interpréter des résultats.