

- TD1**

- Phrase inversée mot par mot (sans `split()` inversé directement).
- 3. Vérifie si la phrase est un **pangramme** (contient toutes les lettres de l'alphabet).

#### **Exercice 4 :**

1. Saisir deux ensembles de nombres (par exemple A et B).
2. Afficher :
  - Éléments communs (intersection),
  - Éléments uniques à A et à B,
  - Éléments pairs dans l'union des deux ensembles,
  - Éléments présents dans un seul ensemble.
3. Construire un nouvel ensemble contenant les **carrés** de tous les nombres présents dans  $A \cup B$ .

#### **Exercice 5 :**

Écrire un programme qui :

1. Permet à l'utilisateur de saisir plusieurs paires nom : note.
2. Construit un dictionnaire associant chaque nom à sa note.
3. Affiche :
  - Moyenne des notes,
  - Le ou les étudiants ayant la meilleure note,
  - Liste des étudiants triée par note décroissante (sans utiliser `sorted()` directement).
4. Supprime toutes les entrées dont la note est inférieure à la moyenne.

#### **Exercice 6 :**

Créer un petit programme qui :

1. Contient une liste de tuples :  

```
etudiants = [("Ali", 15), ("Sara", 18), ("Nadia", 12), ("Youssef", 17)]
```
2. Construit :
  - Dictionnaire notes où la clé est le nom et la valeur la note,
  - Ensemble contenant les étudiants ayant obtenu une note  $\geq 15$ ,
  - Liste triée des noms selon les notes décroissantes (sans utiliser `sorted()`).
3. Affiche les structures résultantes.