

Exercice : Calculatrice CLI (Enoncé)

Objectif

Créer une petite application en ligne de commande (CLI) qui joue le rôle de calculatrice. L'application doit permettre d'effectuer les opérations de base ($+$, $-$, \times , \div), gérer les erreurs, proposer un mode interactif, et (bonus) sauvegarder l'historique des opérations dans un fichier JSON.

Étapes

1. Initialiser le projet

- Créer un dossier : tp_calc_cli.
- Créer les fichiers suivants :
 - calc.py → logique des opérations.
 - calc_cli.py → interface en ligne de commande.
 - test_calc.py → tests unitaires.
 - calc_storage.py (bonus) → persistance de l'historique.
 - README.md → documentation.

2. Implémenter la logique (calc.py)

- Créer des fonctions :
 - add(a, b)
 - sub(a, b)
 - mul(a, b)
 - div(a, b) (⚠ gérer la division par zéro).
- Ajouter des docstrings avec exemples d'utilisation.

3. Interface CLI (calc_cli.py)

- Utiliser argparse pour créer une interface avec les commandes :
 - add a b
 - sub a b
 - mul a b
 - div a b
- Afficher le résultat avec 4 décimales.
- Ajouter un mode interactif (interactive) où l'utilisateur peut entrer des opérations en boucle.
- Gérer les erreurs (inputs invalides, division par zéro).

4. Tests unitaires (`test_calc.py`)

- Utiliser pytest pour tester chaque fonction.
- Cas à couvrir :
 - Opérations simples.
 - Nombres négatifs et flottants.
 - Division par zéro (doit lever une erreur).
 - Entrées sous forme de chaînes convertibles ("2", "3.5").

Sur Windows, exécuter les tests avec :

bash

```
py -m pip install pytest
```

```
py -m pytest -q
```

5. Persistance (bonus) (`calc_storage.py`)

- Sauvegarder chaque opération réussie dans `calc_history.json`.
- Fonctions :
 - `append_operation(op, a, b, result, timestamp)`
 - `load_history()`
- Ajouter une commande CLI `history` pour afficher les dernières opérations.

6. Documentation (`README.md`)

- Expliquer comment installer et exécuter le projet.
- Donner des exemples de commandes.
- Mentionner les bonnes pratiques pour utiliser Copilot (intention + contraintes + exemples).