

## Bonnes pratiques (PEP8, Black, annotation de types)

34.1 : Rechercher avec `pip3` si les paquets `flake8` et `autopep8` existent. Installez-les.

34.2 : Utilisez `flake8` sur un code que vous avez écrit récemment (disons d'au moins 30 ou 40 lignes !). Étudiez les erreurs et warnings rapportées par flake, et essayer les corriger manuellement. Si certains warnings vous semblent trop agressif, utiliser `--ignore` pour spécifier des codes d'erreurs à ignorer.

34.3 : Sur un autre code relativement mal formaté, utiliser `black` pour tenter d'ajuster automatiquement le formatage du code. Sauvegarder la sortie fournie par `black` dans un autre fichier "version 2" et comparer le fichier initial avec le fichier de sortie à l'aide de `diff` ou de `git diff --no-index file1 file2`.

34.4 : Le nouveau fichier est-il exempt de problèmes d'après flake8 ?

35 : Sur votre code, ajouter quelques annotations de types sur vos variables et vos fonctions. Lancez `mypy` sur votre fichier de code pour confirmer que tout est cohérent.

## Debugging avec `pdb`

36 : Récupérer le fichier `to_debug.py` auprès du formateur. Tenter d'exécutez ce script et de le faire fonctionner en résolvant les problèmes. En particulier, ajoutez des breakpoints `pdb` (ou `ipdb`) pour étudiez ce qu'il se passe.