

sortie discussions et commentaire avec Claude

version 0.03

- https://turing.cs.hbg.psu.edu/LOCAL/ucblog/ucbl_7.htm
- <https://fmslogo.sourceforge.io/manual/workspace-control.html>
- <https://www.calormen.com/jslogo/language.html>

● Le fichier PyLogo_0.03.py est créé et prêt. Voici le résumé :

Modifications apportées

1. En-tête de version (lignes 11-28)

Historique des versions ajouté en début de fichier.

2. Classe Workspace (lignes 447-639)

Nouvelle classe pour gérer les procédures :

- define() / get() / exists() / erase()
- bury() / unbury() / is_buried()
- list_visible() / list_buried() / list_all()
- get_source() - reconstruit le code source d'une procédure
- save() - sauvegarde dans un fichier .logo

3. Nouvelles commandes Logo

Commande	Alias	Description
LISTEPROCEDURES	PROCEDURES, POCS	Liste les procédures
MONTREPROCEDURE	PO, PRINTOUT	Affiche le code d'une procédure
ENTERREPROCEDURE	BURY	Protège une procédure
DETERREPROCEDURE	UNBURY	Déprotège une procédure
SAUVEPROCEDURES	SAVE	Sauvegarde dans un fichier
CHARGEPROCEDURES	LOAD	Charge depuis un fichier

4. Intégration

- LogoInterpreter utilise maintenant self.workspace au lieu de self.procedures
- Méthode load_procedures() ajoutée pour charger les fichiers

La syntaxe est validée. Tu peux tester avec des commandes comme :

```
POUR CARRE :taille  
  REPETE 4 [AVANCE :taille DROITE 90]  
FIN  
SAUVEPROCEDURES "mes_dessins  
LISTEPROCEDURES
```

Brewed for 7m 7s

➤ C'est un profond remaniement du code qui va permettre d'être plus solide et cohérent pour la suite.

Je testerais, mais la lecture du code généré est satisfaisante !

- Merci. Effectivement, cette refactorisation apporte plusieurs avantages :

Architecture plus propre :

- Séparation des responsabilités : Workspace gère les procédures, LogoInterpreter exécute
- Encapsulation : toute la logique de gestion des procédures est centralisée
- Code plus testable unitairement

Préparation pour la suite :

- Ajout facile de nouvelles fonctionnalités (bibliothèques, modules)
- Base solide pour l'IDE enfants
- Le système BURY/UNBURY permet de protéger des procédures "système" que tu pourrais fournir

Points à tester en priorité :

1. POUR ... FIN → définition de procédure
2. LISTEPROCEDURES → vérifier qu'elle apparaît
3. SAUVEPROCEDURES "test → vérifier le fichier généré
4. CHARGEPROCEDURES "test → après un redémarrage
5. ENTERREPROCEDURE / DETERREPROCEDURE → protection

N'hésite pas à revenir si tu rencontres des soucis lors des tests ou pour continuer sur d'autres aspects du projet.

