

JEU D' ESSAI

1. Étapes préalables (Avant de lancer le code)

Avant de compiler, vous devez vous assurer que votre environnement est prêt :

1. **Organisation des fichiers** : Vérifiez que vos trois fichiers (`typeEmplacement.h`, `typeEmplacement.c` et le `main.c`,) sont dans le même dossier.
2. **Compilation** : Compile-les ensemble. Si vous utilisez un terminal : `gcc main.c typeEmplacement.c -o camping`
3. **Nettoyage** : S' il existe déjà un fichier nommé `types.dat` dans votre dossier, supprimez-le pour partir sur une base neuve (le programme le recréera proprement).

2. Le scénario du jeu d' essai

Voici les étapes à suivre pour tester la solidité de notre module actuel.

Étape A : Premier lancement (Fichier vide)

- **Action** : Lancez le programme.
- **Résultat attendu** : Le programme doit afficher « Info : Fichier `types.dat` non trouve » (grâce à notre `if (f != NULL)`). C' est normal, c' est la première fois.
- **Vérification** : Choisissez l' option 1 (Afficher les tarifs). La liste doit être vide.

Étape B : Remplissage manuel (Optionnel)

- *Note : Actuellement, nous n' avons pas encore codé de menu « Ajouter un type » dans le main (nous l' avons prévu en bibliothèque).*
- **Action :** Pour le test, nous allons ajouter une petite fonction temporaire dans le main ou modifier le fichier manuellement pour créer les 4 types de l' énoncé (Caravane, Tente, Camping-car, Bungalow).

Étape C : Persistance des données (Le test le plus important)

1. **Action :** Lancez le programme et ajoutez des données.
 2. **Action :** Quittez le programme en choisissant l' option 0.
 3. **Vérification :** Regardez dans votre dossier, un fichier doit être apparu.types.dat
 4. **Action :** Relancez le programme.
 5. **Résultat attendu :** Choisissez l' option 1. Les tarifs saisis précédemment doivent apparaître par magie ! Cela prouve que votre « Chargement au démarrage » fonctionne.
-

3. Ce que nous devons tester (Les cas limites)

Un bon programmeur teste aussi ce qui pourrait casser le code :

- **Le dépassement de capacité :** Qu' arrive-t-il si on essaie de charger 11 types alors que MAX_TYPESwhile (nbTypes < MAX_TYPES) est à 10 ? (Notre boucle doit empêcher le plantage).
 - **Le code inexistant :** Tester la fonction trouverType avec un code qui n' existe pas. Elle doit bien retourner -1.
-

4. Données de test (Basées sur l' énoncé)

Prépare ce petit tableau pour remplir tes données lors du test :

Code	Nom Type	Prix / jour / pers
1	Caravane	13.50
2	Tente	11.00
3	Voiture de camping	14.00
4	Bungalow	17.50