Mini-Projet
Gestion de stock d'un magasin

On propose de faire la gestion du stock d'un magasin de vente de Produits. Un Produit a les caractéristiques suivantes :

Attribut	Type
id	entier
Nom	Chaine de caractère
Typ	***Type***
Date expiration	***MaDate***

Avec :

Type Type	
Attribut	**Type**
idType	entier
NomType	Chaine de caractère
Cat	**Categorie**

Type MaDate	
Attribut	**Type**
JJ	Entier entre 1 et 31
MM	Entier entre 1 et 12
AA	Entier de 4 chiffres

Type Categorie	
Attribut	**Type**
idCat	entier
NomCat	Chaine de caractère

I. Préparation des données/fonctions de base :

1. Définir les types : **Categorie, Type, MaDate** et **Produit**
2. Définir les fonctions suivantes :
   a. **InitCat (Categorie*)** : permet d'initialiser les information d'une catégorie.
   b. **InitType (Type*)** : permet d'initialiser les information d'un Type.
   c. **InitDate (MaDate*)** : permet d'initialiser les informations d'une date. Les contrôles de saisie sont à effectuer (JJ est un entier entre 1 et 31, MM est un entier entre 1 et 12, AA est un entier de 4 chiffres).
   d. **InitPdt (Produit*)** : permet d'initialiser les informations d'un produit. L'utilisation des fonctions *InitType* et *InitDate* est obligatoire.

II. Préparation du menu

Au démarrage, l'application doit proposer à l'utilisateur un Menu principal pour faire un choix.

Après sélection de choix, un deuxième menu sera affiché selon le choix effectué.

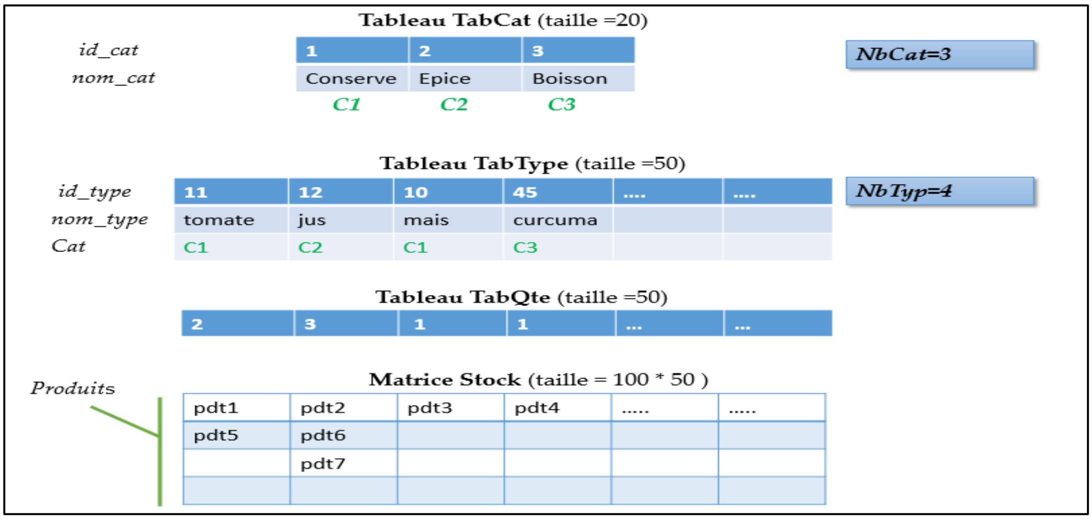
- Les contrôles sont à effectuer.
- L'utilisateur peut quitter à tout moment en choisissant **0**
- Après chaque opération le menu principal sera affiché à nouveau jusqu'à ce que l'utilisateur choisi de quitter l'application (**0**).

```
************************************
********        Bienvenue     **********
************************************
* Veuillez Choisir une option:       *
*  1- Gestion du Stock               *
*  2- Vente et  Statistiques         *
* >> Tapez votre Choix:              *
```

III. Gestion du stock

Pour la gestion de stock, on utilise les données suivantes :

- Un tableau **TabCat**, de taille 20, qui contient la liste des Catégories des produits qui peuvent exister dans le magasin.
- Un tableau **TabType**, de taille 50, qui contient la liste des types des produits associés aux différentes catégories.
- Un tableau **TabQte**, de taille 50, qui contient la quantité de produits associée à chaque type. TabQte[i] contient le nombre de produits dont le type est TabType[i].
- Une matrice **Stock,** de taille 100*50, qui contient la liste des produits disponibles dans le stock organisé par Type. Les produits de la $i^{ème}$ colonne ont comme type TabType[i].
- La quantité maximale de chaque produit dans le stock est égale à 100
- La quantité minimale de chaque produit dans le stock est égale à 10
- La variable **NbCat** contient le nombre de Catégories disponibles dans TabCat
- La variable **NbTyp** contient le nombre de Types disponibles dans TabTyp



1. Fonctions à définir

a. **Afficher stock :** permet d'afficher l'état actuel du stock. Un exemple d'affichage est représenté dans la figure suivante.

```
*********************************************
* Etat du stock:                            *
*                                           *
* Categorie Type    Qte id-Pdt  date_exp    *
* ------------------------------------------*
* Conserve  Tomate   2     50   12/12/2021  *
*                          48   14/11/2022  *
*           Mais     1     27   12/12/2021  *
* Boisson   Jus      3     33   25/10/2021  *
*                          30   13/12/2020  *
*                          60   10/11/2021  *
* Epice     Curcuma  1     11   12/12/2021  *
* ---------------------------------------   *
```

b. **AjouterCat** : Permet d'ajouter une nouvelle catégorie à **TabCat :**
   - L'ajout n'est effectué que si NbCat <20 et que la nouvelle Catégorie n'existe pas dans TabCat
   - NbCat doit être mise à jour après ajout
c. **AjouterTyp**: Permet d'ajouter un nouveau Type à **TabType :**
   - L'ajout n'est effectué que si NbTyp <50 et que la catégorie du nouveau type existe dans TabCat
   - Le nouveau Type à ajouter ne doit pas exister dans TabType
   - NbTyp doit être mise à jour après ajout
d. **AjouterPdt**: Permet d'ajouter un nouveau Produit à **Stock :**
   - L'ajout doit être effectué dans la bonne colonne (correspondant au Type du produit)
   - L'ajout n'est effectué que si la quantité des produits du Type correspondant est <100 et que le produit n'existe pas déjà dans la matrice Stock
   - Le tableau TabQte doit être mis à jour après chaque ajout d'un nouveau produit
e. **SupprimerCat** : Permet de supprimer une catégorie C de **TabCat :**
   - La suppression n'est effectuée que si le tableau TabType ne contient des types dont la catégorie est C.
   - NbCat doit être mise à jour après suppression
   - Le tableau TabCat doit être mis à jour après suppression : faire un décalage des éléments qui se trouve après la case supprimée
f. **SupprimerTyp** : Permet de supprimer une catégorie T de TabType **:**
   - La suppression n'est effectuée que si la matrice Stock ne contient pas des produits dont le type est T.
   - NbTyp doit être mise à jour après suppression
   - Le tableau TabType doit être mis à jour après suppression : faire un décalage des éléments qui se trouve après la case supprimée
g. **SupprimerPdt**: Permet de supprimer un Produit de **Stock :**
   - La suppression n'est effectuée que si la quantité des produits du Type correspondant est >10
   - Le tableau TabQte doit être mis à jour après chaque suppression d'un produit
   - La matrice Stock doit être mise à jour après suppression : faire un décalage des éléments de la même colonne.

IV. Vente et statistiques

On demande de définir les fonctions suivantes :

a. **VendrePdt** : Permet de vendre un produit P dont le type T est donné.
   - Le produit à vendre est celui ayant la date d'expiration minimale et ayant comme type T
   - Le produit vendu sera supprimé de la matrice Stock
   - Les informations du produit vendu seront stockées dans un fichier « Trace.txt » sous la forme suivante :

idPdt NomPdt idType NomType idCat NomCat date_de_vente

- ✓ Les informations d'un même produit sont stockées dans un même ligne
- ✓ Les différentes informations d'un produit sont séparées par des espaces
- ✓ La date de vente est stockée sous la forme JJ/MM/AAAA

b. **StatMois :** Etant donnée un mois M et une année A, la fonction permet de faire des statistiques sur les ventes effectuées durant le mois M de l'année A en affichant les informations suivantes :

- ✓ Nombre de ventes total
- ✓ Nombre de vente par catégorie
- ✓ Nombre de vente par année

c. **StatAnnee :** Etant donnée une année A, la fonction permet de faire des statistiques sur les ventes effectuées durant l'année A en affichant les informations suivantes :

- ✓ Nombre de ventes total
- ✓ Nombre de vente par catégorie
- ✓ Nombre de vente par année