

PROTOTYPE 2022

Épreuve pratique d'informatique

Section : Économie et Gestion

Durée : 1 heure

Coefficient : 0.5

Important : Dans le répertoire **Bac2022** situé sur la racine du disque **C** de votre poste, créez un dossier de travail portant votre numéro d'inscription (6 chiffres) et dans lequel vous devez enregistrer, au fur et à mesure, tous les fichiers solutions de ce sujet.

Afin de gérer le stockage des céréales collectées dans des silos, on se propose d'utiliser la base de données intitulée "**Gestion_Cereale**" décrite par le schéma textuel simplifié suivant :

Cereale (CodeC, NomC, Prix)

Silo (CodeS, Nom, Adresse, Capacité)

Collecte (CodeC#, CodeS#, DateC, Quantité)

Soit la description des colonnes des tables de la base de données "**Gestion_Cereale**" :

Nom	Description	Type	Taille	Format	Contrainte
CodeC	Le code d'une céréale	Texte court	4		Non vide
NomC	Le nom d'une céréale	Texte court	30		Non vide
Prix	Le prix d'un Quintal d'une céréale en Dinar	Numérique	Réel	Nombre général (3 chiffres après la virgule)	Supérieur à zéro
CodeS	Le code d'un silo	Texte court	4		Non vide
Nom	Le nom d'un silo	Texte court	30		Non vide
Adresse	L'adresse d'un silo	Texte court	50		
Capacité	La capacité maximale de stockage d'un silo en Quintal.	Numérique	Entier long		Supérieure à zéro
DateC	La date d'une collecte	Date/Heure		Date abrégé	Non vide
Quantité	La quantité, en Quintal, livrée à un silo à une date bien déterminée	Numérique	Entier		Supérieure à zéro

A) À l'aide du logiciel de gestion de base de données disponible :

- Créer, dans votre dossier de travail, la base de données "**Gestion_Cereale**".
- Créer les tables et les relations de cette base de données tout en respectant les types et les propriétés cités dans la description ci-dessus.
- Remplir les tables par les données suivantes :

Cereale		
CodeC	NomC	Prix
C01	Blé Dur	87.255
C02	Blé Tendre	67.732
C03	Orge	56.500
C04	Triticale	56.500

Silo			
CodeS	Nom	Adresse	Capacité
S01	STBlé	Jendouba	255000
S02	CerNord	Beja	270000
S03	ColKef	Le Kef	189987
S04	StockBlé	Bizerte	265000

Collecte			
CodeC	CodeS	DateC	Quantité
C01	S01	29/06/2021	4200
C02	S02	29/06/2021	4500
C03	S03	30/06/2021	3500
C04	S04	30/06/2021	3900
C01	S02	16/07/2021	4900
C02	S01	28/07/2021	6230



4) Créer les requêtes suivantes :

R1 : Afficher les noms et les quantités des céréales collectées le **30/06/2021**.

R2 : Afficher la liste des silos (**Nom, Adresse, Capacité**) qui ont stocké une céréale de code donné.

R3 : Afficher la somme des quantités collectées dans le silo de code **"S01"**.

R4 : Afficher la liste des céréales collectées (**CodeC, Prix, CodeS, Nom, Quantite, DateC**).

5) Exporter, dans votre dossier de travail et sous le nom de **"Collecte.csv"**, le contenu de la table **"Collecte"** au format **csv** (virgule en séparateur de champs, point en séparateur décimal, page de codes : Unicode (UTF-8) et inclure les noms des champs sur la première ligne).

B) À l'aide de l'environnement de développement pour le langage Python disponible :

1) Créer un script et l'enregistrer sous le nom **"Collecte.py"** dans votre dossier de travail.

2) En se référant à l'annexe de la page 3, choisir les méthodes adéquates, de la bibliothèque Pandas, pour écrire le script **"Collecte.py"** qui permet :

- d'importer le contenu du fichier **"Collecte.csv"** et le stocker dans un DataFrame à nommer **"Collecte"**,
- d'afficher les informations relatives au DataFrame **"Collecte"** ainsi que les 6 premières lignes,
- de remplacer la quantité de la 5^{ème} ligne, dans le DataFrame **"Collecte"**, par 5560 au lieu de 3500,
- d'afficher les informations relatives aux céréales dont la quantité collectée est supérieure ou égale à 4500,
- de trier le DataFrame **"Collecte"** dans un nouveau DataFrame nommé **"Collecte_Tri"** selon la quantité collectée dans l'ordre croissant. Afficher le nouveau DataFrame,
- de calculer et afficher la moyenne des quantités collectées,
- de générer puis afficher, à partir du DataFrame **"Collecte"**, un graphique à barres rouges représentant, pour chaque date de collecte, la quantité collectée.



Grille d'évaluation	Partie	A (12 points)					B (8 points)							
	Question	1	2	3	4	5	1	2-a	2-b	2-c	2-d	2-e	2-f	2-g
	Note	1	4	1.5	4	1.5	1	1	1	1	1	1	1	1

ANNEXE

Les méthodes à utiliser avec la bibliothèque pandas

Catégorie	Syntaxe
Création et importation d'un DataFrame	<pre>Id_DataFrame = pandas.read_excel ("Chemin/ Nom_Fichier.extension", "Nom_Feuille") Id_DataFrame = pandas.read_csv ("Chemin/Nom_Fichier.extension", sep = "séparateur")</pre>
Manipulation d'un DataFrame	<pre>Id_DataFrame.shape ou bien print (Id_DataFrame.shape) Id_DataFrame.size ou bien print (Id_DataFrame.size) Id_DataFrame.info () ou bien print (Id_DataFrame.info ()) Id_DataFrame.describe () ou bien print (Id_DataFrame.describe ()) Id_DataFrame.columns ou bien print (Id_DataFrame.columns) Id_DataFrame.rename (columns = {"colonne1":"nouveau_nom_colonne1", ..., "colonneN":"nouveau_nom_colonneN"}, inplace = True/False) Id_DataFrame.drop ([Indice_Ligne] , inplace = True/False) Id_DataFrame.drop (["Nom_Colonne1",..., "Nom_ColonneN"] , inplace = True/False)</pre>
Affichage des données d'un DataFrame	<pre>Id_DataFrame ou bien print (Id_DataFrame) Id_DataFrame ["Id_Colonne"] ou bien print (Id_DataFrame ["Id_Colonne"]) Id_DataFrame.head (n) ou bien print (Id_DataFrame.head (n)) Id_DataFrame.tail (n) ou bien print (Id_DataFrame.tail (n)) Id_DataFrame.Id_Colonne [Position_Initiale : Position_Finale] ou bien print (Id_DataFrame.Id_Colonne [Position_Initiale:Position_Finale]) Id_DataFrame.iloc [Indice_Ligne] ou bien print (Id_DataFrame.iloc [Indice_Ligne]) Id_DataFrame.iloc [Indice_Ligne,Indice_Colonne] ou bien print (Id_DataFrame.iloc [Indice_Ligne,Indice_Colonne]) Id_DataFrame.loc [N° ligne,"Nom_colonne"] = Valeur Id_DataFrame.loc [N° ligne] = [Liste_Valeur] Id_DataFrame ["Id_Colonne"] = valeur Id_DataFrame [Id_DataFrame["Nom_Colonne"] Op_Comparaison valeur] Id_DataFrame [(Id_DataFrame["Nom_Colonne"] Op_Comp Valeur) Op_Logique (Id_DataFrame["Nom_Colonne"] Op_Comp Valeur)]</pre>
Les fonctions statistiques	<pre>mean () – min () – max () – sum () – count ()</pre>
Tri des éléments d'un DataFrame	<pre>DataFrame_Trié = Id_DataFrame.sort_values (by = [liste_colonnes], ascending = [True/False])</pre>
Création d'un graphique	<pre>Id_DataFrame.plot.bar (x = "Nom_Colonne_Abscisse" , y = "Nom_Colonne_Ordonnée" , title = "Titre Graphique" , color = "couleur_Graphique") Id_DataFrame.plot.line (x = "Nom_Colonne_Abscisse" , y = "Nom_Colonne_Ordonnée" , title = "Titre Graphique" , color = "couleur_Graphique") alias_matplotlib.show ()</pre>

