

RÉPUBLIQUE TUNISIENNE *** Ministère de l'Éducation	EXAMEN DU BACCALAURÉAT	SESSION 2024
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Section : Économie et Gestion	
	Coefficient de l'épreuve : 0.5	Durée : 1h

Le sujet comporte 3 pages numérotées de 1/3 à 3/3

Important : Il est demandé au candidat de créer dans le dossier **Bac2024** situé sur la racine du disque **C**, un dossier de travail portant son numéro d'inscription (6 chiffres) et dans lequel il doit enregistrer, au fur et à mesure, tous les fichiers solutions de ce sujet.

Afin de gérer les opérations d'emprunt des livres de la bibliothèque d'un lycée, on se propose d'utiliser la base de données simplifiée intitulée "**Bibliothèque**" décrite par le schéma textuel suivant :

LIVRE (CodeLivre, Auteur, Titre, NbrExemp, Domaine)

ELEVE (IdElève, NomPrénom, Classe)

EMPRUNT (NumEmprunt, IdElève#, CodeLivre#, DateEmprunt, DateRetour)

Sachant que :

- **LIVRE** est une table contenant les informations relatives aux livres.
- **ELEVE** est une table contenant les informations relatives aux élèves.
- **EMPRUNT** est une table contenant les informations relatives aux opérations d'emprunt.

Soit la description des colonnes des tables de cette base de données :

Nom	Description	Type	Taille	Format	Contrainte
CodeLivre	Code d'un livre	Texte	8		
Auteur	Nom et prénom de l'auteur d'un livre	Texte	50		Null interdit
Titre	Titre d'un livre	Texte	50		Null interdit
NbrExemp	Nombre d'exemplaires d'un livre disponible dans la bibliothèque	Numérique	Entier		> 0
Domaine	Domaine d'un livre	Texte	20		
IdElève	Identifiant d'un élève	Texte	6		
NomPrénom	Nom et prénom d'un élève	Texte	50		Null interdit
Classe	Classe d'un élève	Texte	10		
NumEmprunt	Numéro d'un emprunt	NuméroAuto			
DateEmprunt	Date d'emprunt d'un livre	Date/Heure		Date, abrégé	Null interdit
DateRetour	Date de retour prévue d'un livre	Date/Heure		Date, abrégé	Null interdit

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

- 1) Créer, dans le dossier de travail, la base de données nommée "**Bibliothèque**".
- 2) Créer les tables et les relations relatives à cette base de données tout en respectant les types, les tailles, les formats et les contraintes cités dans la description ci-dessus.

3) Remplir les tables par les données représentées dans les tableaux suivants :

LIVRE				
CodeLivre	Auteur	Titre	NbrExemp	Domaine
LIV24001	Jason R. Briggs	Python pour les kids	15	Informatique
LIV04101	Vincent Sénétaire	La cybersécurité de zéro	10	Informatique
LIV24006	Adam Smith	La richesse des nations	5	Économie
LIV00171	Brian Greene	L'univers élégant	10	Astronomie
LIV01001	Carl Sagan	Cosmos	15	Astronomie

ELEVE		
IdElève	NomPrénom	Classe
EL0410	Samar Salhi	4 EG 1
EL0305	Kamel Yangui	3 EG 2
EL0117	Hedi Tounsi	4 EG 1
EL0518	Souha Makni	3 EG 1
EL0411	Ali Mefteh	4 EG 3

EMPRUNT				
NumEmprunt	IdElève	CodeLivre	DateEmprunt	DateRetour
1	EL0410	Liv24001	18/01/2024	30/01/2024
2	EL0305	Liv24001	18/01/2024	25/02/2024
3	EL0518	Liv24006	20/01/2024	01/02/2024
4	EL0518	Liv24006	01/02/2024	26/02/2024
5	EL0411	Liv04101	22/01/2024	27/03/2024

4) Créer les requêtes suivantes :

R1 : Augmenter de **10** le nombre d'exemplaires du livre ayant le code "**LIV04101**",

R2 : Supprimer l'élève "**Hedi Tounsi**" de la table "**ELEVE**",

R3 : Afficher, pour chaque livre d'informatique, son code, son titre ainsi que le nombre d'emprunts pendant le mois de **janvier 2024**,

5) Exporter, dans le dossier de travail et sous le nom "**livre.csv**", le contenu de la table "**LIVRE**" au format **csv** (**Point-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 disponible et en exploitant la bibliothèque Pandas :

1) Créer un fichier et l'enregistrer sous le nom "**Livre**" dans le dossier de travail.

2) Choisir les méthodes et les propriétés adéquates, à partir de **l'annexe de la page 3/3**, pour écrire dans le fichier "**Livre**" les scripts permettant de (d') :

a) importer le contenu du fichier "**livre.csv**" et le stocker dans un DataFrame à nommer "**dflivre**",

b) afficher les noms des colonnes du DataFrame "**dflivre**",

c) afficher le nombre total d'exemplaires des livres du domaine "**Astronomie**",

d) afficher la liste des livres ("**Titre**" et "**NbrExemp**") dont le nombre d'exemplaires est supérieur à **8**,

e) trier le DataFrame "**dflivre**" par ordre croissant des titres des livres puis afficher son contenu,

f) créer et afficher un graphique, en barres jaunes à partir du DataFrame "**dflivre**", représentant le nombre d'exemplaires ("**NbrExemp**") pour chaque livre ("**CodeLivre**"). Attribuer au graphique le titre "**Nombre d'exemplaires par livre**".

Grille d'évaluation	Partie	A (13 points)					B (7 points)						
	Question	1	2	3	4	5	1	2.a.	2.b.	2.c.	2.d.	2.e.	2.f.
	Note	0.5	4.5	3	4	1	0.5	1.25	0.25	1	1	1.25	1.75

ANNEXE

Les méthodes et les propriétés à utiliser avec la bibliothèque pandas

Catégorie	Syntaxe
Création d'un DataFrame	<code>IdDataFrame = pandas.DataFrame ({"NomColonne1":["Val1",..., "ValN"],..., "NomColonneM":["Val1,...,ValN"]})</code>
Importation des données dans un DataFrame	<code>IdDataFrame = pandas.read_excel ("Chemin/ Nom_Fichier.extension", "Nom_Feuille")</code> <code>IdDataFrame = pandas.read_csv ("Chemin/Nom_Fichier.extension", sep = "séparateur")</code>
Manipulation d'un DataFrame	<code>IdDataFrame.shape</code> ou bien <code>print (IdDataFrame.shape)</code> <code>IdDataFrame.size</code> ou bien <code>print (IdDataFrame.size)</code> <code>IdDataFrame.info ()</code> ou bien <code>print (IdDataFrame.info ())</code> <code>IdDataFrame.describe ()</code> ou bien <code>print (IdDataFrame.describe ())</code> <code>IdDataFrame.columns</code> ou bien <code>print (IdDataFrame.columns)</code> <code>Resultat = IdDataFrame.rename (columns = {"NomColonne1": "NouveauNomColonne1", "NomColonne2": "NouveauNomColonne2", ... })</code> <code>Resultat = IdDataFrame.drop ([N° Ligne1, ... , N° LigneN])</code> <code>Resultat = IdDataFrame.drop (IdDataFrame.index [N° LigneInitial : N° LigneFinal])</code> <code>Resultat = IdDataFrame.drop (columns = ["NomColonne1",..., "NomColonneN"])</code>
Affichage des données d'un DataFrame	<code>IdDataFrame</code> ou bien <code>print (IdDataFrame)</code> <code>IdDataFrame ["NomColonne"]</code> ou bien <code>print (IdDataFrame ["NomColonne"])</code> <code>Id_DataFrame [["NomColonne1","NomColonne2",...]]</code> ou bien <code>print (Id_DataFrame [["NomColonne1","NomColonne2",...]])</code> <code>IdDataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal]</code> ou bien <code>print (IdDataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal])</code> <code>IdDataFrame.iloc [N° Ligne]</code> ou bien <code>print (IdDataFrame.iloc [N° Ligne])</code> <code>IdDataFrame.head (n)</code> ou bien <code>print (IdDataFrame.head (n))</code> <code>IdDataFrame.tail (n)</code> ou bien <code>print (IdDataFrame.tail (n))</code> <code>IdDataFrame.iloc [N° LigneInitial : N° LigneFinal]</code> ou bien <code>print (IdDataFrame.iloc [N° LigneInitial : N° LigneFinal])</code> <code>IdDataFrame.iloc [N° Ligne , N° Colonne]</code> ou bien <code>print (IdDataFrame.iloc [N° Ligne , N° Colonne])</code>
Modification et ajout des données dans un DataFrame	<code>IdDataFrame.loc [N° ligne , "NomColonne"] = Valeur (ou Formule)</code> <code>Id_DataFrame.loc [N° ligne] = [Liste_Valeur]</code> <code>Id_DataFrame ["NomColonne"] = Valeur (ou Formule)</code>
Nettoyage d'un DataFrame	<code>Resultat = Id_DataFrame.drop_duplicates ()</code> <code>Resultat = Id_DataFrame.dropna ()</code> <code>Resultat = Id_DtaFrame.dropna (axis = 1)</code>
Les fonctions statistiques	<code>mean ()</code> – <code>min ()</code> – <code>max ()</code> – <code>sum ()</code> – <code>count ()</code>
Affichage des données d'un DataFrame selon une ou plusieurs conditions	<code>IdDataFrame [IdDataFrame ["NomColonne"] OperateurComparaison Valeur]</code> <code>IdDataFrame [(IdDataFrame ["NomColonne"] OperateurComp Valeur) OperateurLogique (IdDataFrame ["NomColonne"] OperateurComp Valeur)]</code>
Tri des éléments d'un DataFrame	<code>DataFrame_Trié = IdDataFrame.sort_values (by = ["NomColonne1" , "NomColonne2" , ...] , ascending = [True/False , True/False , ...])</code>
Création d'un graphique	<code>IdDataFrame.plot.bar (x = "NomColonneAbscisse", y = "NomColonneOrdonnée", title = "TitreGraphique", color = "CouleurGraphique")</code> <code>IdDataFrame.plot.line (x = "NomColonneAbscisse", y = "NomColonneOrdonnée", title = "TitreGraphique", color = "CouleurGraphique")</code> <code>alias_matplotlib.show ()</code>