

RÉPUBLIQUE TUNISIENNE ☆☆☆ Ministère de l'Éducation	EXAMEN DU BACCALAURÉAT	SESSION 2023
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Section : Économie et Gestion	
	Coefficient de l'épreuve : 0.5	Durée : 1h

*Le sujet comporte 3 pages numérotées de 1 sur 3 à 3 sur 3*

**Important :** Dans le répertoire **Bac2023** 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 les opérations de don de sang au niveau des différents centres de transfusion sanguine, on se propose d'utiliser la base de données simplifiée intitulée "**DonSang**" décrite par le schéma textuel suivant :

**CENTRE** (CodeCentre, NomCentre, AdresseCentre, TelCentre)

**DONNEUR** (CIN, NomPrenom, DateNais, Genre, GS, Poids, TelDonneur)

**TYPEDON** (CodeType, LibType)

**DON** (CIN#, DateDon, CodeCentre#, CodeType#, Quantite)

Sachant que :

- **CENTRE** est une table contenant les informations relatives aux différents centres de transfusion sanguine.
- **DONNEUR** est une table contenant les informations relatives aux différents donneurs.
- **TYPEDON** est une table contenant les informations relatives aux différents types de dons.
- **DON** est une table contenant les dons effectués, à savoir le centre de collecte, le donneur, la date du don, le type du don ainsi que la quantité prise.

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

Nom	Description	Type	Taille	Format	Contrainte
CodeCentre	Code d'un centre	Texte	4		
NomCentre	Nom d'un centre	Texte	100		Null interdit
AdresseCentre	Adresse d'un centre	Texte	100		
TelCentre	Numéro de téléphone d'un centre	Texte	8		
CIN	Numéro de la carte d'identité d'un donneur	Texte	8		
NomPrenom	Nom et prénom d'un donneur	Texte	50		Null interdit
DateNais	Date de naissance d'un donneur	Date/Heure		Date, abrégé	Null interdit
Genre	Genre d'un donneur (M ou F)	Texte	1		Null interdit
GS	Nom du groupe sanguin d'un donneur	Texte	3		Null interdit
Poids	Poids d'un donneur en Kg	Numérique	Entier		Null interdit et $\geq 50$
TelDonneur	Numéro de téléphone d'un donneur	Texte	8		
CodeType	Code du type d'un don	Texte	4		
LibType	Libellé du type d'un don	Texte	100		Null interdit
DateDon	Date d'un don	Date/Heure		Date, abrégé	Null interdit
Quantite	Quantité donnée en ml	Numérique	Entier		Null interdit et $400 \leq \text{Quantite} \leq 600$

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

- 1) Créer, dans votre dossier de travail, la base de données à nommer "**DonSang**".
- 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 différentes tables par les données représentées dans les tableaux suivants :

CENTRE			
CodeCentre	NomCentre	AdresseCentre	TelCentre
C001	C.N.T.S	Rue Djebel Lakhdar	71573884
C002	C.R.T.S Sfax	Rue Majida Boulila	74245462
C003	C.R.T.S Sousse	Rue Farhat Hached	73214200

DONNEUR						
CIN	NomPrenom	DateNais	Genre	GS	Poids	TelDonneur
04589765	Saoussen AYADI	03/06/1998	F	A+	65	54289745
04879652	Salma ABED	22/10/1979	F	O-	76	25687451
05589746	Samir GAZZEH	23/05/1999	M	AB+	60	95365897
05697423	Ali CHAWECH	18/12/1980	M	O-	69	51489736

TYPEDON	
CodeType	LibType
T001	Sang total
T002	Aphérèse
T003	Plasma
T004	Plaquette

DON				
CIN	DateDon	CodeCentre	CodeType	Quantite
04589765	03/01/2023	C001	T004	450
04879652	04/03/2023	C001	T001	400
05589746	18/04/2023	C002	T001	500
05697423	02/04/2023	C003	T001	600

4) Créer les requêtes suivantes :

**R1** : Afficher la liste des donneurs (**CIN, NomPrenom, TelDonneur**).

**R2** : Afficher la liste des dons (**DateDon, NomPrenom, GS, Poids, LibType, Quantite**) réalisés pendant le mois d'Avril 2023, triée par ordre croissant des dates.

**R3** : Afficher, pour chaque groupe sanguin, le nom du groupe ainsi que la quantité totale collectée.

5) Exporter, dans votre dossier de travail sous le nom "**donneur.csv**", le contenu de la table "**DONNEUR**" au format **csv** (utiliser le **Point-virgule** comme séparateur de champs, le **point** comme 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 "**donsang**" dans votre dossier de travail.

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

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

b) afficher le contenu de ce DataFrame,

c) ajouter à la fin du DataFrame "**don**", la ligne suivante :

CIN	NomPrenom	DateNais	Genre	GS	Poids	TelDonneur
10253344	Rim SALHI	13/09/2000	F	O-	51	82111666

d) afficher les noms et les prénoms (**NomPrenom**) des donneurs universels (les donneurs dont le groupe sanguin est "**O-**"),

e) trier le DataFrame "**don**" dans un nouveau DataFrame à nommer "**don\_tri**" selon l'ordre croissant des noms et des prénoms (**NomPrenom**) des donneurs puis afficher le DataFrame trié,

f) Créer un graphique en barres rouges, à partir du DataFrame "**don**", représentant, pour chaque donneur, le poids correspondant. Attribuer le titre "**Poids des donneurs**" pour le graphique.

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.
	Note	0.5	4.5	2	4	1	0.5	1.25	0.25	1	1.5	1.5	2

**ANNEXE****Les méthodes et les propriétés à utiliser avec la bibliothèque pandas**

<b>Catégorie</b>	<b>Syntaxe</b>
<b><i>Création d'un DataFrame</i></b>	<code>IdDataFrame = pandas.DataFrame ({"NomColonne1":["Val1",..., "ValN"],..., "NomColonneM":["Val1,...,ValN"]})</code>
<b><i>Importation des données dans un DataFrame</i></b>	<code>IdDataFrame = pandas.read_excel ("Chemin/ Nom_Fichier.extension", "Nom_Feuille")</code> <code>IdDataFrame = pandas.read_csv ("Chemin/Nom_Fichier.extension", sep = "séparateur")</code>
<b><i>Manipulation d'un DataFrame</i></b>	<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>
<b><i>Affichage des données d'un DataFrame</i></b>	<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>
<b><i>Modification et ajout des données dans un DataFrame</i></b>	<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>
<b><i>Nettoyage d'un DataFrame</i></b>	<code>Resultat = Id_DataFrame.drop_duplicates ()</code> <code>Resultat = Id_DataFrame.dropna ()</code> <code>Resultat = Id_DtaFrame.dropna ( axis = 1 )</code>
<b><i>Les fonctions statistiques</i></b>	<code>mean ()</code> – <code>min ()</code> – <code>max ()</code> – <code>sum ()</code> – <code>count ()</code>
<b><i>Affichage des données d'un DataFrame selon une ou plusieurs conditions</i></b>	<code>IdDataFrame [ IdDataFrame [ "NomColonne" ] OperateurComparaison Valeur ]</code> <code>IdDataFrame [( IdDataFrame [ "NomColonne" ] OperateurComp Valeur) OperateurLogique ( IdDataFrame [ "NomColonne" ] OperateurComp Valeur )]</code>
<b><i>Tri des éléments d'un DataFrame</i></b>	<code>DataFrame_Trié = IdDataFrame.sort_values ( by = [ "NomColonne1" , "NomColonne2" , ... ] , ascending = [ True/False , True/False , ... ] )</code>
<b><i>Création d'un graphique</i></b>	<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>