

RÉPUBLIQUE TUNISIENNE ☆☆☆ Ministère de l'Éducation	EXAMEN DU BACCALAURÉAT		SESSION 2025	
	ÉPREUVE PRATIQUE D'INFORMATIQUE			
	Section : Économie et Gestion			
	Coefficient de l'épreuve: 0.5		Durée: 1h	

Le sujet comporte 4 pages numérotées de 1/4 à 4/4

Important : Dans le répertoire **Bac2025** 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 résidents dans les maisons d'accueil des personnes âgées, le responsable exploite la base de données simplifiée intitulée "**GestionAgés**" décrite par la représentation textuelle suivante :

MAISON (IdMaison, NomMais, Ville, Capacite)

AUTONOMIE (IdType, LibType, CoutMens)

RESIDENT (IdRes, NomPrenRes, DateNaiss, Genre, DateAdm, IdMaison#, IdType#)

SUIVISANTE (IdRes#, DateSuivi, EtatSante)

Sachant que :

- **MAISON** est une table contenant les informations relatives aux maisons d'accueil des personnes âgées.
- **AUTONOMIE** est une table contenant les différents types d'autonomie que peut avoir un résident.
- **RESIDENT** est une table contenant les informations relatives aux différentes personnes âgées résidentes dans une maison d'accueil.
- **SUIVISANTE** est une table contenant les informations relatives aux suivis de la santé des différents résidents.

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

Nom	Description	Type	Taille	Format	Contrainte
IdMaison	Identifiant d'une maison d'accueil.	Texte court	3		
NomMais	Nom d'une maison d'accueil.	Texte court	20		
Ville	Ville où se trouve une maison d'accueil.	Texte court	20		
Capacite	Capacité d'accueil d'une maison.	Numérique	Entier		Null interdit
IdType	Identifiant d'un type d'autonomie.	Texte court	2		
LibType	Libellé d'un type d'autonomie.	Texte court	20		
CoutMens	Prix mensuel payé par un résident.	Numérique	Réel simple		3 décimales
IdRes	Identifiant d'une personne résidente dans une maison d'accueil.	Texte court	3		
NomPrenRes	Nom et prénom d'un résident.	Texte court	30		
DateNaiss	Date de naissance d'un résident.	Date/Heure		Date, abrégé	
Genre	Genre d'un résident (M : Masculin, F : Féminin).	Texte court	1		Contenant obligatoirement la lettre M ou F
DateAdm	Date d'admission d'un résident dans une maison d'accueil.	Date/Heure		Date, abrégé	
DateSuivi	Date de suivi de l'état de santé d'un résident.	Date/Heure		Date, abrégé	
EtatSante	Observation sur l'état de santé d'un résident.	Texte long			

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 "**GestionAgés**".
- 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 :

MAISON			
IdMaison	NomMais	Ville	Capacite
M01	Errahma	Sousse	45
M02	Adhiafa	Nabeul	38
M03	Al Amal	Gafsa	40
M04	Saaidni	Tunis	55

AUTONOMIE		
IdType	LibType	CoutMens
A	Autonome	60
SA	Semi-autonome	100
D	Dépendant	140

SUIVISANTE		
IdRes	DateSuivi	EtatSante
R01	14/01/2025	RAS
R05	20/02/2025	Etat aggravé
R03	14/01/2025	Stationnaire
R04	15/12/2024	Stationnaire
R02	10/01/2025	A opérer
R05	08/01/2025	A hospitaliser
R04	04/03/2025	Légère amélioration
R08	25/04/2025	Etat aggravé

RESIDENT						
IdRes	NomPrenRes	DateNaiss	Genre	DateAdm	IdMaison	IdType
R01	Mannana KEFI	14/05/1954	F	01/05/2024	M01	A
R02	Khalifa SOUSSI	22/03/1948	M	14/03/2023	M02	D
R03	Farid GAFSI	05/05/1944	M	02/06/2024	M01	SA
R04	Zeineb TOUNSI	07/08/1938	F	15/05/2024	M03	D
R05	Zbaida SALHI	20/11/1941	F	20/02/2023	M04	D
R06	Ali NABLI	24/02/1952	M	11/06/2024	M03	A
R07	Aicha TOUATI	09/02/1953	F	20/02/2025	M03	SA
R08	Salah AZZOUZ	12/06/1954	M	18/02/2025	M02	SA

4) Créer les requêtes suivantes :

R1 : Afficher les identifiants, les noms et prénoms des résidents, leurs dates de naissance, leurs genres, leurs dates d'admission, leurs types d'autonomie, les couts mensuels d'admission ainsi que les identifiants de leurs maisons d'accueil triés par ordre croissant des noms et prénoms des résidents.

R2 : Augmenter de 12%, les coûts mensuels des résidents de type d'autonomie "SA".

R3 : Afficher le nombre de résidents par nom de maison d'accueil.

5) Exporter, dans votre dossier de travail et sous le nom "**Mapag.csv**", le résultat de la requête "**R1**" au format **csv** (Utiliser un **point-virgule** comme séparateur de champs, un **point** comme séparateur décimal, la 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 "**MaisonAgés**" dans votre dossier de travail.
- 2) Dans ce fichier, choisir les méthodes et les propriétés adéquates, à partir de l'annexe (page 3 et 4), pour écrire les scripts permettant :
 - a) d'importer le contenu du fichier "**Mapag.csv**" et de le stocker dans un DataFrame à nommer "**maison**",
 - b) d'afficher le nombre d'éléments du DataFrame "**maison**",
 - c) de supprimer la première colonne intitulée "**IdRes**" du DataFrame "**maison**",
 - d) d'afficher le nombre de résidents ayant le type d'autonomie "**D**",
 - e) de calculer, dans une nouvelle colonne à nommer "**MontantAnnuel**", le coût total annuel à payer par chaque résident puis d'afficher le contenu du DataFrame "**maison**",
 - f) de créer et d'afficher un graphique de type histogramme de couleur bleue, à partir du DataFrame "**maison**", représentant le montant annuel payé par résident. Attribuer le titre "**Montant annuel**" à ce graphique.

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	3.75	3	4.5	1.25	0.5	1.25	0.5	0.5	1.25	1.25	1.75

ANNEXE

Les méthodes et les propriétés à utiliser avec la bibliothèque pandas

Catégorie	Syntaxe
<i>Création d'un DataFrame</i>	▪ <code>IdDataFrame = pandas.DataFrame ({"Id_Colonne1":["Val1",...,"ValN"],..., "Id_ColonneM":[Val1,...,ValN]})</code>
<i>Importation des données dans un DataFrame</i>	▪ <code>IdDataFrame = pandas.read_excel ("Chemin/ Nom_Fichier.extension", "Nom_Feuille")</code> ▪ <code>IdDataFrame = pandas.read_csv ("Chemin/Nom_Fichier.extension", sep = "séparateur")</code>
<i>Manipulation d'un DataFrame</i>	▪ <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.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>
<i>Affichage des données d'un DataFrame</i>	▪ <code>IdDataFrame</code> ou bien <code>print (IdDataFrame)</code> ▪ <code>IdDataFrame ["NomColonne"]</code> ou bien <code>print (IdDataFrame ["NomColonne"])</code> ▪ <code>IdDataFrame [["Id_Colonne1","Id_Colonne2",...]]</code> ou bien <code>print (IdDataFrame [["Id_Colonne1","Id_Colonne2",...]])</code> ▪ <code>IdDataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal]</code> ou bien <code>print (IdDataFrame ["NomColonne"] [N° LigneInitial : N° LigneFinal])</code> ▪ <code>IdDataFrame.loc [N° Ligne]</code> ou bien <code>print (IdDataFrame.loc [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.loc [N° LigneInitial : N° LigneFinal]</code> ou bien <code>print (IdDataFrame.loc [N° LigneInitial : N° LigneFinal])</code> ▪ <code>IdDataFrame.loc [[N° Ligne1 , N° Ligne2 , ...]]</code> ou bien <code>print (IdDataFrame.loc [[N° Ligne1 , N° Ligne2 , ...]])</code> ▪ <code>IdDataFrame.loc [N° Ligne , "NomColonne"]</code> ou bien <code>print (IdDataFrame.loc [N° Ligne , "NomColonne"])</code>

Catégorie	Syntaxe
<i>Modification et ajout des données dans un DataFrame</i>	▪ <code>IdDataFrame.loc [N° ligne , "NomColonne"] = Valeur (ou Formule)</code> ▪ <code>IdDataFrame.loc [N° ligne] = [Liste_Valeur]</code> ▪ <code>IdDataFrame ["Id_Colonne"] = Valeur (ou Formule)</code>
<i>Nettoyage d'un DataFrame</i>	▪ <code>IdDataFrame.drop_duplicates ()</code> ▪ <code>IdDataFrame.dropna ()</code> ▪ <code>IdDataFrame.dropna (axis = 1)</code>
<i>Les fonctions statistiques</i>	Liste des fonctions : <code>mean ()</code> – <code>min ()</code> – <code>max ()</code> – <code>sum ()</code> – <code>count ()</code> Syntaxe : <code>IdDataFrame ["NomColonne"].NomFonction()</code> ou bien <code>print (IdDataFrame ["NomColonne"].NomFonction())</code> Utilisation des fonctions avec groupement : Regroupement des données selon une colonne : <code>IdDataFrame.groupby ("Id_Colonne_à_regrouper") ["Id_Colonne_à_aggréger"].NomFonction()</code> ou bien <code>print (IdDataFrame.groupby ("Id_Colonne_à_regrouper") ["Id_Colonne_à_aggréger"].NomFonction())</code> Regroupement des données selon plusieurs colonnes : <code>IdDataFrame [["Id_Colonne1", "Id_Colonne2", ...]].groupby ("Id_Colonne1_Critère").NomFonction()</code> ou bien <code>print (IdDataFrame [["Id_Colonne1", "Id_Colonne2", ...]].groupby ("Id_Colonne1_Critère").NomFonction())</code>
<i>Affichage des données d'un DataFrame selon une ou plusieurs conditions</i>	▪ <code>IdDataFrame [IdDataFrame ["NomColonne"] OperateurComparaison Valeur]</code> ▪ <code>IdDataFrame [(IdDataFrame["NomColonne"]OperateurComp Valeur) OperateurLogique (IdDataFrame ["NomColonne"] OperateurComp Valeur)]</code>
<i>Tri des éléments d'un DataFrame</i>	<code>DataFrame_Trié = IdDataFrame.sort_values (by = ["NomColonne1", "NomColonne2" , ...] ascending = [True/False , True/False , ...])</code>
<i>Création d'un graphique</i>	▪ <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>