

RÉPUBLIQUE TUNISIENNE ★★★ MINISTÈRE DE L'ÉDUCATION	EXAMEN DU BACCALAURÉAT SESSION 2022	NOUVEAU RÉGIME
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Section : Économie et Gestion	
	Coefficient de l'épreuve : 0.5	Durée : 1h

**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 mieux gérer les séances de kinésithérapie, le responsable d'un centre thérapeutique se propose de créer la base de données simplifiée "**Gestion\_Séances**", décrite par le schéma textuel simplifié suivant :

**Patient ( IdPatient, NomP )**

**Kine ( MatKine, NomK )**

**TypeSeance ( CodeType, LibType, PrixSeance )**

**Seance ( IdPatient#, MatKine#, DateSeance, HeureSeance, CodeType# )**

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

Nom	Description	Type	Taille	Format	Contrainte
IdPatient	Identifiant d'un patient	Texte court	4		
NomP	Nom et prénom d'un patient	Texte court	50		Null interdit
MatKine	Matricule d'un kinésithérapeute	Texte court	4		
NomK	Nom et prénom d'un kinésithérapeute	Texte court	50		
CodeType	Code d'un type d'une séance	Texte court	4		
LibType	Libellé d'un type d'une séance	Texte court	100		Null interdit
PrixSeance	Prix d'une séance de kinésithérapeute exprimé en DT	Numérique	Entier		> 0
DateSeance	Date d'une séance	Date/Heure		Date abrégé	
HeureSeance	Heure d'une séance	Date/Heure		Heure abrégé	

#### 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 "**Gestion\_Séances**".
- 2) 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.
- 3) Remplir ces tables par les données représentées dans les tableaux suivants :

Patient	
IdPatient	NomP
P001	Chiheb Ali
P002	Chebbi Kais
P003	Attia Jihène
P004	Sallem Rim

Kine	
MatKine	NomK
K001	Abid Adel
K002	Cheikh Saida
K003	Charfi Sadok

TypeSeance		
CodeType	LibType	PrixSeance
T001	Post-Chirurgie	150
T002	Gériatrie	87
T003	Respiration	40
T004	Cardiovasculaire	180

Seance				
IdPatient	MatKine	DateSeance	HeureSeance	CodeType
P003	K002	13/01/2022	09:00	T004
P004	K003	14/02/2022	10:00	T003
P002	K001	11/04/2022	08:00	T001
P002	K003	11/04/2022	17:00	T001
P001	K002	28/04/2022	09:00	T002
P001	K002	06/05/2022	10:00	T002

**4) Créer les requêtes suivantes :**

**R1** : Afficher la liste des patients (**IdPatient**, **NomP**) traités par le kiné "Cheikh Saida".

**R2** : Augmenter de 20% le prix d'une séance pour le type ayant le libellé "**Respiration**".

**R3** : Afficher pour chaque type de séance son code, son libellé, le prix de la séance et le nombre total des séances effectuées.

**5) Exporter, dans votre dossier de travail et sous le nom "seance.csv", le contenu de la requête "**R3**" 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 :**

- Créer un fichier et l'enregistrer sous le nom "**kine**" dans votre dossier de travail.
- Choisir les méthodes et les propriétés adéquates, à partir de l'annexe (page 3), pour écrire dans le fichier créé les scripts permettant de/d':  
 1) Importer le contenu du fichier "**seance.csv**" et le stocker dans un DataFrame à nommer "**kin**".  
 2) Afficher les noms des colonnes de ce DataFrame.  
 3) Renommer la colonne contenant le nombre total des séances effectuées par "**NbSeances**".  
 4) Calculer et afficher le montant total des séances pour chaque type dans une nouvelle colonne à nommer "**MontantTotal**", sachant que :

$$\text{MontantTotal} = \text{PrixSeance} * \text{NbSeances}$$

- 5) Afficher les types des séances ayant un prix d'une séance qui dépasse 100 DT.
- 6) Créer, un graphique à barres de couleur "**Vert**" représentant les prix des séances en fonction de leurs libellés ayant pour titre "**Les prix par type de séance**".

Grille d'évaluation	Partie	A (13 points)					B (7 points)					
		1	2	3	4	5	1	2	3	4	5	6
	Question	1	4.5	2.5	4	1	1	0.5	0.75	1.5	1.5	1.75
	Note	1	4.5	2.5	4	1	1	0.5	0.75	1.5	1.5	1.75

# **ANNEXE**

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

Catégorie	Syntaxe
<i>Création et importation d'un DataFrame</i>	<pre>IdDataFrame = pandas.read_excel ("Chemin/ Nom_Fichier.extension", "Nom_Feuille") IdDataFrame = pandas.read_csv ("Chemin/Nom_Fichier.extension", sep = " séparateur")</pre>
<i>Manipulation d'un DataFrame</i>	<pre>IdDataFrame.shape ou bien print ( IdDataFrame.shape ) IdDataFrame.size ou bien print ( IdDataFrame.size ) IdDataFrame.info () ou bien print ( IdDataFrame.info () ) IdDataFrame.describe () ou bien print ( IdDataFrame.describe () ) IdDataFrame.columns ou bien print ( IdDataFrame.columns ) IdDataFrame.rename ( columns = { "NomColonne1" : "NouveauNomColonne1", "NomColonne2" : "NouveauNomColonne2", ... } , inplace= True ) Resultat = IdDataFrame.drop ( columns = [ "NomColonne1",...,"NomColonneN" ] ) Resultat = IdDataFrame.drop ( [ Indice_Ligne ] )</pre>
<i>Affichage des données d'un DataFrame</i>	<pre>IdDataFrame ou bien print ( IdDataFrame ) IdDataFrame [ "NomColonne" ] ou bien print ( IdDataFrame [ "NomColonne" ] ) IdDataFrame [ [ "NomColonne1" , "NomColonne2" , ...] ] ou bien print ( IdDataFrame [ [ "NomColonne1" , "NomColonne2" , ...] ] ) IdDataFrame [ "NomColonne" ] [ N° LigneInitial : N° LigneFinal ] ou bien print (IdDataFrame [ "NomColonne" ] [ N° LigneInitial : N° LigneFinal ]) IdDataFrame.head ( n ) ou bien print ( IdDataFrame.head ( n ) ) IdDataFrame.tail ( n ) ou bien print ( IdDataFrame.tail ( n ) ) IdDataFrame.iloc [ N° Ligne ] ou bien print ( IdDataFrame.iloc [ N° Ligne ] ) IdDataFrame.iloc [ N° LigneInitial : N° LigneFinal ] ou bien print ( IdDataFrame.iloc [ N° LigneInitial : N° LigneFinal ] ) IdDataFrame.iloc [ N° Ligne , N° Colonne ] ou bien print ( IdDataFrame.iloc [ N° Ligne , N° Colonne ] ) IdDataFrame.loc [ N° ligne , "NomColonne" ] = Valeur IdDataFrame.loc [ N° ligne ] = [ Liste_Valeur ] IdDataFrame [ "NomColonne" ] = Valeur IdDataFrame [ IdDataFrame [ "NomColonne" ] OperateurComparaison Valeur ] IdDataFrame [( IdDataFrame [ "NomColonne" ] OperateurComp Valeur) OperateurLogique ( IdDataFrame [ "NomColonne" ] OperateurComp Valeur )]</pre>
<i>Les fonctions statistiques</i>	mean ( ) – min ( ) – max ( ) – sum ( ) – count ( )
<i>Tri des éléments d'un DataFrame</i>	DataFrame_Trié = IdDataFrame.sort_values ( by = [ "NomColonne1" , "NomColonne2" , ...] , ascending = [ True/False , True/False , ... ] )
<i>Création d'un graphique</i>	<pre>import matplotlib.pyplot as alias_matplotlib IdDataFrame.plot.bar (x = "NomColonneAbscisse" , y = "NomColonneOrdonnée" , title = "TitreGraphique" , color = "CouleurGraphique" ) IdDataFrame.plot.line (x = "NomColonneAbscisse" , y = "NomColonneOrdonnée" , title = "TitreGraphique" , color = "CouleurGraphique" ) alias_matplotlib.show ( )</pre>