

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 vols quotidiens d'une compagnie aérienne, on se propose d'utiliser la base de données simplifiée intitulée "**Gestion_Vols**" décrite par le schéma textuel suivant :

AVION (CodeAv, ModèleAv, CapacitéAv)

PILOTE (MatPil, NomPrénomPil, AdressePil, TelPil)

VOL (NumVol, CodeAv#, MatPil#, DateVol, VilleDép, VilleArr, NbrePass, VolRéalisé)

Sachant que :

- **AVION** est une table contenant les informations relatives aux avions.
- **PILOTE** est une table contenant les informations relatives aux pilotes.
- **VOL** est une table contenant les informations relatives aux vols.

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

Nom	Description	Type	Taille	Format	Contrainte
CodeAv	Code d'un avion	Texte	4		
ModèleAv	Modèle d'un avion	Texte	50		Null interdit
CapacitéAv	Nombre maximum de passagers d'un avion	Numérique	Entier		≥ 50
MatPil	Matricule d'un pilote	Texte	8		
NomPrénomPil	Nom et prénom d'un pilote	Texte	50		Null interdit
AdressePil	Adresse d'un pilote	Texte	150		Null interdit
TelPil	Numéro de téléphone d'un pilote	Texte	8		Null interdit
NumVol	Numéro d'un vol	NuméroAuto			
DateVol	Date d'un vol	Date/Heure		Date, général	Null interdit
VilleDép	Ville de départ	Texte	50		Null interdit
VilleArr	Ville d'arrivée	Texte	50		Null interdit
NbrePass	Nombre de passagers dans un vol	Entier			>40
VolRéalisé	Vol réalisé (" Oui " ou " Non ")	Oui/Non			"Non" par défaut

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 "**Gestion_Vols**".
- 2) Créer les tables et les relations de cette base de données tout en respectant les types, les propriétés et les contraintes cités dans la description ci-dessus.

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

AVION		
CodeAv	ModèleAv	CapacitéAv
A001	AirBus A310	80
A002	AirBus A330	70
A003	AirBus A330	65
A004	Boeing 747	90

PILOTE			
MatPil	NomPrénomPil	AdressePil	TelPil
11111111	Ali Haddad	Gabes	99444444
22222222	Moez Salhi	Sousse	98333333
33333333	Salma Mejri	Nabeul	55666666

VOL							
NumVol	CodeAv	MatPil	DateVol	VilleDép	VilleArr	NbrePass	VolRéalisé
1	A001	11111111	29/04/2024	Tunis	Alger	68	Oui
2	A003	22222222	29/04/2024	Jedda	Tozeur	50	Non
3	A001	33333333	30/04/2024	Nfidha	Tunis	55	Oui
4	A003	11111111	02/05/2024	Tunis	Caire	60	Non
5	A002	22222222	03/05/2024	Tozeur	Nice	55	Non

4) Créer les requêtes suivantes :

R1 : Afficher les avions (**CodeAv**, **ModèleAv**, **NbrePass**) qui ont réalisé des vols à une date donnée,

R2 : Ajouter six voyageurs au nombre de passagers du vol numéro 3,

R3 : Afficher les vols (**CodeAv**, **ModèleAv**, **MatPil**, **DateVol**, **NbrePass**, **CapacitéAv**, **VolRéalisé**) programmés pendant le mois d'avril 2024.

5) Exporter, dans le dossier de travail, le résultat de la requête "**R3**" dans un classeur à nommer "**vols.xlsx**".

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 "**Vols**" 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 "**Vols**" les scripts permettant de (d') :

a) importer le contenu du classeur "**vols.xlsx**" et le stocker dans un DataFrame à nommer "**dfvol1**",

b) afficher les informations relatives au DataFrame "**dfvol1**",

c) créer, à partir du DataFrame "**dfvol1**", un DataFrame nommé "**dfvol2**" contenant les colonnes "**CodeAv**", "**ModèleAv**", "**DateVol**", "**NbrePass**" et "**CapacitéAv**" des vols réalisés uniquement,

d) renommer les colonnes "**CodeAv**" et "**DateVol**" du DataFrame "**dfvol2**" respectivement par "**Code Avion**" et "**Date Vol**",

e) trier le DataFrame "**dfvol2**" selon l'ordre croissant du nombre de passagers puis afficher son contenu.

f) créer et afficher un graphique, en barres vertes à partir du DataFrame "**dfvol2**", représentant le nombre de passagers ("**NbrePass**") pour chaque vol ("**Code Avion**"). Attribuer au graphique le titre "**Nombre de passagers par vol pour le mois d'avril 2024**".

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	0.5	1	0.75	1.5	1.75

ANNEXE

Les méthodes et les propriétés à utiliser avec la bibliothèque pandas

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