

RÉPUBLIQUE TUNISIENNE ----- MINISTÈRE DE L'ÉDUCATION	EXAMEN DU BACCALAURÉAT SESSION 2026	
	ÉPREUVE PRATIQUE D'INFORMATIQUE	
	Section : Économie et Gestion	
	Coefficient de l'épreuve : 0.5	Durée : 1 heure

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

Important : Dans le répertoire **Bac2026** 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 à ce sujet.

La société **Planète Clean** est spécialisée dans le ramassage des déchets collectés en les transportant par des camions vers des sites de décharge des déchets.

Afin de gérer les opérations de ramassage et de transport, la société utilise une base de données simplifiée intitulée "**BD_Dechets**", décrite par la représentation textuelle suivante :

SITEDECHARGE (IdSite, NomSite, AdrSite, CapaciteMax)

CAMION (Immat, VolumeMax, PoidsMax)

CHARGEDECHET (IdCh, DateCh, PoidsCh, VolumeCh, IdSite#, Immat#)

Soit la description des colonnes des tables de la base de données "**BD_Dechets**" :

Nom	Description	Type	Taille	Format	Contrainte
IdSite	Identifiant d'un site de décharge.	Texte court	4		
NomSite	Nom d'un site de décharge.	Texte court	25		
AdrSite	Adresse d'un site de décharge.	Texte court	40		
CapaciteMax	Capacité maximale d'un site de décharge (en m ³).	Numérique	Entier		Null interdit
Immat	Immatriculation d'un camion utilisé dans le transport de déchets.	Texte court	11		
VolumeMax	Volume maximal qu'un camion peut transporter (en m ³).	Numérique	Entier		
PoidsMax	Poids maximal qu'un camion peut transporter (en Tonne).	Numérique	Réel simple		
IdCh	Identifiant d'une charge de déchets.	Texte court	5		
DateCh	Date de transport d'une charge de déchets.	Date/Heure		Date, abrégé	
PoidsCh	Poids d'une charge de déchets (en Tonne).	Numérique	Réel simple		Supérieur ou égal à 0.5
VolumeCh	Volume d'une charge de déchets (en m ³).	Numérique	Entier		Supérieur à zéro

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 "**BD_Dechets**".
- 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 :

Table : SITEDECHARGE			
IdSite	NomSite	AdrSite	CapaciteMax
S001	Borj Chakir	Tunis	950
S002	Ksibet Elfoul	Sousse	660
S003	Tyna	Sfax	800
S004	Jradou	Zaghouan	850

Table : CAMION		
Immat	VolumeMax	PoidsMax
200 TU 3333	22	3.5
211 TU 9999	13	2.2
120 TU 8888	22	3.5
186 TU 7777	13	2.2

Table : CHARGEDECHET					
IdCh	DateCh	PoidsCh	VolumeCh	IdSite	Immat
C0001	15/02/2025	3.30	20	S001	200 TU 3333
C0002	30/04/2025	2.20	13	S002	211 TU 9999
C0003	11/06/2025	3.35	21	S002	120 TU 8888
C0004	15/02/2025	3.23	19	S001	200 TU 3333
C0005	15/04/2026	3.15	18	S003	120 TU 8888
C0006	17/03/2025	2.10	12	S004	186 TU 7777

4) Créer les requêtes suivantes :

- R1** : Afficher, pour chaque charge de déchets, son identifiant, sa date, son poids, son volume, le nom du site où elle est déchargée, l'adresse de ce site et sa capacité maximale, ainsi que l'immatriculation du camion qui l'a transporté.
- R2** : Afficher les noms des sites qui ont reçu des charges de déchets transportées par le camion de numéro d'immatriculation **120 TU 8888**.
- R3** : Afficher, pour chaque nom du site, le volume total des déchets déposés.
- R4** : Afficher les dates et les poids des déchets transportés par un camion d'immatriculation donnée pendant l'année 2025, triés selon l'ordre croissant des poids.

5) Exporter, dans votre dossier de travail et sous le nom "**Dechets.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 "**Clean**" dans votre dossier de travail.
- 2) En se référant à l'annexe, utiliser les méthodes et les propriétés adéquates, pour écrire dans le fichier **Clean** les scripts permettant :
 - a) d'importer le contenu du fichier **Dechets.csv** et le stocker dans un DataFrame à nommer "**dech**",
 - b) d'afficher le DataFrame **dech**,
 - c) de calculer, dans une nouvelle colonne à nommer "**Cout**", le coût de chaque charge transportée, sachant que : **Cout = PoidsCh * 100**
 - d) de trier le DataFrame **dech** dans un nouveau DataFrame à nommer "**dechtri**" par ordre croissant des noms des sites puis de l'afficher,
 - e) d'afficher, à partir du DataFrame **dechtri**, les charges des déchets dont le volume de charge dépasse 10m³,
 - f) de créer et d'afficher, à partir du DataFrame **dech**, un graphique en histogramme de couleur rouge représentant pour chaque identifiant de charge, le volume de la charge et ayant comme titre "**Volume par charge de déchets**".

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	2.25	6	1.25	0.5	1.25	0.25	1	1.25	1.25	1.5

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> <code>IdDataFrame.groupby (["Id_Colonne"]).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>