

Nom & prénom : Classe : Note :

Partie A : Gestion de données

Afin de gérer le suivi des congés d'employés d'une société, on se propose d'utiliser la base de données intitulée "Gestion_Conges" décrite par le schéma textuel simplifié suivant :

Employe (Identifiant, PrenomNom, Genre, Adresse, Tele, Statut, Salaire)
Conge (Identifiant#, DateDebut, DateFin, Type)

Les tables de données de la base de données :

Employe						
Identifiant	PrenomNom	Genre	Adresse	Tele	Statut	Salaire
E010	Sallami Salim	M	15 rue de la république Sousse	99123456	Cadre	1700
E020	Farj Mfarrej	M	16 rue de révolution Sousse	50556677	Technicien	1300
E030	Molka Mlika	F	17 rue la paie Sousse	22998877	Technicien	1400

Conge			
Identifiant	DateDebut	DateFin	Type
E010	01/02/2021	03/02/2021	Maladie
E010	05/03/2021	25/03/2021	Annuel
E010	11/12/2021	13/12/2021	Accident de travail
E020	20/01/2021	25/01/2021	Maladie
E020	07/07/2021	27/07/2021	Annuel
E030	02/02/2021	22/02/2021	Annuel
E030	06/06/2021	06/09/2021	Maternité

1) Compléter le tableau suivant :

Tables	Clé primaire	Clé étrangère
Employe		
Conge		

2) En déduire la représentation graphique de la base de données :

3) Afficher la liste des femmes employées techniciennes (Identifiant, PrenomNom, Statut) par ordre alphabétique décroissant selon leurs noms et prénoms.

Champs :				
Table :				
Tri :				
Afficher :	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Critères :				
Ou :				

4) Diminuer 1% des salaires des employés qui ont pris des congés (DateDebut) en Mars 2021.

Champs :				
Table :				
Mise à jour :				
Critères :				
Ou :				

5) Supprimer tous les congés pour un identifiant d'employé donné.

Champs :	
Table :	
Supprimer :	où
Critères :	
Ou :	

6) Afficher le total des salaires par statut.

Champs :				
Table :				
Opération :				
Tri :				
Afficher :	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>	<input type="checkbox"/>
Critères :				
Ou :				

Partie B : Analyse de données en langage Python sous la bibliothèque "pandas"

Afin d'analyser les informations concernant les employés, on a exécuté une requête permettant d'afficher la liste d'employés (Identifiant, Genre, Statut, Salaire) ainsi que leurs congés (DateDebut, DateFin, Type)

Identifiant	Genre	Statut	Salaire	DateFin	DateDebut	Type
E010	M	Cadre	1700	03/02/2021	01/02/2021	Maladie
E010	M	Cadre	1700	25/03/2021	05/03/2021	Annuel
E010	M	Cadre	1700	13/12/2021	11/12/2021	Accident de travail
E020	M	Technicien	1326	25/01/2021	20/01/2021	Maladie
E020	M	Technicien	1326	27/07/2021	07/07/2021	Annuel
E030	F	Technicien	1428	22/02/2021	02/02/2021	Annuel
E030	F	Technicien	1428	06/09/2021	06/06/2021	Maternité

La requête a été exportée vers un fichier "R.csv" ayant comme séparateur point-virgule.

Parmi les propositions suivantes, mettre **(V)** si la réponse est correcte sinon **(F)** dans le cas des propositions incorrectes.

1) Importer le contenu du fichier "R.csv", le stoker dans un DataFrame nommé "R" puis l'afficher.

	<code>r=read_csv("R.csv", sep=";")</code> <code>print(R)</code>
	<code>R=read_csv("R.xlsx", sep=";")</code> <code>print(R)</code>
	<code>R=read_csv("R.csv", sep=";")</code> <code>print(R)</code>
	<code>R=read_csv("R.csv", sep=",")</code> <code>print(R)</code>

2) La commande permettant d'afficher les dimensions du DataFrame `print(R.shape)` :

	56
	(8, 7)
	49
	(7, 7)

3) La commande permettant d'afficher le nombre d'éléments du DataFrame `print(R.size)` :

	56
	(8, 7)
	49
	(7, 7)

4) La commande permettant d'afficher les cinq premières lignes :

	<code>print(R.tail(5))</code>
	<code>print(R.head(5))</code>
	<code>print(R.tail(4))</code>
	<code>print(R.head(4))</code>

5) La commande permettant d'afficher les cinq dernières lignes :

	<code>print(R.tail(5))</code>
	<code>print(R.head(5))</code>
	<code>print(R.tail(4))</code>
	<code>print(R.head(4))</code>

6) La commande permettant d'afficher la colonne Identifiant à partir de la deuxième ligne jusqu'à la quatrième ligne du DataFrame :

	<code>print(R.Identifiant[2:3])</code>
	<code>print(R.Identifiant[1:3])</code>
	<code>print(R.Identifiant[1:4])</code>
	<code>print(R.Identifiant[2:4])</code>

7) La commande `print(R.iloc[2,6])` affiche :

	Accident de travail
	F
	M
	Maternité

8) La commande permettant d'afficher le salaire moyen :

	<code>print(r["Salaire"].mean())</code>
	<code>print(R["Identifiant"].mean())</code>
	<code>print(R["Salaire"].mean())</code>
	<code>print(R["Identifiant"].count())</code>

9) La commande permettant la liste des congés des techniciens ayant un salaire supérieur ou égal à 1400 DT

	<code>print(R[(R["Type"] == "Technicien") & (R["Salaire"]>=1400)])</code>
	<code>print(R[(R["Statut"] == "Technicien") & (R["Salaire"]>1400)])</code>
	<code>print(R[(R["Statut"] == "Technicien") & (R["Salaire"]>=1400)])</code>
	<code>print(R[(R["Statut"] == "Technicien") (R["Salaire"]>=1400)])</code>

10) Le script permettant de générer et d'afficher le graphique intitulé "Salaires des employés" en barres représentant les salaires des employés en fonction des identifiants d'employés

	<code>import matplotlib.pyplot as plt R.plot.bar(x="Statut", y="Salaire", title="Salaires des employés", color="black") plt.show()</code>
	<code>import matplotlib.pyplot as plt R.plot.line(x="Statut", y="Salaire", title="Salaires des employés", color="black") plt.show()</code>
	<code>import matplotlib.pyplot as plt R.plot.bar(x="Identifiant", y="Salaire", title="Salaires des employés", color="black") plt.show()</code>
	<code>import matplotlib.pyplot as df R.plot.bar(x="Identifiant", y="Salaire", title="Salaires des employés", color="black") plt.show()</code>

Bonne chance, vous êtes les meilleurs



Grille d'évaluation :

Partie	Questions	Points
Partie A	1	1.5
	2	1.5
	3	2
	4	2
	5	1.5
	6	1.5
Partie B	1	1
	2	1
	3	1
	4	1
	5	1
	6	1
	7	1
	8	1
	9	1
	10	1