|  |  |
| --- | --- |
|  | **مكتب التكوين المهني و إنعاش الشغل** |
| **Office de la Formation Professionnelle**  **et de la Promotion du Travail** |

Direction Régionale Fès Meknès

Examen de fin de module :

« Gestion des données »

Année 2022/2023

Variante 1

|  |  |  |  |
| --- | --- | --- | --- |
| Filière : | DEVOWFS | **Durée :** | **2h 30mn** |
| Niveau : | TS | **Barème :** | /40pts |

**La partie Théorique (1,5 Pts/question)**

**Questions de cours**

**La partie Pratique**

**Dossier 1 : la partie MYSQL (20 Pts)**

On vous présente un extrait d’une base de données relationnelle d’une coopérative qui utilise des produits phytosanitaires

Plantes (idPlante, nomPlante, typePlante, Description, datePlantation, emplacement)

Traitements (id*traitement*, idPlante#, idProduitSanitaire#, quantiteUtilisee, methodeApplication, dateTraitement, description, personnel#)

ProduitsSanitaires (idProduitSanitaire, nomProduit, description, modeAction, dateFabrication, datePeremption, Prix, quantitéStock)

Employes (idEmployé, nom, prenom, poste, contact)

1. Ajouter les contraintes suivantes : (2 Pt)

* La Méthode d’application ne peut être que « Pulvérisation foliaire », « Arrosage au sol », « Fumigation »
* La date de fabrication < la date de péremption

1. Ecrire une procédure qui affiche la liste des produis qui seront périmés le mois prochain (3 Pts)
2. Ecrire une procédure qui affiche la liste des traitements réalisés sur une plante donnée pendant une période donnée (id\_traitement, date\_traitement, description, methodeApplication, nomEmploye) (3 Pts)
3. Ecrire un trigger qui empêche la réalisation d’un traitement si le produit à utiliser est périmé ou si la quantité à utiliser est inférieure à la quantité en stock sinon le trigger doit mettre à jour la quantité en stock (4 Pts)
4. Ecrire une fonction qui retourne le nombre de traitements réalisé avec le produit « chlorpyrifos » pendant les 5 dernières années (3 Pts)
5. Créer 2 comptes « Chef » et « Fellah » (1 Pt)
6. Accorder au compte Fellah le droit select sur la table traitement (1 pts)
7. Créer un rôle « administrateur » et accorder au groupe tous les droits sur tous les objets de la base de données (2 Pts)
8. Affecter le chef au rôle administrateur (1 Pt)

**Dossier 2 : MongoDB (14 Pts)**

Notre BD mongo contient une collection Plantes contenant les champs suivants :

\_id (il s’agit de l’id de la plante), nomPlante, datePlantation, et une liste des traitements livrées à la plante (ProduitSanitaire, quantiteUtilisee, montant, methodeApplication, dateTraitement, matEmploye)

Et une collection Employe contenant les champs suivants : matEmploye, nom, prenom, poste, contact, salaire

Ecrire les requêtes suivantes :

1. Ajouter un document à la collection plante (1 pt)
2. Afficher la liste des plantes (nomPlante, datePlantation) triés par datePlantation (1 pts)
3. Afficher la liste des traitements réalisés le mois dernier (2 Pts)
4. Afficher pour chaque plante le nombre de traitement et la somme totales de ces traitements (2 Pts)
5. Supprimer les plantes ayant subi un traitement avec « chlorpyrifos » (2 Pts)
6. Afficher une liste des traitements réalisés par l’employé nommé « RAMI » (2 Pts)
7. Ecrire la requête qui permet d’augmenter les salaires des employés de 10% (2 Pts)
8. Ecrire le code python qui permet d’afficher pour chaque plante les traitements phytosanitaires subis (2 Pts)
9. Voici le **corrigé complet** pour les parties **SQL (5.a à 5.d)** et **MongoDB (a à h)**, y compris la **fonction demandée pour "chlorpyrifos"**.
10. **✅ Partie SQL**
11. **5. Fonction : nombre de traitements réalisés avec le produit « chlorpyrifos » pendant les 5 dernières années**
12. DELIMITER //
13. CREATE FUNCTION nb\_traitements\_chlorpyrifos()
14. RETURNS INT
15. DETERMINISTIC
16. BEGIN
17. DECLARE nb INT;
18. SELECT COUNT(\*)
19. INTO nb
20. FROM Traitements T
21. JOIN ProduitsSanitaires P ON T.idProduitSanitaire = P.idProduitSanitaire
22. WHERE P.nomProduit = 'chlorpyrifos'
23. AND T.dateTraitement >= DATE\_SUB(CURDATE(), INTERVAL 5 YEAR);
25. RETURN nb;
26. END;
27. //
28. DELIMITER ;
29. **5.a Créer deux comptes : Chef et Fellah**
30. CREATE USER 'Chef'@'localhost' IDENTIFIED BY 'chef123';
31. CREATE USER 'Fellah'@'localhost' IDENTIFIED BY 'fellah123';
32. **5.b Donner le droit SELECT à Fellah sur la table Traitements**
33. GRANT SELECT ON Traitements TO 'Fellah'@'localhost';
34. **5.c Créer un rôle administrateur avec tous les droits**
35. CREATE ROLE 'administrateur';
36. GRANT ALL PRIVILEGES ON \*.\* TO 'administrateur';
37. **5.d Affecter le compte Chef au rôle administrateur**
38. GRANT 'administrateur' TO 'Chef'@'localhost';
39. SET DEFAULT ROLE 'administrateur' TO 'Chef'@'localhost';
40. **✅ Partie MongoDB**
41. **a. Ajouter un document à la collection Plantes**
42. db.Plantes.insertOne({
43. \_id: 1,
44. nomPlante: "Tomate",
45. datePlantation: ISODate("2024-10-01"),
46. traitements: [
47. {
48. ProduitSanitaire: "chlorpyrifos",
49. quantiteUtilisee: 2,
50. montant: 15,
51. methodeApplication: "Pulvérisation foliaire",
52. dateTraitement: ISODate("2025-03-15"),
53. matEmploye: "E01"
54. }
55. ]
56. })
57. **b. Afficher les plantes triées par date de plantation**
58. db.Plantes.find({}, { nomPlante: 1, datePlantation: 1, \_id: 0 }).sort({ datePlantation: 1 })
59. **c. Afficher les traitements du mois dernier**
60. let now = new Date();
61. let firstDay = new Date(now.getFullYear(), now.getMonth() - 1, 1);
62. let lastDay = new Date(now.getFullYear(), now.getMonth(), 0);
63. db.Plantes.aggregate([
64. { $unwind: "$traitements" },
65. { $match: {
66. "traitements.dateTraitement": { $gte: firstDay, $lte: lastDay }
67. }},
68. { $project: {
69. nomPlante: 1,
70. traitement: "$traitements"
71. }}
72. ])
73. **d. Nombre de traitements + somme totale par plante**
74. db.Plantes.aggregate([
75. { $unwind: "$traitements" },
76. { $group: {
77. \_id: "$nomPlante",
78. nbTraitements: { $sum: 1 },
79. totalMontant: { $sum: "$traitements.montant" }
80. }}
81. ])
82. **e. Supprimer les plantes traitées avec "chlorpyrifos"**
83. db.Plantes.deleteMany({
84. "traitements.ProduitSanitaire": "chlorpyrifos"
85. })
86. **f. Afficher les traitements faits par l’employé RAMI**
87. let employe = db.Employe.findOne({ nom: "RAMI" });
88. db.Plantes.aggregate([
89. { $unwind: "$traitements" },
90. { $match: { "traitements.matEmploye": employe.matEmploye } },
91. { $project: {
92. nomPlante: 1,
93. traitement: "$traitements"
94. }}
95. ])
96. **g. Augmenter les salaires de 10%**
97. db.Employe.updateMany({}, { $mul: { salaire: 1.10 } })
98. **h. Code Python : traitements par plante**
99. from pymongo import MongoClient
100. client = MongoClient("mongodb://localhost:27017/")
101. db = client["Cooperative"]
102. plantes = db.Plantes.find()
103. for plante in plantes:
104. print(f"Plante : {plante['nomPlante']}")
105. traitements = plante.get("traitements", [])
106. for t in traitements:
107. print(f" ➤ Produit : {t['ProduitSanitaire']}, Méthode : {t['methodeApplication']}, Date : {t['dateTraitement']}")
108. Souhaites-tu que je te **prépare un fichier Word ou PDF** avec ce corrigé pour que tu puisses le **télécharger facilement** ?