Lycée Ib	n Elhaythem	Matmata	Nouvelle
----------	-------------	---------	----------

Date: 08/03/2019

Durée: 2h

Niveau: 4 SI

Coefficient: 1.5

Nom : Enseignant : Mr : HELALI Miloud

Prénom :

Correction de Devoir de synthèse N°2

Matière: BDD

Exercice 1 : (3 points)

✓ Répondre pour chaque proposition par **vrai** ou **faux**:

Proposition	Réponse (vrai/faux)
La valeur de la clé primaire d'une table quelconque est unique.	vrai
La clé primaire et la clé étrangère doivent être identiques seulement en nom.	faux
Une table peut avoir plusieurs clés étrangères.	vrai
La requête de jointure peut être utilisée sur une seule table.	faux
Lors d'une requête de tri on peut remplacer dans la clause order by les	vrai
noms des champs par leurs indices dans la clause select .	
Lors de la gestion d'une base de données on commence par les requêtes de	faux
LMD puis les requêtes de LDD.	

Exercice 2: (5 points)

> Soit la liste des colonnes suivante :

Champ	Description	Type	Taille	entité
num_ele	Le numéro de l'élève.	Entier		Elève
nom_pre	Le nom et prénom de l'élève.	Chaine	50	Elève
classe	La classe de l'élève.	Chaine	50	Elève
genre	Le genre de l'élève.	Caractère	1	Elève
age	L'âge de l'élève.	Entier		Elève
code_liv	Le code du livre.	Chaine	20	Livre
titre	Le titre de livre.	Chaine	100	livre
auteur	L'auteur de livre.	Chaine	100	livre
categ	La catégorie de livre.	Chaine	100	livre
num_emp	Le numéro de l'opération de l'emprunt.	Entier		Emprunt
num_ele	Le numéro de l'élève.	Entier		Emprunt
code_liv	Le code du livre.	Chaine	20	Emprunt
date_emp	La date de l'opération de l'emprunt.	Date		Emprunt
duree_emp	La durée de l'opération de l'emprunt.	Entier		Emprunt

1/ Déduire la liste des tables. (1pt)

Liste des tables				
Nom Table	Description	sujet		
Elève	Regroupe les élèves de la bibliothèque	Elèves		
Livre	Regroupe les livres de la bibliothèque	Livres		
Emprunt	stocke les emprunts de la bibliothèque	emprunts		

2/ Donner la liste des liens entre les tables. (2pts)

Table mère	Table fille	Clé primaire	Clé étrangère
Elève	Emprunt	num_ele	num_ele
Livre	emprunt	Code_liv	Code_liv

3/ Donner une représentation textuelle de la base de données. (2pts)

Elève (<u>num_ele</u>, nom_pre, classe, genre, age)

Livre (**code liv**, titre, auteur, categ)

Emprunt (<u>num_emp</u>, num_ele# ,code_liv#, date_emp, duree_emp)

Exercice 3: (12 points)

✓ On suppose qu'on veut créer la base des données dite «**gestion_hopital**» dont la quelle on va créer les trois tables suivantes :

Table « patient »					
champ	type	taille	Contrainte		
<u>numéro</u>	entier		Clé primaire		
Nom	chaine	30			
Prénom	chaine	30			
Genre	Charactère	1	'M' or 'F'		
Age	entier				

Table « medecin »						
champ type taille Contrainte						
<u>Matricule</u>	entier		Clé primaire			
Nom_prenom	chaine	50				
spécialité	chaine	50				
Tel	chaine	10				

Table « ordonnance »				
champ	type	taille	Contrainte	
<u>code</u>	chaine	10	Clé primaire	
numero#	entier		Clé étrangère	
Matricule#	chaine	50	Clé étrangère	
Date_ord	date	_		
List_med	chaine	150		

O Partie I: (5 points)

➤ On utilisant le langage de Définition de données (LDD), répondre aux requêtes suivantes :

```
1/ Créer la base de données «gestion_hopital». (0.5pt)
                                                   Votre requête SQL a été exécutée avec succès (traitement: 0.0213 sec.)
  create database gestion hopital;
                                                   requête SQL:
                                                   CREATE DATABASE destion hopital
                                                                      [Modifier] [Créer source PHP]
2/ Créer la table «patient». (1pt)
create table patient
                                                Votre requête SQL a été exécutée avec succès (traitement: 0.1811 sec.
(numero int primary key,
                                                requête SQL:
nom varchar(30),
                                                  numero INT PRIMARY KEY
prenom varchar(50),
                                                  nom '
genre char check genre ='F' or genre='M',
                                                 prenom
                                                                50
                                                         AR CHECK genre = 'F' OR genre = 'M'
                                                  genre C
age int);
                                                  age INT
                                                                     [Modifier] [Créer source PHP]
Créer la table «medecin».
                                             Votre requête SQL a été exécutée avec succès (traitement: 0.0558 sec.
create table medecin
(matricule int primary key,
                                             requête SQL:
nom prenom varchar(50),
                                               matricule INT PRIMARY KEY
                                               nom_prenom VAR
specialite VARCHA
tel VARCHAR( 10
specialite varchar(50),
                                                            AR( 50
tel varchar(10));
                                                                   [Modifier] [Créer source PHP]
3/ Créer la table «ordonnance». (1pt)
create table ordonnanace
        varchar(10) primary key,
numero int references patient(numero) on delete cascade,
matricule
            int references medecin (matricule) on delete cascade,
date ord
             date.
                         Votre requête SQL a été exécutée avec succès (traitement: 0.1516 sec.)
list_med
             text);
                         requête SQL:
                         CREATE TABLE ordonnanace(
code VARCHAR( 10 ) PRIMARY KEY
                                     T REFERENCES patient( numero ) ON DELETE CASCADE .

NT REFERENCES medecin( matricule ) ON DELETE CASCADE
                           numero INT
                           matricule
                           date_ord
                           list med TEX
                                                   [Modifier] [Créer source PHP]
4/ Ajouter le champ « adresse » à la table «patient». (0.5pt)
alter table patient add adresse varchar(100);
          Votre requête SQL a été exécutée avec succès (traitement: 0.1632 sec.)
          requête SQL:
              ER TABLE patient ADD adresse VARCHAR( 100
                                    [Modifier] [Créer source PHP]
5/ Supprimer la clé primaire de la table «ordonnance» puis la remplacer par les trois champs
« Numero », « Matricule » et « Date ord ». (Deux requêtes)
                                                                           (1pt)
alter table ordonnance drop primary key;
                                                Votre requête SQL a été exécutée avec succès (traitement: 0.0910 sec.)
                                                requête SQL:
                                                ALTER TABLE ordonnance DROP PRIMARY KEY
                                                                     [Modifier] [Créer source PHP]
alter table ordonnance add primary key(numero,matricule,date_ord);
                 Votre requête SQL a été exécutée avec succès (traitement: 0.0735 sec.)
                 requête SQL:
                 ALTER TABLE ordonnance ADD PRIMARY KEY ( numero, matricule, date ord
                                           [Modifier] [Créer source PHP]
6/ Désactiver puis réactiver la clé primaire de la table «patient» (1pt)
  alter table patient disable constraint primary key;
```

alter table patient enable constraint primary key;

o Partie II: (7 points)

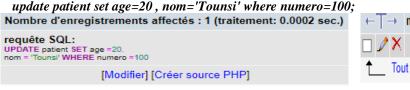
> On utilisant le langage de LMD, répondre aux requêtes suivantes :

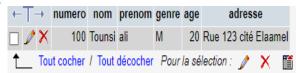
1/ Insérer l'enregistrement suivant dans la table «patient» (1pt)

numero	Nom	Prenom	Genre	age
100	Gabsi	Ali	M	18

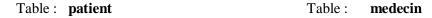
insert into patient values(100, 'gabsi', 'ali', 'M', 18, 'Rue 123 cité Elaamel');







• On suppose que les tables contiennent les enregistrements suivants :



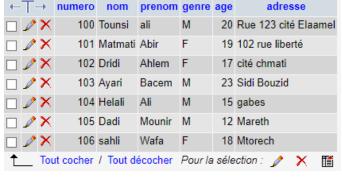




Table: Ordonnance

←T→	code	numero	matricule	date_ord	list_med
	Ord1000	100	10	2018-01-09	Algesic,fervex,butalin,saifoxyl
	Ord1001	102	11	2018-02-06	fervex,paracetyl,pectolyse,butalin,famodar
	Ord1002	104	13	2018-01-16	A cupan, A spasmed, but a lin, fervex, gripex, algesic
	Ord1003	106	14	2018-01-30	pectolyse,stopalgic,saifoxyl
	Ord1004	102	11	2018-02-08	algesic,fervex,famodar,butalin
	Ord1005	103	13	2018-01-11	Algesic,butalin,acupan,paracetyl,fervex
	Ord1006	101	12	2018-01-04	pectolyse,algesicbutalin,acupan,paracetyl
	Ord1007	105	14	2018-03-22	Algesic,fervex,famodar,saifoxyl
↑ Tou	ıt cocher	/ Tout dé	cocher Po	our la sélectio	on: 🤌 🗙 🛗

3/ Supprimer de la table «medecin» le médecin ayant la spécialité orthopédie ou pédiatrie (0.5pt) delete from medecin where specialite like 'orthopédie' or specialite like 'pédiatrie';

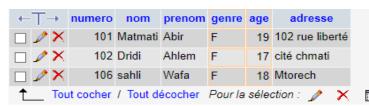


Le contenu de la table medecin :



4/ Afficher la liste des patients qui ont des âges inférieurs à 20 et qui sont des filles. (1pt) select * from patient where age<20 and genre='F';

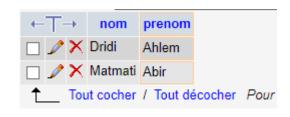




5/ Afficher les noms et les prénoms des patients qui ont des âges<= 20, qui sont des filles et leurs prénoms commencent par une voyelle et qui sont triées d'une manière croissante selon leurs âges. (1pt)

select nom,prenom from patient where age<=20 and genre='F' and (prenom like 'a%' or prenom like 'e%' or prenom like 'i%' or prenom like 'u%' or prenom like 'o%' or prenom like 'y%')
order by age asc;



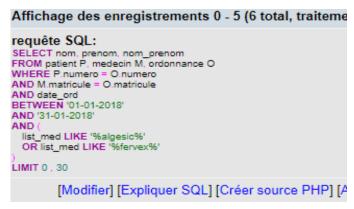


6/ Afficher les noms et prénoms des patients et des médecins pour les ordonnances de janvier 2018 et qui contiennent les médicaments «Algesic» ou «fervex». (1pt)

select nom,prenom,nom_prenom

from patient P, medecin M, ordonnance O

where P.numero=O.numero and M.matricule=O.matricule and date_ord between '01-01-2018' and '31-01-2018' and (list_med like '%algesic%' or list_med like '%fervex%');





7/ Afficher le nombre des patients de l'année 2018 qui sont examinés par les médecins de pédiatrie. (1pt) select count(numero) 'nbre des patients'

from ordonnance O, medecin M

where M.matricule=O.matricule and specialite like 'pédiatrie'

group by (specialite)



8/ Afficher le nombre des visites faites par le médecin ayant la matricule « 10 » pour l'année 2018 (1pt)

select count(matricule) 'nbre de visite de medecin numero 10 en 2018' from ordonnance O where date_ord between '01-01-2018' and '31-12-2018' and matricule=10 group by (matricule);

Affichage des enregistrements 0 - 0 (1 total, traitement: 0.0002 sec.)	
requête SQL: SELECT COUNT(matricule) 'nbre de visite de medecin numero 10 en 2018' FROM ordonnance O WHERE date_ord BETWEEN '01-01-2018' AND '31-12-2018' AND matricule =10 GROUP BY (matricule	
LIMIT 0 , 30	
[Modifier] [Expliquer SQL] [Créer source PHP] [Actualiser]	
Afficher: 30 ligne(s) à partir de l'enregistrement n° (centre	
← ⊤ →	
nbre de visite de medecin numero 10 en 2018	
1	

Bon travail.