

## Examen final

### Exercice 1 :

Soit la base de données ayant le schéma relationnel suivant, les clés primaires sont soulignées et les clés étrangères sont indiquées à la suite à chaque relation :

**recette**(IdRecette int, etapes text(20), categorie int)

recette.categorie référence Categorie.idCateg

**Ingredient** (nomIng text(20), typeIng text(20));

**Categorie**(idCateg int, nomCateg text(20));

**Chef** (idChef int, nomChef text(20));

**Emission**(nomEmission text, présentateur text(20), jourDiffusion text(20), heureDiffusion int);

RecetteEmission(IdRecette int, nomEmission text(20), datepassage date, Nbvues int)

RecetteEmission.IdRecette référence Recette.IdRecette

RecetteEmission.nomEmission référence Emission.nomEmission

**RecetteChef** (idRecette int, IdChef int)

recetteChef.IdChef référence Chef.idChef

recetteChef.idRecette référence Recette.idRecette

**IngRecette** (idRecette int, nomIng text(20), Qte int, mesure text(20))

IngRecette.idRecette référence Recette.idRecette

IngRecette.nomIng référence Ingredient.nomIng

1. Donner les requêtes SQL répondant aux besoins suivants :

<p>· Qui sont les identifiants des chefs ayant le plus de vues (2pt)</p> <pre>select idchef, sum(nbvues) totalvues from recetteEmission re, recettechef rc where re.idrecette=rc.idrecette group by idchef  having totalvues = (select max(totalViews) from (select idchef,sum(nbvues) totalViews from recetteEmission re, recettechef rc where re.idrecette=rc.idrecette group by idchef))</pre>	<p>2 points si requête correcte (syntaxiquement et sémantiquement)</p> <p>0 sinon</p>
---	---

<p>· Quelles sont les catégories (id et nom) de recettes ne recensant aucune recette (1pt)</p> <pre>select * from categorie c where idcateg not in ( select categorie from recette ) ***Ou**** Select c.* from categorie c left join recette r on c.idcateg=r.categorie where r.categorie is null</pre>	<p>1 point si requête correcte</p> <p>0 sinon</p>
<p>· Les identifiants des chefs ayant au moins une recette chacun passées dans la même émission (2pts)</p> <pre>select distinct chef1, chef2 from ( select idr1, idr2, chef1, rcc.idchef chef2 from recettechef rcc join( select idr1, idr2, rc.idchef chef1 from recettechef rc join( select R.idrecette idr1, R1.idrecette idr2 from recetteemission R join recetteemission R1 where R1.nomEmission=R.nomEmission and r.idrecette&lt;&gt;R1.idrecette )T where rc.idrecette=T.idr1 )T2 where rcc.idrecette=T2.idr2 )T3 where chef1 &lt;&gt; chef2</pre> <p>· Les identifiants des chefs ayant au moins une recette chacun passées dans les émissions 1 et 2 (2pts)</p> <pre>select distinct idchef from recettechef where idrecette in (select idrecette from recetteemission where nomemission="1" intersect select idrecette from recetteemission where nomemission="2")</pre>	<p>2 points si requête correcte</p> <p>0 sinon</p>

<p>2. Indiquer quelles sont les requêtes non valides, et justifiez votre réponse (1pt) :</p> <pre>select jourdiffusion, count(nomemission) from emission group by jourdiffusion where jourdiffusion='Lundi' OR jourdiffusion='Mercredi'</pre> <p><i>Requête 1 non valide car where vient après le group by</i></p> <pre>select * from recetteEmission R, emission E where R.nomEmission= E.nomEmission and R.datepassage='2020-03-25' and présentateur like 's_mira'</pre> <p><i>Requête 2 valide</i></p>	<p>0,5 pt pour requête 1 non valide et 0,5pt pour le justificatif</p>
<p>3. Ecrivez en une seule phrase ce que cette requête exprime (1pt)</p> <pre>SELECT COUNT(DISTINCT rc2.idchef) FROM chef c, recettechef rc1, recettechef rc2 WHERE rc1.idchef=c.idchef AND nomChef='Karima' AND rc2.idchef&lt;&gt;c.idchef AND rc1.idrecette=rc2.idrecette</pre> <p><i>Le nombre de co-chef (i.e. qui ont au moins une même recette) d'un chef appelé « Karima »</i></p>	<p>1 pt si correct</p> <p>0 sinon</p>
<p>4. Donner le script de création de la vue V1 qui affiche l'identifiant des recettes et le nb d'ingrédients indiqués dans la recette (1pt)</p> <p>create view v1 as</p> <pre>select idrecette, count(nomIng) nbIng from ingrecette group by idrecette</pre>	<p>1 pt si correct : 0,5 pour le « create view as »</p> <p>Et 0,5 pour la requête de la vue</p>

5. Donner les requêtes ALGÈBRIQUES répondant aux besoins suivants :

<p>· Le nombre d'émissions n'ayant pas passé de recettes du chef 'Salim' (1,5pt)</p> <pre> r1&lt;-restrict(chef,nomchef='Salim') r2&lt;-join(r1,recettechef) r3&lt;-join(r2,recetteemission) r5&lt;-project(r3,nomemission) r6&lt;-difference(project(emission,nomemission),r5) r7&lt;-agregat(r6,count,*) </pre>	<p>1,5 pt si correct</p> <p>0.75 pour l'affichage des noms d'émissions n'ayant pas passé de recettes du chef 'Salim' sans compter leur nombre</p> <p>0 sinon</p>
<p>· La liste des émissions ayant passé la recette « 1 » plus d'une fois. On souhaite afficher toutes les informations de l'émission (1,5pt)</p> <pre> r1&lt;-restrict(recetteemission,idrecette=1) r2&lt;-agregat(r1,nomemission,count,idrecette:nb) r3&lt;-restrict(r2,nb&gt;1) r4&lt;-join(r3,emission) </pre>	<p>1,5 pt si correct</p> <p>0,75 pour l'affichage des noms émissions ayant passé la recette 1 plus d'une fois (sans leurs informations)</p> <p>0 sinon</p>

6. Nous souhaitons enrichir la BD par une nouvelle relation Restaurant (id, nom) (id est la clé primaire) indiquant les chefs possédant des restaurants. Un restaurant n'appartient qu'à un seul chef.

<p>· Donner le script SQL permettant la modification de la base (1pt)</p> <pre> create table restaurant (idrestau int, nomrestau text(40), idchef int, primary key(idrestau), foreign key (idchef) references Chef(idChef) ) </pre>	<p>1 pt si correct</p> <p>0 sinon</p>
---	---------------------------------------

Donner le script permettant d'insérer deux enregistrements à la fois dans la nouvelle table (1pt)	1 pt si correct
insert into restaurant values (1,'restau1',1),(2,'restau1',2)	0 sinon

### **Exercice 2:**

1. Soit le descriptif suivant d'une base de données d'une société qui gère plusieurs parcs automobiles : chaque parc est caractérisé par son identifiant, son nom et son adresse, est décomposé en plusieurs zones. Une zone est décrite par son identifiant, sa position et sa capacité maximale. Le parc reçoit des voitures garées à un emplacement (couloir et numéro) dans une zone. Chaque voiture est décrite par son matricule et son propriétaire. Le parc garde trace de l'identifiant, du nom, et l'adresse du propriétaire.

- Voici la relation universelle de cette base :  
Parc (idParc, nomParc, adrParc, idZone, positionZone, CapaciteZone, EmplacementVoiture, Matricule, idProprietaire, NomProprietaire, AdrProprietaire)
- Identifier les dépendances fonctionnelles qui ne sont pas vérifiées (1pt)  
idParc → nomParc, adrParc  
idParc → idZone  
idZone → positionZone, CapaciteZone  
idProprietaire → Matricule  
EmplacementVoiture → idZone

<p>Les DF non vérifiées :</p> <p>idParc-&gt;idZone car le parc a plusieurs zones</p> <p>idProprietaire -&gt; Matricule car un propriétaire peut avoir plusieurs voitures et donc plusieurs matricules</p>	<p>0,25 pour chaque DF et 0,25 pour sa justification</p>
---	--

En quelle forme normale est cette relation ? (1pt)

N'est pas en première forme normale (positionVoiture est multivalué)	0,5 pour réponse + 0,5 pour la justification
--	--

2. Soit la relation suivante :

$R(A,B,C,D,E,F,G,H)$

Soit l'ensemble F des DF définies sur R :

$A,B \rightarrow C$ ;  $A,C \rightarrow E$ ;  $B \rightarrow D$ ;  $C,D,E \rightarrow F$ ;  $E \rightarrow G$ ;  $G,F \rightarrow A$ ;  $F \rightarrow H$

· Lesquelles de ces ensembles d'attributs forment une clé (3pt) : AC, AEF, ABC, CGF, AB

<p>AB est la seule clé car</p> <p><math>AB^+(F) = R</math></p> <p>et <math>A^+(F) \neq R</math></p> <p>et <math>B^+(F) \neq R</math></p> <p>Pour tous les autres ensembles, B est une source donc tout ensemble qui ne contient pas B n'est pas une clé et pour ABC c'est une super clé</p>	<p>(0,5pt * 3) donc 0,5 pour chaque partie de la réponse (<math>AB^+</math>, <math>A^+</math> et <math>B^+</math>, sinon expliquer que B est une source et calculer <math>AB^+</math> et <math>B^+</math> uniquement)</p> <p>1,5 pt pour l'indication des non clés (1pt pour ABC non clé et 0,5pt pour le reste des non-clés)</p> <p>-0,25pt pour chaque erreur (y compris le rajout d'une clé qui ne l'est pas)</p> <p>0 sinon</p>
---	---

· En considérant chacune des clés identifiées à la question précédente :

a. Quelles sont les DF (de l'ensemble F) qui ne respectent pas la 3FN (1pt) ?

<p><math>C,D,E \rightarrow F</math>      <math>E \rightarrow G</math>      <math>F \rightarrow H</math></p> <p><math>B \rightarrow D</math></p>	0.25 par DF
---	-------------

b. Quelles sont les DF (de l'ensemble F) qui ne respectent pas la BCNF (1pt) ?

<p>Toutes les DF sauf la première : <math>A,C \rightarrow E</math>;  <math>B \rightarrow D</math>; <math>C,D,E \rightarrow F</math>; <math>E \rightarrow G</math>; <math>G,F \rightarrow A</math>; <math>F \rightarrow H</math></p>	0,25 pour chaque DF correcte (maximum 1 pt)
---	---