

Evaluation 2017-2018

Filière : Télécommunications

Classe : T2

Cours : SI1

Date : Mai 2018

Temps : 15h00 à 16h35. Pour ceux qui ont droit au temps supplémentaire : 15h00 à 17h04

Examinatrice : H. Chabbi Drissi

Nom et prénom : Zambar Yavick

Note :

Note max = 6

Toutes les réponses sont à reporter sur l'énoncé

Aucun document n'est autorisé mis à part l'annexe disponible sur le site dans l'état sans aucun ajout de votre part.

Toute réponse doit être pleinement justifiée.

Le tableau ci-dessous vous indique les points accordés à chaque partie.

Le temps estimé y est donné à titre indicatif.

	Points max	Temps estimés	Points obtenus
Lecture énoncé	-	5min	-
Exercice 1	1	15min	1
Exercice 2	0.7	15min	0.7
Exercice 3	1.4 + 0.2	25min	1.5
Exercice 4	1.9	35min	1.8
Total	5.2	95min	5.

Convention : Dans la suite, chaque V = 0.1pt

Exercice 1 Connaissances théoriques

Pour chacune des questions suivantes :

- entourer les réponses correctes. V par bonne réponse *Rare concours!*
- ou compléter les phrases (là où il y a des pointillés!) VV par bonne réponse

1. Dans un modèle EA, quand on définit une spécialisation on doit porter dans le rond (ou triangle sur DBmain) de cette spécialisation, la réponse à deux questions. Celle qui porte sur le lien entre la mère et ses filles, concerne la propriété du "1" → si la spécialisation est totale ou non

2. Dans une table qui a une contrainte foreign key en autoréférence (référence vers sa propre clé primaire), on pourrait la remplacer par une contrainte de type check : VRAI ou FAUX?

3. On peut créer une vue qui utilise une requête sur autre vue : VRAI ou FAUX?

4. Une foreign key doit avoir le même nom que la primary key qu'elle référence : VRAI ou FAUX?

5. Soient deux relations qui ont le même schéma :

$R(A(int), B(char(10)))$

$S(A(int), B(char(10)))$

On définit sur R la contrainte unique(A,B) et sur S la contrainte unique(B,A). Ces contraintes ne vont pas nous permettre d'insérer exactement les même tuples dans R que dans S : VRAI ou FAUX?

6. On suppose disposer de deux relations ayant le même schéma relationnel, où les attributs soulignés indiquent la clé primaire :

$R(\underline{id(int)}, a(int))$

$S(\underline{id(int)}, a(int))$

On exécute alors cette requête :

```
SELECT id FROM R
WHERE id NOT IN (SELECT S.id FROM S)
```

Laquelle des requêtes suivantes (entourée la bonne requête) donne toujours la même réponse que la requête présentée précédemment :

```
-- Requête 1:
SELECT id FROM R
MINUS
SELECT id FROM S
-- Requête 2:
SELECT id FROM R
WHERE NOT EXISTS (SELECT S.id FROM S) X
-- Requête 3:
SELECT R.id FROM R,S
WHERE R.id <> S.id X
```

7. Soient les deux relations suivantes. Les attributs soulignés indiquent leurs clés primaires :

Personne(NumP(int), nom, adresse) avec 5 tuples

Inscription(NumP(int), NumActivite(int), date) avec 10 tuples

(a) Quel est le nombre maximum de tuples que retourne la requête suivante. Entourez la bonne réponse :

```
SELECT *
FROM Personne, Inscription;
```

i. 5

ii. 10

iii. 15

iv. 50

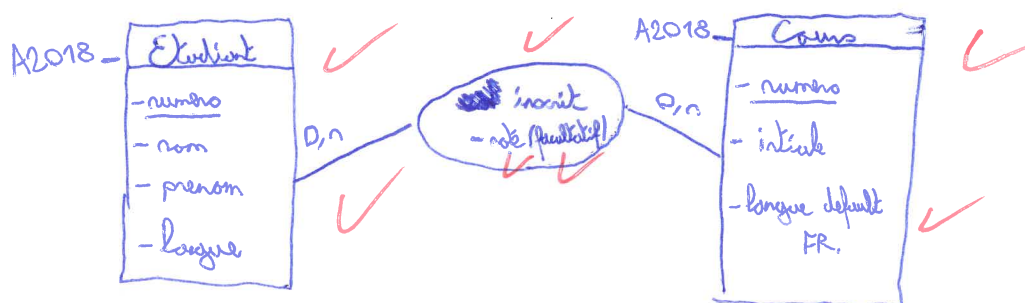
v. On ne peut rien dire cela dépend des valeurs des tuples

(b) Ecrire une contrainte, réaliste de votre choix, relative au schéma précédent qu'il est impossible d'implémenter via le SQL LDD. ~~On ne peut avoir que 5 personnes inscrites par cours (ou activité).~~

On ne peut avoir que 5 personnes inscrites par cours (ou activité).

Exercice 2 Modélisation EA

Donnez le modèle EA qui correspond exactement au modèle relationnel donné en annexe.



Exercice 3 SQL LMD

On reprend le schéma relationnel :

1. Soit la requête sql suivante avec la réponse retournée par oracle

```
SELECT T1.numero, T3.numero
FROM A2018_Etud T1, A2018_Inscription T2,
      A2018_Cours T3
WHERE T1.numero = T2.numero AND
      T3.numero = T2.numroc AND
      T1.langue = 'FR' AND
      T3.langue <> 'FR';
```

-- Réponse de oracle

T1.numero T3.numero

1 10

- (a) Donnez le remplissage **minimum** que chacune des tables de la base doit avoir pour que le résultat affiché sous la requête dans le script précédent, soit correct. Mettre la réponse aux emplacements prévus ci-dessous. VVV

--Contenu de la table A2018_Etud

NUMERO NOM PRENOM LANGUE

1 'Zamba' 'Yannick' 'FR'

--Contenu de la table A2018_Cours

NUMERO INTITULE LANGUE

10 'Telecom' 'EN'

--Contenu de la table A2018_Inscription

NUMEROE NUMEROC NOTE

1 10 null

On veut la liste des étudiants francophones inscrits à des cours ^{donnés dans} une langue autre que le français.

On affiche les binômes "numéro étudiant - numéros cours"

- (b) **Bonus** : formulez la requête précédente en Français VV

~~On cherche à insérer les numéros des étudiants francophones~~
~~On veut la liste (= les numéros) des étudiants francophones~~
~~associés aux cours (= leurs numéros)~~

2. Ecrire les requêtes suivantes en SQL. Cet SQL doit être **direct** et **efficace** : pas d'utilisation de tables en trop ou d'opérateurs non nécessaires. Soignez la présentation de vos requêtes en respectant l'indentation pour la lisibilité. Une requête mal présentée sera considérée comme fausse.

- (a) Affichez pour chaque langue le nombre total d'étudiants qui parle cette langue

VV

A2018-
SELECT langue, count(*) FROM Etudiant
Group By langue;

✓✓

- (b) Affichez pour chaque intitulé de cours le nombre total de langues dans lequel il est donné. On ne veut pas des cours qui ne sont donnés que dans une seule langue.

VVV

A2018-
SELECT intitulé, count(~~langue~~) FROM Cours
GROUP By intitulé HAVING count(~~langue~~) < 1;

Donc ✓✓
✓✓
Donc

- (c) Affichez les intitulés des cours qui ont au moins un inscrit. On veut deux versions : (version jointure et version in)

VV VV

A2018-
SELECT ~~intitulé~~ FROM Cours
WHERE Cours.numero IN
(SELECT numeroC FROM Inscription);

✓✓

A2018-
DISTINCT
SELECT Cours, intitulé FROM Cours, Inscription
WHERE Cours.numero = inscription.numeroC;

✓✓

(d) Mettre toutes les notes qui sont à NULL actuellement dans la base, à la valeur 1

VV

```
UPDATE table A2018 - inscription
SET note = 1
WHERE note is NULL;
```

✓✓

Exercice 4 Mise en place d'un schéma relationnel

Vous êtes sur le compte oracle t2-1 actuellement vide. Vous exécutez alors le script sql suivant :

```
CREATE TABLE A2018_Etud (
    NUMERO NUMBER(3,0),
    NOM CHAR(10) NOT NULL,
    PRENOM CHAR(10) NOT NULL,
    PRIMARY KEY (NUMERO));

CREATE TABLE a2018_cours (
    NUMERO NUMBER(2,0),
    INTITULE CHAR(10) NOT NULL,
    LANGUE CHAR(2) DEFAULT 'FR' NOT NULL,
    PRIMARY KEY (NUMERO)
);
```

Vous n'avez pas le droit de faire de DROP de ce qui vient d'être créé.

1. Ajoutez une contrainte à la table a2018_cours, pour qu'un intitulé de cours ne puisse pas être inséré 2 fois dans la même langue.

VV

```
ALTER TABLE a2018 - cours ADD
CONSTRAINT c1 UNIQUE (intitule, langue);
```

W

2. On vous demande :

- D'ajouter à la table a2018_etud une colonne langue obligatoire,
- qui par défaut vaut 'FR',
- Par ailleurs, la langue est soit 'FR' soit 'EN' soit 'DE' et rien d'autre.
- On vous ajoute que attention, cette dernière contrainte, on doit pouvoir l'enlever dans le futur sans problème. Et d'ailleurs, on vous demande d'écrire dans votre script, la commande SQL pour enlever cette contrainte et de la mettre sous forme de commentaire SQL.

VVVVV

ALTER TABLE A2018-etud ADD

Langue CHAR(2) ~~NOT NULL~~ DEFAULT 'FR' NOT NULL, ✓✓

ALTER TABLE A2018-etudiant ADD

Contraint langue1 CHECK (langue in ('FR', 'EN', 'DE')); ✓✓

- // ALTER TAB A2018-etudiant DROP Contraint langue1; ✓

3. Ecrire le script sql qui va vous permettre de créer la table a2018_inscription qui correspond à celle donnée dans le schéma relationnel de l'annexe. On vous ajoute que quand un cours (ou respectivement un étudiant) est enlevé (delete) de sa table alors les inscriptions à ce cours (resp. de cet étudiant) doivent automatiquement être enlevés également.

VVVVV

CREATE TABLE a2018 - inscription /

NumeroE Number(3,0) NOT NULL, ✓

NumeroC Number(2,0) NOT NULL,

note Integer NULL,

Contraint inoc1 PRIMARY KEY (NumeroE, NumeroC),

Contraint inoc2 FOREIGN KEY (NumeroE) REFERENCES a2018-etud(numero)
on delete cascade;

Contraint inoc3 FOREIGN KEY (NumeroC) REFERENCES a2018-Cours(numero)
on delete cascade

);

4. Ajouter une vue A2018_etud_EN qui permet de visualiser le numéro, le nom et le prénom des étudiants qui parle anglais ('EN').

VV

```
CREATE VIEW A2018-etud-EN AS (
  SELECT numero, nom, prenom FROM A2018-etud
  WHERE langue = 'EN');
```

W

5. On suppose que dans la table a2018_etud il n'y a qu'une ligne la suivante : (10, 'Sam', 'Smith', 'EN'). On suppose votre vue implémentée. Donnez le résultat retourné par oracle pour chacune des instructions du script ci-dessous : VVV

```
INSERT INTO a2018_etud_EN VALUES(1, 'Martin', 'Dupont');
```

```
-- ENTOURER LA BONNE REPONSE: OK ou ERREUR
```

```
-- Si ERREUR indiquez la cause:
```

Oracle veut ajouter l'entrée à son origine réelle → A2018-etud,

Or il lui manque une langue, qui est un champ obligatoire

il y a une langue par défaut donc OK

X

```
SELECT * FROM a2018_etud_EN;
```

```
-- ECRIRE CE QU'AFFICHE ORACLE
```

Numero	nom	prenom
10	SMITH ↑ 'SMITH'	SMITH ↑ 'SAM'

✓


```
SELECT * FROM a2018_etud;
-- ECRIRE CE QU'AFFICHE ORACLE
```

Numero	nom	prenom	Sargue
10	'Smith'	'SAM'	'EN'

✓

6. On suppose que pour l'instant vous êtes le seul ayant des droits sur ce schéma relationnel. Donnez les droits au compte i2_1, qui existe sur votre oracle, pour qu'il puisse voir et insérer des lignes dans la table a2018_etud. Il ne doit pas pouvoir passer ses droits à un autre compte. VV

```
SELECT,
GRANT INSERT ON a2018_etud TO i2_1;
```

✓✓