



Contrôle terminal de BDBIO - session 1

UCBL - Département informatique (2022/2023)

Pour assurer l'anonymat, ne notez pas votre nom ou numéro étudiant sur la copie.
Une feuille papier A4 autorisée. Durée : 1h30.
Remplissez les cases sur la dernière feuille, de préférence au stylo noir. Les questions avec un symbole ♣ ont 0, 1 ou plusieurs bonnes réponses. Les autres ont une unique bonne réponse.

Lassés de réfléchir à de nouveaux sujets d'examen chaque année, vos enseignants vous chargent de concevoir un outil permettant de générer des sujets à partir d'une base de données. Comme vous arrivez – exceptionnellement – en retard au cours, vous étudiez ce que les autres étudiant·e·s ont déjà réalisé.

Un premier groupe s'est lancé dans la modélisation relationnelle. Ci-dessous son schéma et les données de la table PROPOSER :

EXERCICES (*idE, énoncé, difficulté, durée, points*)
 QUESTIONS (*#idE, numéro, énoncé, points, #idT*)
 TYPESQUESTION (*idT, label*)
 CHOIXRÉPONSE (*idC, valeur*)
 NIVEAUX (*idNi, label*)
 NOTIONS (*idNo, label*)
 PROPOSER (*#idE, #numéro, #idC, estCorrect*)
 TRAVAILLER (*#idE, #numéro, #idNo, #idNi*)

idE	numéro	idC	estCorrect
1	1	1	0
1	1	2	1
1	2	3	0
1	2	4	0
2	1	1	1
2	1	3	1

D'autres groupes ont réalisé un document JSON (à gauche), une DTD pour XML (en haut à droite) et une description partielle de données RDF (en bas à droite).

```
{
  "exercices": [{
    "_id": 1,
    "titre": "Compréhension des sources"
    "points": 6,
    "énoncé": "Afin de vous familiariser avec les sources, répondez aux
    ↪ questions suivantes.",
    "difficulté": 2,
    "questions": [{
      "_id": "1-1",
      "type": "QCM",
      "énoncé": "Deux questions d'un même exercice
      ↪ peuvent-elles avoir le même énoncé ?",
      "choix": ["Oui", "Non"],
      "solutions": ["Vous espériez vraiment l'avoir ?"],
      "barème": "NBCx0.5-NMC"
    },{
      "_id": "1-2",
      "type": "QCM",
      "énoncé": "Une question peut-elle avoir plusieurs
      ↪ réponses correctes ?",
      "choix": ["Oui", "Non"],
      "solutions": ["Elle est facile... si on évite le
      ↪ piège."],
      "barème": "NBCx0.5-NMC"
    }
  ]
  }
}]
```

```
1 <!DOCTYPE generateur [
2 <!ELEMENT question (énoncé, choix?, type)>
3 <!ELEMENT énoncé (#PCDATA)>
4 <!ATTLIST question solution IDREFS>
5 <!ELEMENT choix (#PCDATA)>
6 <!ATTLIST choix id ID>
7 <!ELEMENT type EMPTY>
8 <!ATTLIST type label CDATA>
9 ]>
```

```
@prefix bd:<http://bd.db> .
@prefix rdf:<http://www.w3.org/rdf/> .
@prefix rdfs:<http://www.w3.org/rdfs/> .
@prefix xsd:<http://www.w3.org/xsd/> .

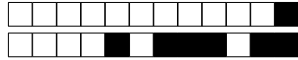
bd:type rdfs:domain bd:Question .
bd:type rdfs:range xsd:string .
bd:propose rdfs:domain bd:Question .
bd:propose rdfs:range bd:Choix .
bd:valeur rdfs:domain bd:Choix .
bd:valeur rdfs:range xsd:string .
bd:estCorrect rdfs:domain bd:Choix .
bd:estCorrect rdfs:range xsd:boolean .
```

Exercice 1 – Compréhension des sources (6 points)

Aidez les différents groupes à progresser en répondant aux questions suivantes.

Question 1 Dans le schéma relationnel, deux questions d'un même exercice peuvent-elles avoir le même énoncé ?

Non Oui



Question 2 Dans le schéma relationnel, une question peut-elle avoir plusieurs réponses correctes ?

Non Oui

Question 3 Dans le diagramme E/A qui produirait le schéma relationnel fourni, comment sont reliées les entités dérivées des relations EXERCICES et QUESTIONS ?

Association ternaire avec 1,1 Association réflexive Entité faible
 Agrégation Association ternaire sans 1,1 Spécialisation
 Association binaire avec 1,1 Association binaire sans 1,1 Autre réponse

Question 4 Dans le diagramme E/A qui produirait le schéma relationnel fourni, comment est représentée TRAVAILLER ?

Entité faible Association quaternaire avec 1,1 Association ternaire avec 1,1
 Association ternaire sans 1,1 Spécialisation Association binaire avec 1,1
 Association quaternaire sans 1,1 Association binaire sans 1,1 Autre réponse

Question 5 ♣ Quelle(s) ligne(s) de la DTD ne permettent pas de stocker les mêmes données que celles contenues dans le schéma relationnel ?

Définition de type (lignes 7-8) Définition de solution (ligne 4)
 Définition de question (ligne 2) Définition de énoncé (ligne 3)
 Définition de choix (lignes 5-6) Aucune

Question 6 ♣ D'après la source RDF, quel(s) nouveau(x) triplet(s) pouvez-vous déduire ?

bd :Question bd :type xsd :string bd :Question rdf :type rdfs :Class
 xsd :boolean rdf :type rdfs :Class bd :estCorrect rdf :type rdfs :Class
 bd :Choix rdfs :range bd :valeur bd :valeur rdf :type rdf :Property

Question 7 Combien de tuples sont retournés par la requête suivante sur les données relationnelles ?
SELECT DISTINCT idE, numéro FROM PROPOSER WHERE idC > numéro;

2 4 1 5
 0 6 3 Autre

Question 8 Combien de tuples sont retournés par la requête suivante sur les données relationnelles ?
SELECT COUNT(p1.estCorrect) FROM PROPOSER p1 INNER JOIN PROPOSER p2 ON p1.idE = p2.idE;

20 1 2 0 12
 16 8 24 4 Autre

Question 9 ♣ Quels sont le(s) tuple(s) résultat de la requête suivante sur les données relationnelles ?
SELECT idC - idE FROM PROPOSER WHERE estCorrect NOT IN (SELECT idE FROM PROPOSER) UNION
SELECT idC + estCorrect FROM PROPOSER p1 WHERE NOT EXISTS (SELECT * FROM PROPOSER p2 WHERE p1.numéro <= p2.estCorrect) ;



6
 -2

3
 0

-1
 5

2
 1

4
 Aucun

Exercice 2 – Interrogation des sources (6 points)

Écrivez les requêtes suivantes dans le langage demandé sur la source correspondante.

Question 10 L'identifiant des notions contenant *SQL* (dans leur label) et proposées à un niveau dont le label vaut *appliquer* (SQL). Le résultat sera trié par ordre décroissant.

0 0.25 0.5 0.75 1 1.25 1.5 *Reservé*

```
SELECT n.idNo
FROM NOTIONS no NATURAL JOIN TRAVAILLER NATURAL JOIN NIVEAUX ni
WHERE no.label LIKE '%SQL%' AND ni.label = 'appliquer'
ORDER BY n.idNo DESC;
```

Question 11 Pour chaque exercice ayant plus de 2 questions, donner le nombre de types (de question) différents ainsi que le nombre de points minimum et maximum de ses questions (SQL).

0 0.25 0.5 0.75 1 1.25 1.5 *Reservé*

```
SELECT idE, COUNT(DISTINCT idT) AS nb, MIN(points), MAX(points)
FROM QUESTIONS
GROUP BY idE
HAVING COUNT(*) > 1;
```

Question 12 Pour chaque question, affichez son énoncé puis les réponses correctes (XQUERY). Le résultat sera de la forme `<p>énoncéréponse1réponse2</p>`. La fonction `id(seq_idref)` retourne les noeuds dont l'identifiant est dans la séquence `seq_idref`.

0 0.25 0.5 0.75 1 1.25 1.5 *Reservé*

```
for $q in //question
  let $solutions := id($q/solution)
  return <p> { $q/enonce/text() }
    {
      for $s in $solutions
        return <em>{ $s/text() } </em>
    }
  </p>
```



Question 13 ♣ Les enseignants veulent connaître les solutions des questions de type QCM. Les étudiant-e-s proposent des solutions en pagaille. Mettez tout le monde d'accord en cochant les triplets manquants (SPARQL).
SELECT ?v WHERE { ... }

- | | | |
|---|---|---|
| <input type="checkbox"/> ?t rdf:type ?q | <input checked="" type="checkbox"/> ?q bd:propose ?c | <input type="checkbox"/> ?c rdf:type "true" |
| <input type="checkbox"/> FILTER(bd:estCorrect "true") | <input checked="" type="checkbox"/> ?c bd:estCorrect "true" | <input checked="" type="checkbox"/> ?q bd:type ?t |
| <input type="checkbox"/> FILTER(?q = "QCM") | <input type="checkbox"/> ?v bd:valeur ?c | <input type="checkbox"/> ?q rdf:type bd:QCM |
| <input type="checkbox"/> ?v bd:estCorrect ?c | <input type="checkbox"/> ?c bd:estCorrect ?v | <input type="checkbox"/> ?t rdf:type bd:QCM |
| <input type="checkbox"/> ?c bd:propose ?q | <input type="checkbox"/> FILTER(?c = "true") | <input checked="" type="checkbox"/> ?c bd:valeur ?v |
| <input type="checkbox"/> ?q rdf:type ?t | <input checked="" type="checkbox"/> FILTER(?t = "QCM") | <input type="checkbox"/> ?t bd:propose ?q |
| | | <input type="checkbox"/> ?t bd:type ?q |

Exercice 3 – Bonus (1 point)

Question 14 ♣ D'après une étude Career Builder de 2018, quelles sont les pires excuses pour justifier un retard ?

- | | |
|--|---|
| <input checked="" type="checkbox"/> Un astrologue avait prédit un accident sur l'auto-route, donc j'ai pris les petites routes | <input checked="" type="checkbox"/> Je pensais qu'Halloween était un jour férié |
| <input type="checkbox"/> J'ai malencontreusement marché sur une plaque de béton frais | <input type="checkbox"/> Un oiseau est rentré dans ma maison, j'ai mis du temps à le faire sortir |
| | <input checked="" type="checkbox"/> J'étais présent, mais je me suis endormi au parking |

Exercice 4 – Intégration de données (8 points)

Finalement, vous décidez de conserver la BD relationnelle, que vous trouvez plus simple, et la BD RDF, qui plait à vos enseignants.

Pour répondre aux questions suivantes, vous utiliserez les langages de requête appropriés en fonction des sources. Le reste du programme informatique sera codé en pseudo-langage (syntaxe libre, mais suffisamment explicite pour que le programme soit implémentable). Pour le niveau de détail, utilisez des appels de fonctions pour simplifier le code (e.g., si vous devez trier un tableau, écrivez `tab_sorted = sort(tab)` accompagné d'un commentaire mais n'écrivez pas un algorithme complet de tri de tableau!). Utilisez des commentaires, par exemple pour expliquer comment vous résolvez les conflits.

Question 15 Écrivez tout d'abord un programme pour intégrer les données JSON dans la BD relationnelle. Considérez que vous avez un dictionnaire Python pour la source de données JSON.



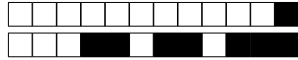
```
json = connexion(source2.json)
bdrel = connexion(source1)

for exo in json['exercices']: # pour chaque exo
    req = "insert into Exercices values (NULL, '{exo['énoncé']}', {exo['difficulté']}, NULL, {exo['points']})"
    idexo = bdrel.executer(req)

    for q in exo['questions']: # pour chaque question
        req = "select idT from TypesQuestion where label = {q['type']}"
        res = bdrel.executer(req)
        if not res : # type inexistant, insertion
            idt = bdrel.executer("insert into TypesQuestion values (NULL, {q['type']})")
        else:
            idt = res[0][0]

        # insertion question
        numq = q['_id'].partition('-')[2]
        req = "insert into Questions values ({idexo}, {numq}, '{q['énoncé']}', NULL, idt)"
        idexo = bdrel.executer(req)

        # insertion choix et proposer (lien question-choix)
        solutions = q['solutions']
        for ch in q['choix'] : # pour chaque réponse
            req = "select idC from ChoixReponse where valeur = {ch}"
            res = bdrel.executer(req)
            if not res : # choix inexistant, insertion
                idch = bdrel.executer("insert into ChoixReponse values (NULL, {ch})")
            else:
                idch = res[0][0]
            correct = 1 if ch in solutions else 0
            req = "insert into Proposer values ({idexo}, {numq}, {idch}, {correct})"
```



Question 16 Écrivez ensuite un programme qui retourne l'identifiant des questions et leur nombre de réponses, en interrogeant les sources relationnelles et RDF.

0 0.5 1 1.5 2 2.5 3 *Reservé*

```
bdrel = connexion(source1)
bdrdf = connexion(source4)

# récupération données relationnelles
req = "select CONCAT(idE, numéro) AS id, COUNT(*) from Proposer GROUP BY idE, numéro"
res1 = bdrel.executer(req)

# récupération des données RDF
req = ""select ?q (COUNT(?v) AS ?nbc)
      where {
          ?q bd:propose ?c .
          ?c bd:valeur ?v .
      }
      group by ?q""
res2 = bdrdf.executer(req)

union = set(res1) | set(res2)
return union
```