

## LABD

avril 2017

### Initiation à SPARQL

Pour réaliser les TP sur SPARQL, nous utiliserons le logiciel Corese<sup>1</sup> développé par une équipe d'Inria. La version à utiliser (3.1.1) est disponible sur le portail. Il s'agit d'un jar exécutable.

Les fichiers à utiliser pour ce premier TP sur SPARQL sont à récupérer dans une archive également disponible sur le portail. Le seul format accepté par Corese est le format RDF/XML ; une version Turtle des documents RDF est malgré tout également disponible dans l'archive.

#### Exercice 1 : Interrogation de l'instance RDF seule

Lancez le programme Corese. L'interface propose deux onglets : (System) pour charger des fichiers RDF et visualiser les traces d'exécution, et (+) pour charger ou écrire des requêtes et visualiser leur résultat. Chargez dans Corese les triplets définis dans le fichier `human_2007_09_11.rdf`<sup>2</sup> en utilisant le menu File, item 3-Load RDF. La fenêtre de log vous confirme le chargement. Cliquez ensuite sur l'onglet + : une requête par défaut Query1 est alors créée :

```
SELECT ?x ?t WHERE {  
  x rdf:type ?t  
}
```

*Remarque : On peut noter que le préfixe `rdf:` est défini par défaut. C'est aussi le cas des préfixes `rdfs:` et `xsd:`. Par ailleurs, il est possible de sauvegarder ses requêtes via le menu File.*

Question 1 : Que fait cette requête ? Vérifiez en exécutant la requête (bouton Query) qui fournit sa réponse au format XML en liant (élément `binding`) variables (attribut `name`) et valeurs (contenu de l'élément `binding`). Combien de réponses obtenez-vous ? Expliquez ce résultat. Retrouvez John et son type.

Question 2 : En une phrase traduisez ce que demande la requête SPARQL suivante :

```
PREFIX humans: <http://www.inria.fr/2007/09/11/humans.rdfs#>  
SELECT * WHERE {  
  ?x humans:hasSpouse ?y  
}
```

Combien de réponses obtenez-vous ?

Question 3 : Regardez dans le fichier RDF quelle est la propriété utilisée pour donner l'âge d'une personne. Donnez une requête pour trouver les personnes qui ont plus de 20<sup>3</sup> ans.

Question 4 : Regardez dans le fichier RDF<sup>4</sup> quelle est la propriété utilisée pour donner la pointure des chaussures des personnes.

---

1. <http://wimmics.inria.fr/corese>

2. On ne charge pas l'autre fichier pour l'instant.

3. la fonction `xsd:integer(param)` convertit son paramètre `param` de type chaîne de caractères en entier.

4. N'oubliez pas qu'une version Turtle est fournie.

1. En déduire une requête pour extraire toutes les personnes (**Person**) avec leur pointure.
2. Modifiez cette requête pour extraire toutes les personnes et, si elle est disponible, leur pointure.
3. Modifiez cette requête pour extraire toutes les personnes et, si elle est disponible, leur pointure à condition que celle-ci soit supérieure à 8.
4. Écrire une requête pour extraire toutes les personnes dont la pointure est supérieure à 8 ou dont la taille de chemise est supérieure à 12 (3 réponses).

Question 5 : Dans la réponse précédente, repérez l’URI de **John**.

1. Formulez une requête pour trouver toutes les propriétés de **John**.
2. Demandez une description de **John** au moteur en utilisant la clause SPARQL prévue pour cela.

Question 6 : Regardez dans le fichier RDF quelle est la propriété utilisée pour donner les enfants d’une personne.

1. Donnez tous les couples (*parent, enfant*).
2. Formulez une requête pour trouver les personnes qui ont au moins un enfant.
3. Combien de réponses obtenez-vous ? Combien de doublons identifiez-vous dans ces réponses ?
4. Donnez un moyen d’éviter les doublons. Combien de réponses obtenez-vous alors ?
5. Donnez une requête pour trouver les hommes (**Man**) qui n’ont pas d’enfant<sup>5</sup>.
6. Quelles sont les femmes (**Woman**) mariées, avec éventuellement leurs enfants ?

Question 7 : Quels sont les couples de personnes qui ont la même taille de chemise (enlevez les redondances) ?

Question 8 : Construisez la clôture symétrique de la relation **hasFriend** en utilisant le bon énoncé SPARQL.

Question 9 : Recherchez toutes les personnes qui ne sont pas des hommes. Expliquez vos résultats.

## Exercice 2 : Interrogation du schéma RDFS seul

Question 1 : En visualisant le fichier, notez quel est l’espace de nom associé à l’ontologie qui y est décrite. Repérez les domaines et co-domaines des propriétés de cette ontologie. En particulier, sur quelles classes peut porter la propriété **âge** ?

Réinitialisez le moteur Corese (menu **Engine**, item **Reset**) puis chargez le fichier **human\_2007\_09\_11.rdfs** (et seulement lui).

Question 2 : Écrire une requête permettant de retrouver toutes les classes de l’ontologie.

Question 3 : Écrire une requête permettant de retrouver tous les liens **subClassOf** de l’ontologie.

Question 4 : Demandez les définitions et la traduction de **"shoe size"**.

Question 5 : Trouvez les synonymes du terme **"personne"** en Français. Quelles sont les réponses ?

---

5. la fonction **bound(?x)** à retour booléen permet de savoir si la variable **?x** est liée à une valeur.

Question 6 : Trouvez les différents sens en français des termes "size" et "homme" (homonymie). Quelles sont les réponses ?

### Exercice 3 : Utilisation conjointe du schéma et de l'instance

Question 1 :

1. Réinitialisez de nouveau le moteur Corese puis rechargez le fichier `human_2007_09_11.rdf` (et seulement lui). Créez et exécutez une requête pour obtenir toutes les ressources de type `Animal` et une autre pour obtenir toutes les ressources de type `Person`. Combien de réponses obtenez-vous ?
2. Sans réinitialiser le moteur Corese, chargez en plus le schéma `human_2007_09_11.rdfs` et exécutez de nouveau les deux requêtes précédentes. Expliquez le résultat.

Question 2 :

1. Écrire une requête permettant de retrouver les mâles (type `Male`) et leur épouse. Combien de réponses obtenez-vous ? Expliquez ce résultat.
2. Ajoutez dans l'instance RDF le fait que Lucas a pour père Karl. Réinitialisez Corese, rechargez l'ontologie et l'instance RDF puis redemandez les personnes de sexe masculin et leurs épouses. Expliquez le nouveau résultat.

Question 3 : Demandez tous les professeurs (`Lecturer`) et leur type. Combien de réponses obtenez-vous ? Regardez comment ce typage est déclaré dans l'instance RDF et expliquez le résultat. Demandez les instances communes de `Person` et `Male`. Regardez comment ce typage est déclaré dans l'ontologie et expliquez la présence de `John`.

Question 4 : Demandez les instances de la relation `hasAncestor`. Expliquez le résultat après avoir vérifié que cette propriété n'est pas utilisée dans le fichier instance.