

INF8410 - Exercices de SPARQL

1 Format d'une réponse

Quel est le format d'une réponse à une requête SPARQL ?

2 Régimes d'inférence

Qu'est-ce qu'un régime d'inférence en SPARQL ? Quels sont les régimes d'inférence définis en SPARQL 1.1 ?

3 Fédération de requêtes en SPARQL

En quoi consiste la fédération de requêtes en SPARQL et comment cela se fait-il ?

4 Langage de mise à jour

Décrivez en gros le langage de mise à jour de SPARQL 1.1.

5 Protocole HTTP de SPARQL

Décrivez en gros le protocole HTTP de SPARQL 1.1 et discutez de son utilité.

6 Régimes d'inférence

Soit le graphe RDF suivant :

```
ex:book1 rdf:type ex:Publication .
ex:book2 rdf:type ex:Article .
ex:Article rdfs:subClassOf ex:Publication .
ex:publishes rdfs:range ex:Publication .
ex:MITPress ex:publishes ex:book3 .
```

Proposez des requêtes qui font appel aux différents régimes d'inférences RDF possibles en SPARQL.

7 Requête SPARQL

Soit le graphe RDF suivant :

```
@prefix : <http://www.exemple.org/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix dbpedia: <http://dbpedia.org/resource/> .

:Cathedrale rdfs:subClassOf :Eglise .
:Eglise rdfs:subClassOf :LieuDeCulte .
:cathedraleAmos a :Cathedrale ;
    rdfs:label "Cathédrale d'Amos" ;
    :localisation :Abitibi ;
    :débutConstruction "1922"^^xsd:gYear ;
    :finConstruction "1923"^^xsd:gYear ;
    :mobilier [ a :Orgue ;
        :arrivee "1954"^^gYear ;
        :constructeur dbpedia:Casavant ] .
:ABC :annéeNaissance "1875"^^xsd:gYear ;
    foaf:name "Aristide Beaugrand-Champagne" ;
    :annéeDécès "1950"^^xsd:gYear ;
    :concepteurDe :cathedraleAmos .
```

Traduisez en SPARQL les requêtes suivantes :

1. À quels endroits en Abitibi (pas nécessairement des églises) des orgues de Casavant et Frères ont-ils été installés depuis 1950 ?
2. Parmi les bâtiments conçus par Aristide Beaugrand-Champagne, combien sont des lieux de culte ?
3. Combien de temps a-t-il fallu pour construire la cathédrale d'Amos ?
4. Parmi les bâtiments conçus par Aristide Beaugrand-Champagne, combien y en a-t-il pour chaque type de bâtiment, c'est-à-dire combien d'églises, combien d'écoles, etc ?

8 Requêtes SPARQL

a) Dites ce que l'on veut extraire avec la requête SPARQL suivante :

```
PREFIX dcterms: <http://purl.org/dc/terms/>

select distinct ?p where {
  [] dcterms:creator [] ;
  ?p ?z .
}
```

b) Dites ce que l'on veut extraire avec la requête SPARQL suivante :

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dcterms: <http://purl.org/dc/terms/>

select distinct ?t ?d where {
  ?y foaf:name "Victor Hugo".
  ?x dcterms:creator ?y ;
  dcterms:title ?t ;
  dcterms:date ?d .

  FILTER(xsd:date(?d) < "1840"^^xsd:dateTime)
}
```

9 Requête SPARQL

Soit la base de connaissances suivante :

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

_:p1 foaf:name "Paul" ;
     foaf:age "17"^^xsd:nonNegativeInteger ;
     foaf:mbox "paul@abc.com" .
_:p2 foaf:name "Ana" ;
     foaf:age "27"^^xsd:nonNegativeInteger ;
     foaf:mbox "ana@abc.com" .
_:p3 foaf:name "Léa" ;
     foaf:age "21"^^xsd:nonNegativeInteger ;
     foaf:mbox "lea@uuu.com" .
_:p4 foaf:name "Guy" ;
     foaf:age "19"^^xsd:nonNegativeInteger ;
     foaf:mbox "guy@uuu.com" .
_:p5 foaf:name "Gil" ;
     foaf:age "20"^^xsd:nonNegativeInteger ;
     foaf:mbox "gil@xyz.ca" .
_:p6 foaf:name "Lola" ;
     foaf:age "30"^^xsd:nonNegativeInteger ;
     foaf:mbox "lola@xyz.ca" .

_:p1 foaf:knows _:p2 .
_:p1 foaf:knows _:p3 .
_:p2 foaf:knows _:p1 .
_:p2 foaf:knows _:p3 .
```

a) Dites quel sera le résultat de chacune des requêtes suivantes :

Requête Q1 :

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT DISTINCT ?n
WHERE {
  ?x foaf:mbox ?m ;
     foaf:name ?n ;
     foaf:age ?a1 ;
     foaf:knows/foaf:age ?a2
  FILTER (?a2 > ?a1)
}
```

Requête Q2 :

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT ?dom (AVG(?age) as ?a)
WHERE {
    ?x foaf:age ?age ;
        foaf:name ?n ;
        foaf:mbox ?m .

    BIND (strafter(?m,'@') as ?dom)
}
GROUP BY ?dom
```

Remarque : la fonction *strafter(s,p)* retourne la sous-chaîne qui commence immédiatement après la sous-chaîne *p*.

Requête Q3 :

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT DISTINCT ?n
WHERE {
    {
        SELECT DISTINCT ?p
        WHERE {
            ?p foaf:age ?a
            FILTER (?a > 20)
        }
    }
    ?x foaf:name ?n .
    {?x foaf:knows ?p } UNION {?p foaf:knows ?x }
}
```

b) Écrivez une requête SPARQL qui permet de déterminer si le graphe des connaissances contient un cycle (c'est le cas dans notre exemple, puisque Paul connaît Ana, qui elle-même connaît Paul).

c) Écrivez une requête SPARQL pour extraire le nom des personnes qui ont plus de 18 ans et moins de 30 ans, qui connaissent quelqu'un, et dont le courriel n'est pas dans le domaine "uuu.com".

10 Ordre des triplets dans une requête SPARQL

Expliquez, en fournissant un exemple, pourquoi l'ordre des items dans une requête SPARQL peut influencer les performances lors de son exécution.