5/9/2016 EX4

martedì 5 gennaio 2021 21:12

Exercise 4

(a) Write an RDF/RDFS model representing the following statements about URIs `Person`, `Director`, `Actor`, `Writer`, `Movie`, `Country`, `Comedy`, `Drama`, `Man`, `Woman`, `filmedIn`, `hasBoxOfficeGross`, `isDirectorOf`, `isWriterOf`, `actsIn`, `bornIn`, `Joe`, `Mary`, `Ann`, `Paul`, `Italy`, `France`, `ABC`, `XYZ`.

1. `Person`, `Director`, `Writer`, `Actor`, `Country`, `Movie`, `Comedy`, `Drama`, `Man`, and `Woman` are classes;
2. `Man` and `Woman` are subclasses of `Person`;
3. `Comedy` and `Drama` are subclasses of `Movie`;

4. `actsIn`, `bornIn`, `filmedIn`, `isDirectorOf` and `isWriterOf` are properties;
5. `isDirectorOf` has domain `Director` and range `Movie`;
6. `flmedIn` has domain `Movie` and range `Country`;
7. `bornIn` has domain `Person` and range `Country`;
8. `actsIn` has domain `Actor` and range `Movie`;
9. `hasBoxOfficeGross` has domain `Movie` and range `xsd:integer`;
10. Ann is the director and the writer of movie XYZ;
11. Joe and Paul act in movie ABC;
12. ABC was filmed in France;
13. Ann is a woman;
14. Paul is a man.

(b) Write SPARQL queries corresponding to the following requests: (b1) "return the actors born in France who acted with Joe in at least one movie and, optionally, the country where the director of the movie was born in"; (b2) "return every director who directed at least one movie whose box office gross is above $\$10,000,000$".

a)

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema>
@prefix myns: <http://www.example.org/myVocabulary/>

1) myns:Person rdf:type rdfs:Class.
myns:Director rdf:type rdfs:Class.
myns:Writer rdf:type rdfs:Class.
myns:Actor rdf:type rdfs:Class.
myns:Country rdf:type rdfs:Class.

myns:Movie rdf:type rdfs:Class.
myns:Comedy rdf:type rdfs:Class.
myns:Drama rdf:type rdfs:Class.
myns:Man rdf:type rdfs:Class.
myns:Woman rdf:type rdfs:Class.

2) myns:Man rdfs:subClassOf myns:Person.
   myns:Woman rdfs:subClassOf myns:Person.
3) myns:Comedy rdfs:subClassOf myns:Movie.
   myns:Drama rdfs:subClassOf myns:Movie.
4) myns:actsIn rdf:type rdf:Property.
   myns:bornIn rdf:type rdf:Property.
   myns:filmedIn rdf:type rdf:Property.
   myns:isDirectorOf rdf:type rdf:Property.
   myns:isWriterOf rdf:type rdf:Property.
5) myns:isDirectorOf rdfs:domain myns:Director.
   myns:isDirectorOf rdfs:range myns:Movie.
6) myns:filmedIn rdfs:domain myns:Movie.
   myns:filmedIn rdfs:range myns:Country.
7) myns:bornIn rdfs:domain myns:Person.
   myns:bornIn rdfs:range myns:Country.
8) myns:actsIn rdfs:domain myns:Actor.
   myns:actsIn rdfs:range myns:Movie.
9) myns:hasBoxOfficeGross rdfs:domain myns:Movie.
   myns:hasBoxOfficeGross rdfs:range xds:integer.
10) myns:Ann myns:isDirectorOf myns:XYZ.
    myns:Ann myns:isWriterOf myns:XYZ.
11) myns:Joe myns:actsIn myns:ABC.
    myns:Paul myns:actsIn myns:ABC.
12) myns:ABC myns:filmedIn myns:France.
13) myns:Ann rdf:type myns:Woman.

14) myns:Paul rdf:type myns:Man.

b)

(b1)

```
PREFIX
    rdf
    myns
SELECT ?a ?c
WHERE{
    ?a myns:bornIn myns:France.
    ?a myns:actsIn ?m.
    myns:Joe myns:actsIn ?m.
    OPTIONAL{
        ?d myns:isDirectorOf ?m.
        ?d myns:bornIn ?c
    }.
}
```

(b2)

```
PREFIX
    rdf
    myns
SELECT ?d
WHERE{
    ?d myns:isDirectorOf ?m.
    ?m myns:hasBoxOfficeGross ?z.
    FILTER {?z > 10,000,00}.
}
```