Semantic Web handout
lecture questions and practical sessions

In this document, you must provide your answers to the questions asked during the course **and** to the questions of the practical sessions; everything in one document. The questions of the course have been repeated here; do not delete the questions but provide your answer to each question just below the question. At the end, you must generate and submit only one final PDF file based on this template. In questions where you are asked to create, invent or use your own data, make sure they are different from other student's.

Day 01: questions from the course.

Q1.1 Practice XML replace missing parts

```
<book>
 <title>Architecture Now</title>
 <author>Jodidio, Philip</author>
 <ID isbn10="3822840912" />
</book>
```

Q1.2 Provide 10 first lines

Great White Shark

http://www.bbc.co.uk/nature/life/Great_white_shark

http://www.bbc.co.uk/nature/life/Great_white_shark.rdf

http://www.bbc.co.uk/nature/life/Great_white_shark

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd"><html
xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-GB">
<head profile="http://dublincore.org/documents/dcq-html/">

        <title>BBC Nature - Great white shark videos, news and facts</title>
<meta name="description" content="Great white sharks are at the very top of the
marine food chain. " />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="schema.dcterms" href="http://purl.org/dc/terms/" />
                                <meta name="DCTERMS.created" content="2009-07-
28T12:00:00Z" />
        <meta name="DCTERMS.modified" content="2014-01-29T14:47:18+00:00" />
```

http://www.bbc.co.uk/nature/life/Great_white_shark.rdf

```
<?xml version="1.0" encoding="utf-8"?><rdf:RDF
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/terms/"
    xmlns:dctypes="http://purl.org/dc/dcmitype/"
    xmlns:skos="http://www.w3.org/2004/02/skos/core#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:po="http://purl.org/ontology/po/"
```

Q1.3 DBpedia

1. Find “London” on DBpedia.org; e.g. Google: "london site:dbpedia.org“
   make sure you are on the English chapter (dbpedia.org) as there are many others (fr.dbpedia.org, de.dbpedia.org)
2. Find dbp:populationDemonym and give its value
3. Find rdf:type and click on value yago:WikicatCapitalsInEurope
4. Find “Vienna” and get its URI
   (careful: with content negotiation and redirection, the URL of the page you are currently viewing may be different from the URI of the resource it describes)
5. Access to Vienna and find its native name?

1. http://dbpedia.org/page/London
2. dbp:populationDemonym   Londoner
3. http://dbpedia.org/class/yago/WikicatCapitalsInEurope
4. http://dbpedia.org/resource/Vienna
5. Wien

Q1.4 CURL

1. Ten first lines:

```
curl -o Paris.html -L -H "Accept: text/html" http://dbpedia.org/resource/Paris

curl -o Paris-rdf-xml.txt -L -H "Accept: application/rdf+xml" http://dbpedia.org/resource/Paris
```

2. Ten first lines for HTML and RDF http://ns.inria.fr/fabien.gandon#me

3. Ten first lines for HTML and RDF for ‘Vienna’ on Dbpedia

4. Ten first lines for HTML and RDF http://www.bbc.co.uk/nature/life/Great_white_shark

5. Ten first lines for HTML and RDF
   http://purl.uniprot.org/uniprot/P43121

6. What is the topic and format of data obtained with
   curl -o json.txt -L -H "Accept: application/json" https://www.wikidata.org/wiki/Special:EntityData/Q551861

7. What is the topic and format of data obtained with
   curl -o turtle.txt -L -H "Accept: text/turtle" http://dx.doi.org/10.1007/3-540-45741-0_18

1)

curl -o Paris.html -L -H "Accept: text/html" http://dbpedia.org/resource/Paris

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:dbpprop="http://dbpedia.org/property/"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    version="XHTML+RDFa 1.0"
    xml:lang="en">
```

curl -o Paris-rdf-xml.txt -L -H "Accept: application/rdf+xml" http://dbpedia.org/resource/Paris

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF
     xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
     xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
     xmlns:owl="http://www.w3.org/2002/07/owl#"
     xmlns:dbo="http://dbpedia.org/ontology/"
     xmlns:dct="http://purl.org/dc/terms/"
     xmlns:foaf="http://xmlns.com/foaf/0.1/"
     xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#"
     xmlns:prov="http://www.w3.org/ns/prov#"
```

2)

curl -L -H "Accept: text/html" http://ns.inria.fr/fabien.gandon#me -o tmp ; head tmp

```
<?xml version="1.0" encoding="utf-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
 <head>
  <meta http-equiv="Content-Type" content="text/html;charset=utf-8" />
  <title>FOAF profile of Fabien GANDON</title>
 </head>
<body>
 <h1>FOAF profile of Fabien GANDON</h1>
 <p>You may have been redirected here by your browser.</p>
```

curl -L -H "Accept: application/rdf+xml" http://ns.inria.fr/fabien.gandon#me -o tmp ; head tmp

```
<?xml version='1.0' encoding='utf-8' ?>
<rdf:RDF
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
      xmlns:foaf="http://xmlns.com/foaf/0.1/"
          xml:base="http://ns.inria.fr/fabien.gandon">

          <foaf:PersonalProfileDocument rdf:about="">
            <foaf:maker rdf:resource="#me"/>
                <foaf:primaryTopic rdf:resource="#me"/>
```

3)

curl -L -H "Accept: text/html" http://dbpedia.org/resource/Vienna -o tmp ; head tmp

```
<?xml version="1.0" encoding="UTF-8" ?>
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
```

```
<html xmlns="http://www.w3.org/1999/xhtml"
    xmlns:dbpprop="http://dbpedia.org/property/"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    version="XHTML+RDFa 1.0"
    xml:lang="en">
```

curl -L -H "Accept: application/rdf+xml" http://dbpedia.org/resource/Vienna -o tmp ; head tmp

```
<?xml version="1.0" encoding="utf-8" ?>
<rdf:RDF
        xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
        xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
        xmlns:owl="http://www.w3.org/2002/07/owl#"
        xmlns:dbo="http://dbpedia.org/ontology/"
        xmlns:dbp="http://dbpedia.org/property/"
        xmlns:dct="http://purl.org/dc/terms/"
        xmlns:foaf="http://xmlns.com/foaf/0.1/"
        xmlns:skos="http://www.w3.org/2004/02/skos/core#"
```

4)

curl -L -H "Accept: text/html" http://www.bbc.co.uk/nature/life/Great_white_shark -o tmp ; head tmp

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd"><html
xmlns="http://www.w3.org/1999/xhtml" xml:lang="en-GB">
<head profile="http://dublincore.org/documents/dcq-html/">

        <title>BBC Nature - Great white shark videos, news and facts</title>
<meta name="description" content="Great white sharks are at the very top of
the marine food chain. " />
<meta http-equiv="Content-Type" content="text/html; charset=UTF-8" />
<link rel="schema.dcterms" href="http://purl.org/dc/terms/" />
                                <meta name="DCTERMS.created" content="2009-
07-28T12:00:00Z" />
        <meta name="DCTERMS.modified" content="2014-01-29T14:47:18+00:00" />
```

curl -L -H "Accept: application/rdf+xml" http://www.bbc.co.uk/nature/life/Great_white_shark -o tmp ; head tmp

```
<?xml version="1.0" encoding="utf-8"?><rdf:RDF
    xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:owl="http://www.w3.org/2002/07/owl#"
    xmlns:foaf="http://xmlns.com/foaf/0.1/"
    xmlns:dc="http://purl.org/dc/terms/"
    xmlns:dctypes="http://purl.org/dc/dcmitype/"
    xmlns:skos="http://www.w3.org/2004/02/skos/core#"
    xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
    xmlns:po="http://purl.org/ontology/po/"
```

5)

curl -L -H "Accept: text/html" http://purl.uniprot.org/uniprot/P43121 -o tmp ; head tmp

```
<!DOCTYPE html SYSTEM "about:legacy-compat">
<html xmlns="http://www.w3.org/1999/xhtml" lang="en"
xml:lang="en"><head><title>MCAM - Cell surface glycoprotein MUC18 precursor
- Homo sapiens (Human) - MCAM gene &amp; protein</title><meta
content="IE=edge" http-equiv="X-UA-Compatible"/><meta content="text/html;
```

```
charset=UTF-8" http-equiv="Content-Type"/><meta content="width=device-width,
initial-scale=1" name="viewport"/><link href="/" rel="home"/><link
href="http://creativecommons.org/licenses/by-nd/3.0/" rel="license"/><link
type="image/vnd.microsoft.icon" href="/favicon.ico" rel="shortcut
icon"/><link href="/uniprot.min.css2018_01" type="text/css"
rel="stylesheet"/><script type="text/javascript">
```

curl -L -H "Accept: application/rdf+xml" http://purl.uniprot.org/uniprot/P43121 -o tmp ; head tmp

```
<?xml version='1.0' encoding='UTF-8'?>
<rdf:RDF xml:base="http://purl.uniprot.org/uniprot/"
xmlns="http://purl.uniprot.org/core/"
xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:skos="http://www.w3.org/2004/02/skos/core#"
xmlns:bibo="http://purl.org/ontology/bibo/"
xmlns:foaf="http://xmlns.com/foaf/0.1/"
xmlns:void="http://rdfs.org/ns/void#" xmlns:sd="http://www.w3.org/ns/sparql-
service-description#" xmlns:faldo="http://biohackathon.org/resource/faldo#">
```

6)

curl -o json.txt -L -H "Accept: application/json" https://www.wikidata.org/wiki/Special:EntityData/Q551861

The data format obtain is JSON. It is large file, 40Kb. It contains resource information about Xavier Dolan.

7)

curl -o turtle.txt -L -H "Accept: text/turtle" http://dx.doi.org/10.1007/3-540-45741-0_18

The file is a Turtle file that describe an article named “Distributed Artificial Intelligence for Distributed Corporate Knowledge Management” that was published in 2002 in the book “Cooperative Information Agents VI” from page 202 to 217.

It authors were Fabien Gandon and Rose Dieng-Kuntz.

Q1.5 Recall five best practices of linked open data

On the web, open license
Machine readable data
Non proprietary format
RDF standards
Linked RDF

Q1.6 Spotlight demo

Reproduce the demo:

1. Copy a text from Wikipedia (e.g. Muse Band page)
2. Find the DBpedia Spotlight service page
3. Paste the text and run the detection
4. Try with other texts and copy-paste one of the results you get.

1) From https://en.wikipedia.org/wiki/U2 :
   U2 are an Irish rock band from Dublin formed in 1976. The group consists of Bono (lead vocals and rhythm guitar), the Edge (lead guitar, keyboards, and backing vocals), Adam Clayton (bass guitar), and Larry Mullen Jr. (drums and percussion). Initially rooted in post-punk, U2's musical style evolved throughout their career, yet has maintained an anthemic sound built on Bono's expressive vocals and the Edge's effects-based guitar textures. Their lyrics, often embellished with spiritual imagery, focus on personal and sociopolitical themes. Popular for their live performances, the group have staged several ambitious and elaborate tours over their career.
2) http://demo.dbpedia-spotlight.org/
3) Done and clicked on Annotate:
4) Here is the output :
   U2 are an Irish rock band from Dublin formed in 1976. The group consists of Bono (lead vocalsand rhythm guitar), the Edge (lead guitar, keyboards, and backing vocals), Adam Clayton (bass guitar), and Larry Mullen Jr. (drums and percussion). Initially rooted in post-punk, U2's musical style evolved throughout their career, yet has maintained an anthemic sound built on Bono's expressive vocals and the Edge's effects-based guitar textures. Their lyrics, often embellished with spiritual imagery, focus on personal and sociopolitical themes. Popular for their live performances, the group have staged several ambitious and elaborate tours over their career.

Day 02: questions from the course on RDF.

Q2.0 What is the mathematical structure built by the RDF triples?
(give the type of structure and its definition/explanation)

The mathematical structure built by RDF triples (i.e. subject, predicate, object) is a directed labeled multigraph where:

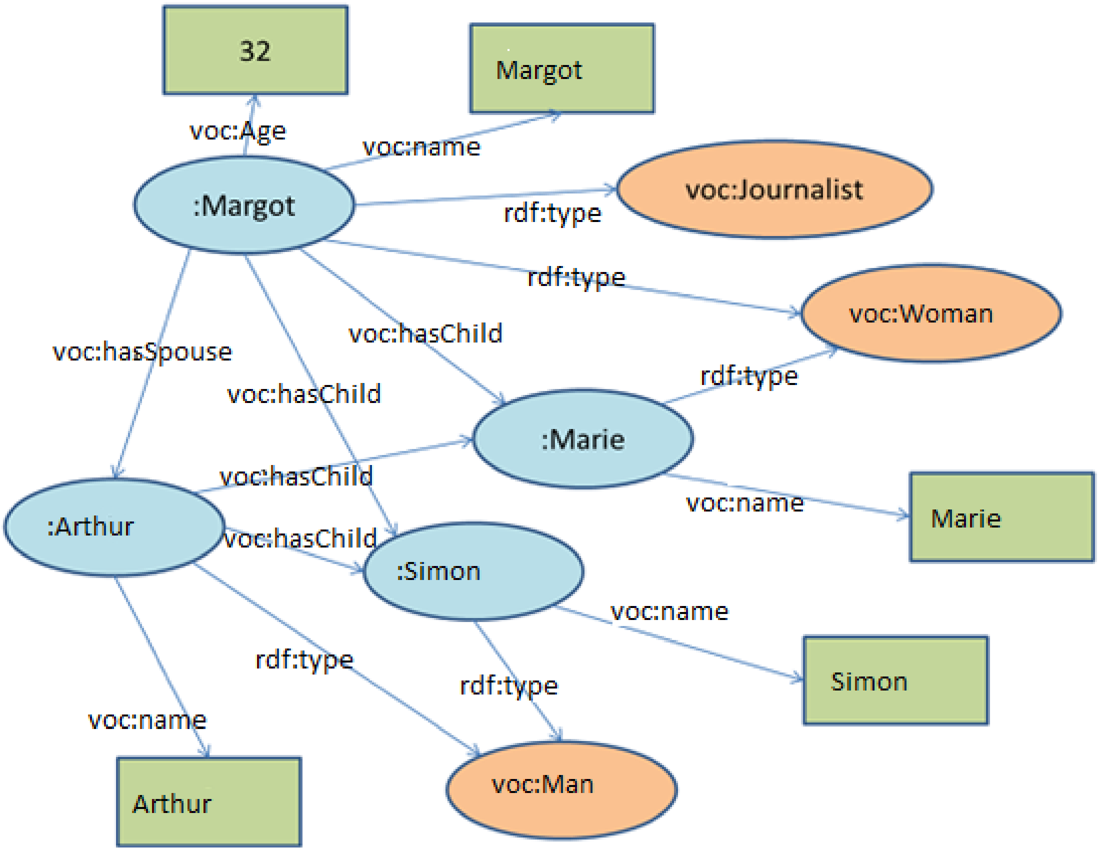
- The starting node is the subject,
- The arrival node is the object
- The edge between them is the predicate that links the subject and the object.

Q2.1 Fill the blanks

"Margot is a journalist woman, 32 years old, married to Arthur who is a man with whom she had two children: Marie who is a woman and Simon who is a man".

To fill the blanks we use the values: `:Arthur, :Simon, voc:name, voc:hasChild, voc:age, voc:hasSpouse, rdf:type, voc:Woman, voc:Man, "Margot", "Arthur", "Marie", "Simon"`

For each person we also explicitly specify the name



Q2.2 Fill the blanks (RDF/XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [   <!ENTITY vocabulaire "http://www.unice.fr/voc">
<!ENTITY xsd "http://www.w3.org/2001/XMLSchema#"> ]>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:voc="&vocabulaire;#" xml:base="http://www.unice.fr/data">
 <voc:women rdf:about="#Margot">
    <voc:name>Margot</voc:name>
    <voc:age
rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">32</voc:age>
    <voc:hasSpouse rdf:resource="#Arthur"></voc:hasSpouse>
    <voc:hasChild rdf:resource="#Simon"></voc:hasChild>
    <voc:hasChild>
      <rdf:Description rdf:about="#Marie">
        <voc:name>Marie</voc:name>
        <rdf:type rdf:resource="&vocabulaire;#Woman"></rdf:type>
      </rdf:Description>
    </voc:hasChild>
    <rdf:type rdf:resource="&vocabulaire;#Journalist"></rdf:type>
 </voc:women>
 <voc:man rdf:about="#Arthur">
    <voc:name>Arthur</voc:name>
    <voc:hasChild rdf:resource="#Simon"></voc:hasChild>
    <voc:hasChild rdf:resource="#Marie"></voc:hasChild>
  </voc:man>
  <voc:Man rdf:about="#Simon">
    <voc:name>Simon</voc:name>
  </voc:Man>
</rdf:RDF>
```

Q2.3 Fill the blanks (N3/Turtle)

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix voc: <http://www.unice.fr/voc#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
<http://www.unice.fr/data#Margot> a voc:Journalist , voc:women ;
    voc:age "32"^^xsd:string ;
    voc:hasChild <http://www.unice.fr/data#Marie>,
<http://www.unice.fr/data#Simon>;
    voc:hasSpouse <http://www.unice.fr/data#Arthur> ;
    voc:name "Margot" .
<http://www.unice.fr/data#Arthur> a voc:Man ;
    voc:hasChild <http://www.unice.fr/data#Marie>,
        <http://www.unice.fr/data#Simon> ;
    voc:name "Arthur" .
<http://www.unice.fr/data#Marie> a voc:Woman ;
    voc:name "Marie" .
<http://www.unice.fr/data#Simon> a voc:Man ;
     voc:name "Simon" .
```

Q2.4 Visit me please

1. Get the RDF data from: http://ns.inria.fr/fabien.gandon#me
2. What is the syntax used?
3. Validate it and see the graph:
   http://www.w3.org/RDF/Validator/
4. Translate into Turtle/N3:
   http://rdf-translator.appspot.com/
   http://www.easyrdf.org/converter
5. Visualize it also with:
   https://graves.cl/visualRDF/
   http://www.easyrdf.org/converter (PNG, SVG)
6. Adapt to your data and do it again

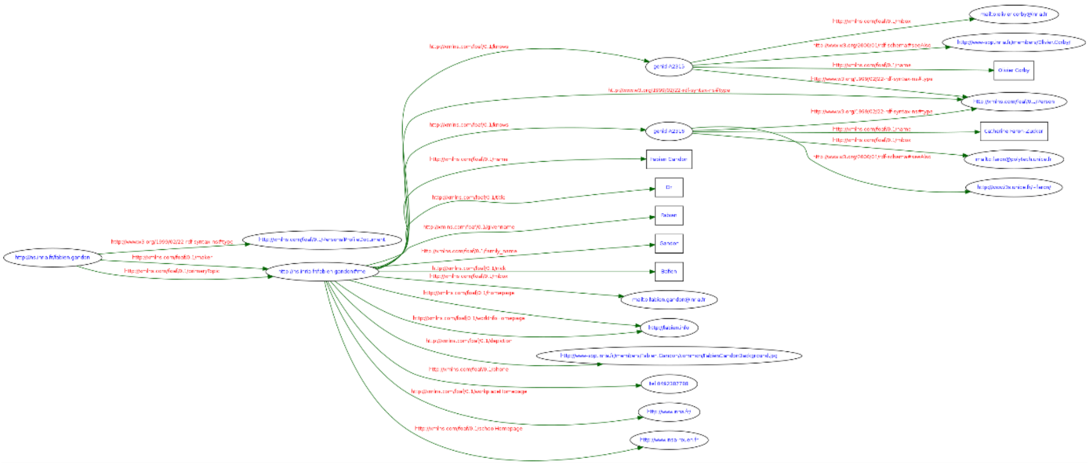
1) http://ns.inria.fr/fabien.gandon/foaf.html#me RDF profile

```
<?xml version='1.0' encoding='utf-8' ?>
<rdf:RDF
      xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
      xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
      xmlns:foaf="http://xmlns.com/foaf/0.1/"
       xml:base="http://ns.inria.fr/fabien.gandon">

       <foaf:PersonalProfileDocument rdf:about="">
         <foaf:maker rdf:resource="#me"/>
           <foaf:primaryTopic rdf:resource="#me"/>
       </foaf:PersonalProfileDocument>

       <foaf:Person rdf:ID="me">

         <foaf:name>Fabien Gandon</foaf:name>
           <foaf:title>Dr</foaf:title>
           <foaf:givenname>Fabien</foaf:givenname>
           <foaf:family_name>Gandon</foaf:family_name>
           <foaf:nick>Bafien</foaf:nick>

           <foaf:mbox rdf:resource="mailto:fabien.gandon@inria.fr"/>
           <foaf:homepage rdf:resource="http://fabien.info"/>
           <foaf:depiction rdf:resource="http://www-
sop.inria.fr/members/Fabien.Gandon/common/FabienGandonBackground.jpg"/>
           <foaf:phone rdf:resource="tel:0492387788"/>

           <foaf:workplaceHomepage rdf:resource="http://www.inria.fr/"/>
           <foaf:workInfoHomepage rdf:resource="http://fabien.info"/>
           <foaf:schoolHomepage rdf:resource="http://www.insa-rouen.fr"/>

           <foaf:knows>
             <foaf:Person>
               <foaf:name>Olivier Corby</foaf:name>
                 <foaf:mbox rdf:resource="mailto:olivier.corby@inria.fr"/>
                 <rdfs:seeAlso rdf:resource="http://www-
sop.inria.fr/members/Olivier.Corby/"/>
             </foaf:Person>
           </foaf:knows>
           <foaf:knows>
             <foaf:Person>
               <foaf:name>Catherine Faron-Zucker</foaf:name>
                 <foaf:mbox rdf:resource="mailto:faron@polytech.unice.fr"/>
                 <rdfs:seeAlso rdf:resource="http://www.i3s.unice.fr/~faron/"/>
             </foaf:Person>
           </foaf:knows>

        </foaf:Person>

</rdf:RDF>
```

2) **What is the syntax used?**

RDF XML

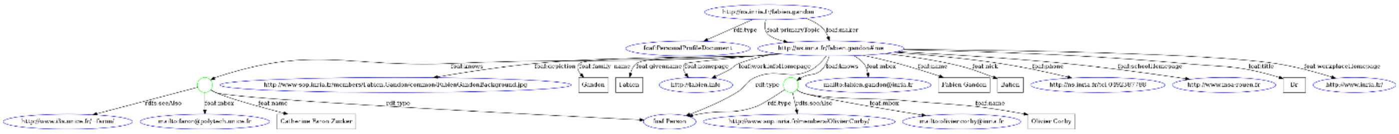
3) **Validate it and see the graph: http://www.w3.org/RDF/Validator/**


http://ns.inria.fr/fabien.gandon
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://xmlns.com/foaf/0.1/PersonalProfileDocument
http://xmlns.com/foaf/0.1/maker
http://xmlns.com/foaf/0.1/primaryTopic
http://ns.inria.fr/fabien.gandon#me
http://xmlns.com/foaf/0.1/knows
genid:A2313
http://xmlns.com/foaf/0.1/mbox
mailto:olivier.corby@inria.fr
http://www.w3.org/2000/01/rdf-schema#seeAlso
http://www-sop.inria.fr/members/Olivier.Corby/
http://xmlns.com/foaf/0.1/name
Olivier Corby
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://xmlns.com/foaf/0.1/Person
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://xmlns.com/foaf/0.1/knows
genid:A2315
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://xmlns.com/foaf/0.1/name
Catherine Faron-Zucker
http://xmlns.com/foaf/0.1/mbox
mailto:faron@polytech.unice.fr
http://www.w3.org/2000/01/rdf-schema#seeAlso
http://www.i3s.unice.fr/~faron/
http://xmlns.com/foaf/0.1/name
Fabien Gandon
http://xmlns.com/foaf/0.1/title
Dr
http://xmlns.com/foaf/0.1/givenname
Fabien
http://xmlns.com/foaf/0.1/family_name
Gandon
http://xmlns.com/foaf/0.1/nick
Bafien
http://xmlns.com/foaf/0.1/mbox
mailto:fabien.gandon@inria.fr
http://xmlns.com/foaf/0.1/homepage
http://fabien.info
http://xmlns.com/foaf/0.1/workInfoHomepage
http://xmlns.com/foaf/0.1/depiction
http://www-sop.inria.fr/members/Fabien.Gandon/common/FabienGandonBackground.jpg
http://xmlns.com/foaf/0.1/phone
tel:0492387788
http://xmlns.com/foaf/0.1/workplaceHomepage
http://www.inria.fr/
http://xmlns.com/foaf/0.1/schoolHomepage
http://www.insa-rouen.fr

4) **Translate into Turtle/N3: http://rdf-translator.appspot.com/**

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://ns.inria.fr/fabien.gandon> a foaf:PersonalProfileDocument ;
    foaf:maker <http://ns.inria.fr/fabien.gandon#me> ;
    foaf:primaryTopic <http://ns.inria.fr/fabien.gandon#me> .

<http://ns.inria.fr/fabien.gandon#me> a foaf:Person ;
    foaf:depiction <http://www-
sop.inria.fr/members/Fabien.Gandon/common/FabienGandonBackground.jpg> ;
    foaf:family_name "Gandon" ;
    foaf:givenname "Fabien" ;
    foaf:homepage <http://fabien.info> ;
    foaf:knows [ a foaf:Person ;
            rdfs:seeAlso <http://www.i3s.unice.fr/~faron/> ;
            foaf:mbox <mailto:faron@polytech.unice.fr> ;
            foaf:name "Catherine Faron-Zucker" ],
        [ a foaf:Person ;
            rdfs:seeAlso <http://www-sop.inria.fr/members/Olivier.Corby/> ;
            foaf:mbox <mailto:olivier.corby@inria.fr> ;
            foaf:name "Olivier Corby" ] ;
    foaf:mbox <mailto:fabien.gandon@inria.fr> ;
    foaf:name "Fabien Gandon" ;
    foaf:nick "Bafien" ;
    foaf:phone <http://ns.inria.fr/tel:0492387788> ;
    foaf:schoolHomepage <http://www.insa-rouen.fr> ;
    foaf:title "Dr" ;
    foaf:workInfoHomepage <http://fabien.info> ;
    foaf:workplaceHomepage <http://www.inria.fr/> .
```

5) **Visualize it also with: https://graves.cl/visualRDF/ http://www.easyrdf.org/converter (PNG, SVG)**



6) Adapt to your data and do it again

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://dsti.institute/alain.tholon> a foaf:PersonalProfileDocument ;
    foaf:maker <http://dsti.institute/alain.tholon> ;
    foaf:primaryTopic <http://dsti.institute/alain.tholon> .

<http://dsti.institute/alain.tholon#me> a foaf:Person ;
    foaf:family_name "Tholon" ;
    foaf:givenname "Alain" ;
    foaf:mbox <mailto:alain.tholon@edu.dsti.institute> ;
    foaf:name "Alain Tholon" ;
    foaf:knows [ a foaf:Person ;
            rdfs:seeAlso <http://ns.inria.fr/fabien.gandon#me/> ;
            foaf:mbox <mailto:fabien.gandon@inria.fr> ;
            foaf:name "Fabien Gandon" ].
```

Q2.5 what is the meaning of this RDF? What is this description saying?

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:exs="http://example.org/schema#">
 <rdf:Description rdf:about="http://example.org/doc.html">
  <rdf:type rdf:resource="http://example.org/schema#Report"/>
  <exs:theme rdf:resource="http://example.org#Music"/>
  <exs:theme rdf:resource="http://example.org#History"/>
  <exs:nbPages
rdf:datatype="http://www.w3.org/2001/XMLSchema#int">23</exs:nbPages>
 </rdf:Description>
</rdf:RDF>
```

Translated into Turtle for readability:

```
@prefix exs: <http://example.org/schema#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://example.org/doc.html> a exs:Report ;
    exs:nbPages "23"^^xsd:int ;
    exs:theme <http://example.org#History>,
        <http://example.org#Music> .
```

It is the description of a document named http://example.org/doc.html

The document should be a report, contains23 pages and be about history and music

Q2.6 Visit to Victor Hugo

1. See HTML data from: http://id.loc.gov/authorities/names/n79091479.html
2. Get RDF data from: http://id.loc.gov/authorities/names/n79091479.rdf
3. What is the syntax?
4. Translate into Turtle/N3: http://rdf-translator.appspot.com/
5. Any remark about the values of the properties of Victor Hugo?

1) See HTML data from: http://id.loc.gov/authorities/names/n79091479.html

The page contains information about Victor Hugo in many languages.

2) Get RDF data from: http://id.loc.gov/authorities/names/n79091479.rdf

```
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#">
  <madsrdf:PersonalName
rdf:about="http://id.loc.gov/authorities/names/n79091479"
xmlns:madsrdf="http://www.loc.gov/mads/rdf/v1#">
    <rdf:type rdf:resource="http://www.loc.gov/mads/rdf/v1#Authority"/>
    <madsrdf:authoritativeLabel xml:lang="en">Hugo, Victor, 1802-
1885</madsrdf:authoritativeLabel>
    <madsrdf:elementList rdf:parseType="Collection">
      <madsrdf:FullNameElement>
```

```
     <madsrdf:elementValue xml:lang="en">Hugo, Victor,</madsrdf:elementValue>
      </madsrdf:FullNameElement>
      <madsrdf:DateNameElement>
     <madsrdf:elementValue xml:lang="en">1802-1885</madsrdf:elementValue>
      </madsrdf:DateNameElement>
    </madsrdf:elementList>
    <madsrdf:hasVariant>
      <madsrdf:PersonalName>
     <rdf:type rdf:resource="http://www.loc.gov/mads/rdf/v1#Variant"/>
     <madsrdf:variantLabel xml:lang="en">Hiwkō, Vik‘t‘or, 1802-
1885</madsrdf:variantLabel>
..
```

3) What is the syntax?

RDF/XML

4) Translate into Turtle/N3: http://rdf-translator.appspot.com/

```
@prefix cs: <http://purl.org/vocab/changeset/schema#> .
@prefix identifiers: <http://id.loc.gov/vocabulary/identifiers/> .
@prefix madsrdf: <http://www.loc.gov/mads/rdf/v1#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix ri: <http://id.loc.gov/ontologies/RecordInfo#> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix skosxl: <http://www.w3.org/2008/05/skos-xl#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://id.loc.gov/authorities/subjects/sh86005256> a madsrdf:Authority,
        madsrdf:DeprecatedAuthority ;
    rdfs:label "sh86005256" .
..

<http://id.loc.gov/rwo/agents/n79091479> a
<http://id.loc.gov/ontologies/bibframe/Person>,
        madsrdf:RWO,
        <http://xmlns.com/foaf/0.1/Person> ;
    madsrdf:associatedLanguage <http://id.loc.gov/vocabulary/languages/fre> ;
    madsrdf:birthDate [ a skos:Concept ;
            rdfs:label "(edtf) 1802" ] ;
    madsrdf:deathDate [ a skos:Concept ;
            rdfs:label "(edtf) 1885" ] ;
    madsrdf:occupation [ a madsrdf:Occupation ;
            rdfs:label "Novelist" ] .
```

5) Any remark about the values of the properties of Victor Hugo?

There is a mistake in the file, the language is set to English for all translated string.

Q2.7 What is the syntax of the following RDF statement? What does it mean?

```
@prefix dcterms: <http://purl.org/dc/terms/>.
GRAPH  <http://inria.fr/topics/algebra>
{
  <http://inria.fr/rr/doc.html>
    dcterms:subject
      <http://data.bnf.fr/ark:/12148/cb121105993> .
}
```

The syntax is Trig.

A new graph <http://inria.fr/topics/algebra> is created that described the triple about <http://inria.fr/rr/doc.html>

Q2.8 Visit Leukocyte surface antigen CD53

1. See HTML data from: http://www.uniprot.org/uniprot/Q61451
2. Get RDF data from: http://www.uniprot.org/uniprot/Q61451.rdf
3. What is the syntax?
4. Translate into Turtle/N3: http://rdf-translator.appspot.com/
5. Any remark about the structure of the data?

1) See HTML data from: http://www.uniprot.org/uniprot/Q61451

Biological data about a mouse

2) Get RDF data from: http://www.uniprot.org/uniprot/Q61451.rdf

```
<?xml version='1.0' encoding='UTF-8'?>
<rdf:RDF xml:base="http://purl.uniprot.org/uniprot/"
xmlns="http://purl.uniprot.org/core/" xmlns:dcterms="http://purl.org/dc/terms/"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:skos="http://www.w3.org/2004/02/skos/core#"
xmlns:bibo="http://purl.org/ontology/bibo/"
xmlns:foaf="http://xmlns.com/foaf/0.1/" xmlns:void="http://rdfs.org/ns/void#"
xmlns:sd="http://www.w3.org/ns/sparql-service-description#"
xmlns:faldo="http://biohackathon.org/resource/faldo#">
<owl:Ontology rdf:about="http://purl.uniprot.org/uniprot/">
<owl:imports rdf:resource="http://purl.uniprot.org/core/"/>
</owl:Ontology>
<rdf:Description rdf:about="http://purl.uniprot.org/uniprot/Q61451">
<rdf:type rdf:resource="http://purl.uniprot.org/core/Protein"/>
<reviewed
rdf:datatype="http://www.w3.org/2001/XMLSchema#boolean">true</reviewed>
<created rdf:datatype="http://www.w3.org/2001/XMLSchema#date">1997-11-
01</created>
<modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2017-11-
22</modified>
<version rdf:datatype="http://www.w3.org/2001/XMLSchema#int">126</version>
<mnemonic>CD53_MOUSE</mnemonic>
```

3) What is the syntax?

RDF / XML

4) **Translate into Turtle/N3: http://rdf-translator.appspot.com/**

```
@prefix : <http://purl.uniprot.org/core/> .
@prefix bibo: <http://purl.org/ontology/bibo/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix faldo: <http://biohackathon.org/resource/faldo#> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix sd: <http://www.w3.org/ns/sparql-service-description#> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .
@prefix void: <http://rdfs.org/ns/void#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://purl.uniprot.org/uniprot/> a owl:Ontology ;
    owl:imports : .

<http://purl.uniprot.org/uniprot/#_60D30EDFA19FA8F3_rdfs.comment_A8EE4971C012C3
8A> a rdf:Statement ;
    :attribution <http://purl.uniprot.org/uniprot/Q61451#attribution-
1F1D235CEE6E9F517881B603655DF439> ;
    rdf:object "Belongs to the tetraspanin (TM4SF) family." ;
    rdf:predicate rdfs:comment ;
    rdf:subject <http://purl.uniprot.org/uniprot/Q61451#SIP8205AE30B0C44D1E> .
```

5) **Any remark about the structure of the data?**

rdf:Statement appears everywhere in Turtle/N3

Day 02: Answers to the practical session on RDF.

Software requirements

- A real text editor (e.g. Notepad++, Gedit, Sublime Text, Emacs, etc.)
- The RDF XML online validation service by W3C: https://www.w3.org/RDF/Validator/
- The RDF online translator: http://rdf-translator.appspot.com/
- The SPARQL Corese engine: http://wimmics.inria.fr/corese

Create RDF

Read carefully the following statements:

> *"Margot is a 32 year old woman, she has a shoe size of 38 and trouser size of 40. She is married to Arthur, with whom she has two children: Simon and Marie. Margot is a teacher and Alice and Pierre are her colleagues. Margot 's mother is called Simone."*

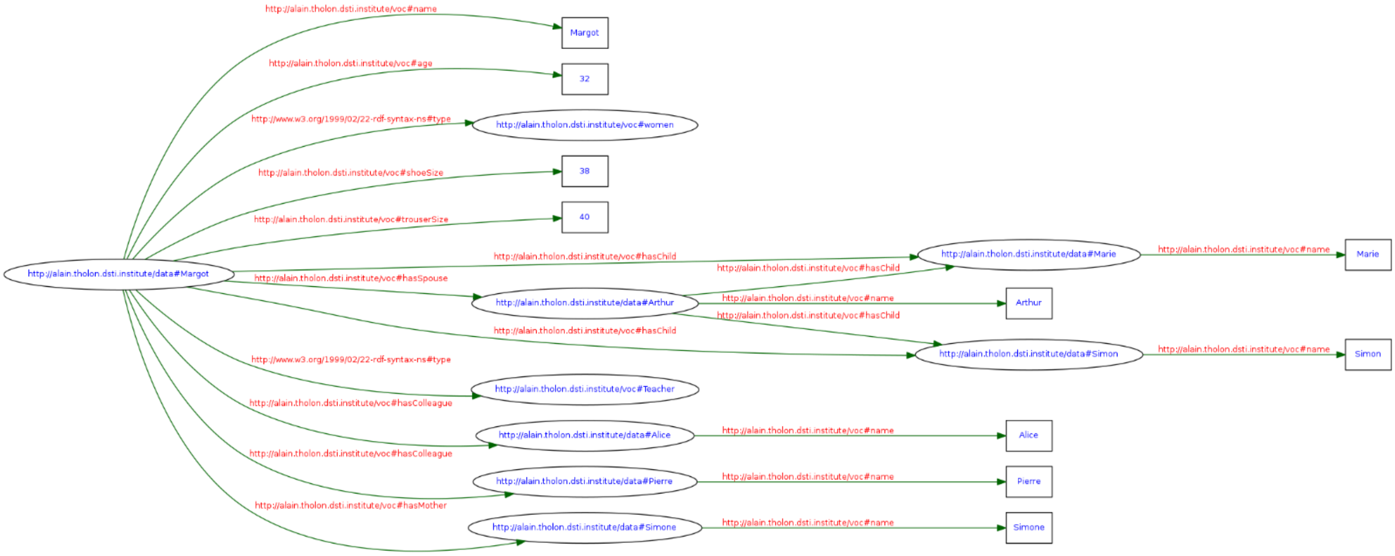
1. Use your text editor and write the above statements in RDF in N3 syntax inventing your own vocabulary. Save you file as "Margot.ttl"
2. Use your favorite text or XML editor and write the above statements in RDF in XML syntax reusing the same vocabulary "Margot.rdf"
3. Use the RDF XML online validation service to validate your XML and see the triples https://www.w3.org/RDF/Validator/
4. In the validator use the option to visualize the graph
5. Use the RDF online translator to validate your N3 and translate it into RDF/XML: http://rdf-translator.appspot.com/
6. Compare your RDF/XML with the result of the N3 translation
7. Translate in other formats to see the results.

Code of validated RDF in N3 syntax:

```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix voc: <http://alain.tholon.dsti.institute/voc#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
<http://alain.tholon.dsti.institute/data#Margot> a voc:Teacher , voc:women ;
    voc:age "32"^^xsd:integer ;
    voc:shoeSize "38"^^xsd:integer ;
    voc:trouserSize "40"^^xsd:integer ;
    voc:hasChild <http://alain.tholon.dsti.institute/data#Marie>,
<http://alain.tholon.dsti.institute/data#Simon>;
    voc:hasSpouse <http://alain.tholon.dsti.institute/data#Arthur> ;
     voc:hasMother <http://alain.tholon.dsti.institute/data#Simone> ;
     voc:hasColleague <http://alain.tholon.dsti.institute/data#Alice> ;
     voc:hasColleague <http://alain.tholon.dsti.institute/data#Pierre> ;
    voc:name "Margot" .
<http://alain.tholon.dsti.institute/data#Arthur>
    voc:hasChild <http://alain.tholon.dsti.institute/data#Marie>,
        <http://alain.tholon.dsti.institute/data#Simon> ;
    voc:name "Arthur" .
<http://alain.tholon.dsti.institute/data#Marie> voc:name "Marie" .
<http://alain.tholon.dsti.institute/data#Simon> voc:name "Simon" .
<http://alain.tholon.dsti.institute/data#Simone> voc:name "Simone" .
<http://alain.tholon.dsti.institute/data#Alice> voc:name "Alice" .
<http://alain.tholon.dsti.institute/data#Pierre> voc:name "Pierre" .
```

Code of validated RDF in XML syntax:

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [   <!ENTITY vocabulaire
"http://alain.tholon.dsti.institute/voc">     <!ENTITY xsd
"http://www.w3.org/2001/XMLSchema#"> ]>
<rdf:RDF
   xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:voc="&vocabulaire;#"
   xml:base="http://alain.tholon.dsti.institute/data"
>
  <rdf:Description rdf:about="#Margot">
    <voc:name>Margot</voc:name>
    <voc:age rdf:datatype="xsd:integer">32</voc:age>
    <rdf:type rdf:resource="&vocabulaire;#women"/>
    <voc:shoeSize rdf:datatype="xsd:integer">38</voc:shoeSize>
    <voc:trouserSize rdf:datatype="xsd:integer">40</voc:trouserSize>
    <voc:hasSpouse>
       <rdf:Description rdf:about="#Arthur">
          <voc:name>Arthur</voc:name>
          <voc:hasChild rdf:resource="#Marie"/>
          <voc:hasChild rdf:resource="#Simon"/>
       </rdf:Description>
    </voc:hasSpouse>
    <voc:hasChild>
       <rdf:Description rdf:about="#Simon">
          <voc:name>Simon</voc:name>
       </rdf:Description>
    </voc:hasChild>
    <voc:hasChild>
       <rdf:Description rdf:about="#Marie">
          <voc:name>Marie</voc:name>
       </rdf:Description>
    </voc:hasChild>
    <rdf:type rdf:resource="&vocabulaire;#Teacher"/>
    <voc:hasColleague>
       <rdf:Description rdf:about="#Alice">
          <voc:name>Alice</voc:name>
       </rdf:Description>
    </voc:hasColleague>
    <voc:hasColleague>
       <rdf:Description rdf:about="#Pierre">
          <voc:name>Pierre</voc:name>
       </rdf:Description>
    </voc:hasColleague>
    <voc:hasMother>
       <rdf:Description rdf:about="#Simone">
          <voc:name>Simone</voc:name>
       </rdf:Description>
    </voc:hasMother>
  </rdf:Description>
</rdf:RDF>
```


http://alain.tholon.dsti.institute/data#Margot
http://alain.tholon.dsti.institute/voc#name
Margot
http://alain.tholon.dsti.institute/voc#age
32
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://alain.tholon.dsti.institute/voc#women
http://alain.tholon.dsti.institute/voc#shoeSize
38
http://alain.tholon.dsti.institute/voc#trouserSize
40
http://alain.tholon.dsti.institute/voc#hasChild
http://alain.tholon.dsti.institute/data#Marie
http://alain.tholon.dsti.institute/voc#name
Marie
http://alain.tholon.dsti.institute/voc#hasSpouse
http://alain.tholon.dsti.institute/data#Arthur
http://alain.tholon.dsti.institute/voc#hasChild
http://alain.tholon.dsti.institute/voc#name
Arthur
http://alain.tholon.dsti.institute/voc#hasChild
http://alain.tholon.dsti.institute/voc#hasChild
http://alain.tholon.dsti.institute/data#Simon
http://alain.tholon.dsti.institute/voc#name
Simon
http://www.w3.org/1999/02/22-rdf-syntax-ns#type
http://alain.tholon.dsti.institute/voc#Teacher
http://alain.tholon.dsti.institute/voc#hasColleague
http://alain.tholon.dsti.institute/data#Alice
http://alain.tholon.dsti.institute/voc#name
Alice
http://alain.tholon.dsti.institute/voc#hasColleague
http://alain.tholon.dsti.institute/data#Pierre
http://alain.tholon.dsti.institute/voc#name
Pierre
http://alain.tholon.dsti.institute/voc#hasMother
http://alain.tholon.dsti.institute/data#Simone
http://alain.tholon.dsti.institute/voc#name
Simone

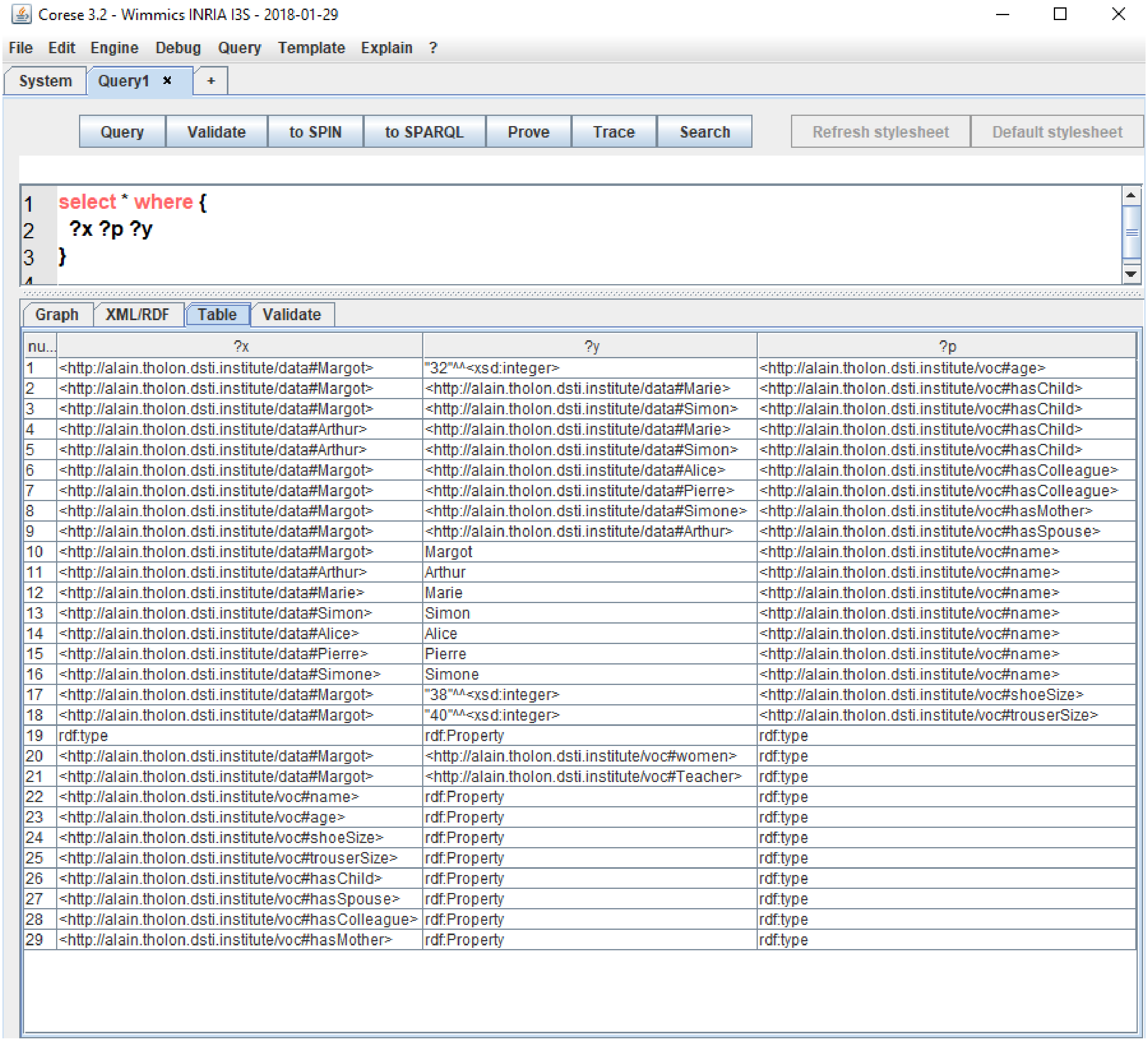
Query your data

Download the Corese.jar library and start it as a standalone application: On Window double-click the file “.jar”. If it does not work or on other platforms, run the command "java - jar" followed by the name of the “.jar” archive. Notice that you need java on your machine and proper path configuration.

This interface provides two tabs: (1) one to load input files and see traces of execution, and (2) the default tab to start loading or writing queries and see their result. Load the annotations contained in the file “Margot.rdf” you created and validated before. The interface contains a default SPARQL query:

```
Select ?x ?t where { ?x rdf:type ?t}
```

The SPARQL language will be presented in the next course. Just know that this query can find all of the resources referred to in the data you loaded and their types. Launch the query and check the results.



```
select * where {
 ?x ?p ?y
}
```

nu...	?x	?y	?p
1	<http://alain.tholon.dsti.institute/data#Margot>	"32"^^<xsd:integer>	<http://alain.tholon.dsti.institute/voc#age>
2	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Marie>	<http://alain.tholon.dsti.institute/voc#hasChild>
3	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Simon>	<http://alain.tholon.dsti.institute/voc#hasChild>
4	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/data#Marie>	<http://alain.tholon.dsti.institute/voc#hasChild>
5	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/data#Simon>	<http://alain.tholon.dsti.institute/voc#hasChild>
6	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Alice>	<http://alain.tholon.dsti.institute/voc#hasColleague>
7	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Pierre>	<http://alain.tholon.dsti.institute/voc#hasColleague>
8	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Simone>	<http://alain.tholon.dsti.institute/voc#hasMother>
9	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/voc#hasSpouse>
10	<http://alain.tholon.dsti.institute/data#Margot>	Margot	<http://alain.tholon.dsti.institute/voc#name>
11	<http://alain.tholon.dsti.institute/data#Arthur>	Arthur	<http://alain.tholon.dsti.institute/voc#name>
12	<http://alain.tholon.dsti.institute/data#Marie>	Marie	<http://alain.tholon.dsti.institute/voc#name>
13	<http://alain.tholon.dsti.institute/data#Simon>	Simon	<http://alain.tholon.dsti.institute/voc#name>
14	<http://alain.tholon.dsti.institute/data#Alice>	Alice	<http://alain.tholon.dsti.institute/voc#name>
15	<http://alain.tholon.dsti.institute/data#Pierre>	Pierre	<http://alain.tholon.dsti.institute/voc#name>
16	<http://alain.tholon.dsti.institute/data#Simone>	Simone	<http://alain.tholon.dsti.institute/voc#name>
17	<http://alain.tholon.dsti.institute/data#Margot>	"38"^^<xsd:integer>	<http://alain.tholon.dsti.institute/voc#shoeSize>
18	<http://alain.tholon.dsti.institute/data#Margot>	"40"^^<xsd:integer>	<http://alain.tholon.dsti.institute/voc#trouserSize>
19	rdf:type	rdf:Property	rdf:type
20	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#women>	rdf:type
21	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#Teacher>	rdf:type
22	<http://alain.tholon.dsti.institute/voc#name>	rdf:Property	rdf:type
23	<http://alain.tholon.dsti.institute/voc#age>	rdf:Property	rdf:type
24	<http://alain.tholon.dsti.institute/voc#shoeSize>	rdf:Property	rdf:type
25	<http://alain.tholon.dsti.institute/voc#trouserSize>	rdf:Property	rdf:type
26	<http://alain.tholon.dsti.institute/voc#hasChild>	rdf:Property	rdf:type
27	<http://alain.tholon.dsti.institute/voc#hasSpouse>	rdf:Property	rdf:type
28	<http://alain.tholon.dsti.institute/voc#hasColleague>	rdf:Property	rdf:type
29	<http://alain.tholon.dsti.institute/voc#hasMother>	rdf:Property	rdf:type

Understand existing data

1, Get the RDF/XML about http://ns.inria.fr/fabien.gandon#me and translate the RDF/XML into Turtle/N3

Code of validated RDF in N3 syntax:

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://ns.inria.fr/fabien.gandon> a foaf:PersonalProfileDocument ;
    foaf:maker <http://ns.inria.fr/fabien.gandon#me> ;
    foaf:primaryTopic <http://ns.inria.fr/fabien.gandon#me> .

<http://ns.inria.fr/fabien.gandon#me> a foaf:Person ;
    foaf:depiction <http://www-
sop.inria.fr/members/Fabien.Gandon/common/FabienGandonBackground.jpg> ;
    foaf:family_name "Gandon" ;
    foaf:givenname "Fabien" ;
    foaf:homepage <http://fabien.info> ;
    foaf:knows [ a foaf:Person ;
            rdfs:seeAlso <http://www.i3s.unice.fr/~faron/> ;
            foaf:mbox <mailto:faron@polytech.unice.fr> ;
            foaf:name "Catherine Faron-Zucker" ],
        [ a foaf:Person ;
            rdfs:seeAlso <http://www-sop.inria.fr/members/Olivier.Corby/> ;
            foaf:mbox <mailto:olivier.corby@inria.fr> ;
            foaf:name "Olivier Corby" ] ;
    foaf:mbox <mailto:fabien.gandon@inria.fr> ;
    foaf:name "Fabien Gandon" ;
    foaf:nick "Bafien" ;
    foaf:phone <http://ns.inria.fr/tel:0492387788> ;
    foaf:schoolHomepage <http://www.insa-rouen.fr> ;
    foaf:title "Dr" ;
    foaf:workInfoHomepage <http://fabien.info> ;
    foaf:workplaceHomepage <http://www.inria.fr/> .
```

Can you guess the link between http://ns.inria.fr/fabien.gandon and http://ns.inria.fr/fabien.gandon#me

http://ns.inria.fr/fabien.gandon is the URL of the profile that describes the person whose URL is http://ns.inria.fr/fabien.gandon#me (that happens to be also the author of the profile).

2, Using CURL get the RDF/XML data of the White Shark on the BBC web site. Try to validate it on the W3C validation service.

Do you get an error? Why? How can you fix it?

```
curl -L -H "Accept: application/rdf+xml"
http://www.bbc.co.uk/nature/life/Great_white_shark -o Great_white_shark.rdf
```

Yes https://www.w3.org/RDF/Validator/ gives an at line25 because HTML tags are used inside description.

It can be fixed by adding declaring the value of the description as litteral as follows:

```
 <dc:description rdf:parseType='Literal'>
```

3, Get the Turtle data of Paris on DBpedia.org then in the file find the triple that declares it as a capital in Europe.

The triple is:

```
curl -L -H "Accept: text/turtle" http://dbpedia.org/resource/Paris -o Paris.ttl
```

```
dbr:Paris  dct:subject     dbc:Cities_in_France ,
           dbc:European_culture ,
           dbc:French_culture ,
           dbc:Cities_in_Île-de-France ,
           dbc:Departments_of_Île-de-France ,
           dbc:Prefectures_in_France ,
           dbc:Paris ,
           dbc:Populated_places_established_in_the_3rd_century_BC ,
           dbc:Capitals_in_Europe ,
           dbc:Catholic_pilgrimage_sites ,
           dbc:Companions_of_the_Liberation .
```

4, At the following address you will find an RDF file containing several annotations:

http://wimmics.inria.fr/doc/tutorial/human_2013.rdf

Download the file and use the RDF XML online validation service to validate the XML and see the triples and the graph.

1. What is the namespace used for instances / resources created in this file?

   The namespace used for instances / resources created in this file is http://www.inria.fr/2007/09/11/humans.rdfs-instances

2. By which mechanism is the association between instances and namespace done i.e. how was the instance namespace specified?

```
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [
   <!ENTITY humans   "http://www.inria.fr/2007/09/11/humans.rdfs">
   <!ENTITY xsd      "http://www.w3.org/2001/XMLSchema#"> ]>

<rdf:RDF
   xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
   xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
   xmlns:xsd ="&xsd;"
   xmlns     ="&humans;#"
   xml:base  ="&humans;-instances" >
```

All instances in the document are defined using relative URL which are automatically prefixed by xml:base. xml:base resolves to http://www.inria.fr/2007/09/11/humans.rdfs-instances

3. What is the namespace of the RDF schema used and how is it associated with the tags?

The namespace of the RDF schema used is http://www.inria.fr/2007/09/11/humans.rdfs.

It is associated with the tags through the document XML default namespace defined by xmlns="…"

4. Explain the code `xmlns="&humans;#'`

It defines the default namespace for the XML tags of the document that do not have an explicit namespace.

5. Find everything about information on John in this file.
   all the information:

John is the child of Harry
John is a person, has a shoe size of 14 an age of 37.
John is the father of Mark
John is the friend of Alice
John is the spouse of Jennifer
John has a shirt size of 12, a trouser size of 44 and for parent Sophie.

6. Translate the file in turtle and save it as human_2013.ttl
   10 first lines:

```
@prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve> a :Lecturer,
     :Person ;
  :hasFriend <http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice> ;
  :hasSpouse <http://www.inria.fr/2007/09/11/humans.rdfs-instances#David> ;
  :name "Eve" .
```

7. In the turtle version find everything about Laura.
   all the information:

Laura is a person, a lecturer and a researcher.

Alice is the friend of Laura.

Laura is the mother of Catherine

Laura is the spouse of William

Day 03: questions from the course on SPARQL.

Q3.1 Test SPARQL online

Connect to: https://corese.inria.fr/srv/tutorial/sparql

Answers to the query:

```
prefix v: <http://www.inria.fr/2015/humans#>
select * where {  ?x a v:Person . }
```

```
     x
1    <http://www.inria.fr/2015/humans-instances#John>
2    <http://www.inria.fr/2015/humans-instances#Sophie>
3    <http://www.inria.fr/2015/humans-instances#Mark>
4    <http://www.inria.fr/2015/humans-instances#Eve>
5    <http://www.inria.fr/2015/humans-instances#David>
6    <http://www.inria.fr/2015/humans-instances#Laura>
7    <http://www.inria.fr/2015/humans-instances#William>
8    <http://www.inria.fr/2015/humans-instances#Karl>
```

Q3.2 Test SPARQL online

Connect to

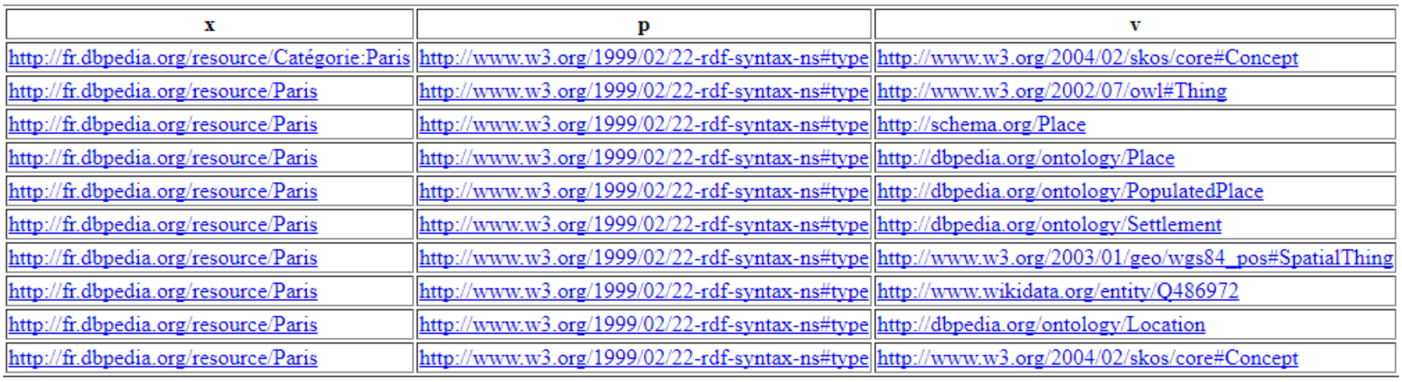
http://dbpedia.org/snorql/ or

http://fr.dbpedia.org/sparql or …

http://wiki.dbpedia.org/Internationalization/Chapters

Answers to the query:

```
SELECT * WHERE {
  ?x rdfs:label "Paris"@fr .
  ?x ?p ?v .
}
LIMIT 10
```



x	p	v
http://fr.dbpedia.org/resource/Catégorie:Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2004/02/skos/core#Concept
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2002/07/owl#Thing
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://schema.org/Place
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://dbpedia.org/ontology/Place
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://dbpedia.org/ontology/PopulatedPlace
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://dbpedia.org/ontology/Settlement
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2003/01/geo/wgs84_pos#SpatialThing
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.wikidata.org/entity/Q486972
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://dbpedia.org/ontology/Location
http://fr.dbpedia.org/resource/Paris	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/2004/02/skos/core#Concept

Q3.3 Test SPARQL online

Connect to:

https://query.wikidata.org/

What does this query retrieve?

```
SELECT distinct ?p ?n WHERE
{      wd:Q30 p:P6 [ ps:P6 ?p ].
       ?p rdfs:label ?n .
       FILTER (lang(?n)="en") }
```

Discover wd:Q30 using the namespace attached to wd:

PREFIX wd: <http://www.wikidata.org/entity/>

Discover p:P6 using the namespace attached to p:

PREFIX p: <http://www.wikidata.org/prop/>

Find q-name of the property “given name”

https://www.wikidata.org/wiki/Wikidata:List_of_properties

The query retrieves the list of the president of the US.

https://www.wikidata.org/wiki/Q30 : “United States of America”

https://www.wikidata.org/wiki/Property:P6 “head of government”

https://www.wikidata.org/wiki/Wikidata:List_of_properties: when searching for “given name”, one of the answer is :

> https://www.wikidata.org/wiki/Property:P735 “given name” : first name or another given name of this person; values used with the property shouldn't link disambiguations nor family names

Q3.4 SPARQL query to return 20 persons at most (use type foaf:Person)

```
SELECT ?x WHERE { ?x rdf:type foaf:Person }
LIMIT 20
```

Q3.5 SPARQL query to return 20 persons (at most), after the 10th result i.e. from 11th to 30th

```
SELECT ?x WHERE { ?x a foaf:Person }
OFFSET 10
LIMIT 20
```

Q3.6 You have two properties: c:name and c:age

1.Find the age of resources whose name is ‘Fabien’

2.Find the name of resources whose age is less than 50

3.Find property values of resources whose name is ‘Fabien’ and whose age is less than 50

4.Find other names of resources whose name is ‘Fabien’

5.Find resources which have two different properties with the same value

6.Find resources which have the same property with two different values

1)
```
SELECT ?a WHERE { ?x c:name “Fabien” ; c:age ?a }
```

2)
```
SELECT ?n WHERE { ?x c:name ?n ; c:age ?a . FILTER(?a < 50) }
```

3)
```
SELECT ?p ?v WHERE { ?x c:name “Fabien” ; c:age ?a ; ?p ?v . FILTER(?a < 50) }
```

4)
```
SELECT ?n WHERE { ?x c:name “Fabien”, ?n . FILTER(?n != “Fabien”) }
```

5)
```
SELECT ?x WHERE { ?x ?p1 ?v ; ?p2 ?v . FILTER(?p1 != ?p2) }
```

6)
```
SELECT ?x WHERE { ?x ?p ?v1, ?v2 . FILTER(?v1 != ?v2) }
```

Q3.7 Could this query return `ex:a c:memberOf ex:b` and why ?

```
select * where {
  ?x c:memberOf ?org .
  minus { ex:a c:memberOf ex:b }
}
```

Yes because there is no variable in the “minus” (therefore minus cannot be applied to the first part of the request as there is no common variable between them).

Q3.8 get the members of organizations (c:memberOf) but remove the resources author of a document (c:author) by using ‘not exists’

```
SELECT ?x WHERE { ?x c:members ?org . FILTER(! exists {?x c:author ?doc} ) }
```

Q3.9 what is retrieving this query ?

```
prefix ex: <http://example.org/>
select ?x (count(?doc) as ?c)
where {  ?x ex:author ?doc  }
group by ?x
order by desc(count(?doc))
```

This query retrieves all the authors of documents, sorting them in descending order according to their number of publications.

Q3.10 What expression should we use to find the ?x related to ?y by paths composed of properties foaf:knows and/or rdfs: seeAlso?

- ?x (foaf:knows | rdfs:seeAlso)+ ?y
- ?x foaf:knows+ | rdfs:seeAlso+ ?y
- ?x (foaf:knows / rdfs:seeAlso)+ ?y

```
?x (foaf:knows | rdfs:seeAlso)+ ?y
```

Q3.11 what is this query retrieving?

```
prefix foaf: <http://xmlns.com/foaf/0.1/>
select ?x (if (bound(?n), ?n, "John Doe") as ?m)
where {
   ?x foaf:knows ?y
   optional { ?y foaf:name ?n }
}
```

This query retrieves the name of the friends of the resources, replacing those that haven't got a name by "John Doe"

Q3.12 what is this query retrieving?

```
prefix ex:  <http://example.org/>
select ?x (avg(?a) as ?b)
where {
  ?x ex:knows ?y .
  ?y ex:age ?a
}
group by ?x
```

This query returns all resources, providing the average age of their friends.

Q3.13 You have two properties: c:name and c:study and the resources c:Informatics and c:Mathematics

1. Find resources that study informatics or mathematics
2. In addition return the name of the resource if it has a name
3. In addition return the graph where the name is given

1)

```
SELECT ?x WHERE {{?x c:study c:Informatics} UNION {?x c:study c:Mathematics}}
```

2)

```
SELECT * WHERE { {?x c:study c:Informatics} UNION {?x c:study c:Mathematics}
OPTIONAL {?x c:name ?n} }
```

3)

```
SELECT * WHERE { {?x c:study c:Informatics} UNION {?x c:study c:Mathematics}
OPTIONAL {graph ?g {?x c:name ?n}} }
```

Q3.14 On which graph(s) is calculated ?x ?p ?y

On which graph(s) is calculated graph ?g { ?y ?q ?z }

```
prefix ex:  <http://example.org/>
select *
from ex:g1
from named ex:g2
where {
   ?x ?p ?y .
   graph ?g { ?y ?q ?z}  }
```

?x ?p ?y is calculated on unnamed graph, i.e. ex:g1

?y ?q ?z is calculated on named graph, i.e. ex:g2

Q3.15 Write a query to change foaf:name into rdfs:label

```
DELETE { ?x foaf:name ?n }
INSERT { ?x rdfs:label ?n }
WHERE  { ?x foaf:name ?n }
```

Q3.16 what is this query performing?

```
prefix ex: <http://example.org/>
delete { ?x ex:age ?a }
insert { ?x ex:age ?i }
where  {
  select ?x (xsd:integer(?a) as ?i)
  where {
    ?x ex:age ?a
    filter(datatype(?a) = xsd:string)
  }
}
```

This query transforms the value of the age property from string to integer, for all resources where the value of the age property is a string.

Day 03: Answers to the practical session on SPARQL.

Software requirements

- The RDF XML online validation service by W3C: https://www.w3.org/RDF/Validator/
- The RDF online translator: http://rdf-translator.appspot.com/
- The SPARQL Corese engine: http://wimmics.inria.fr/corese

Basic query on RDF human.rdf

If you haven't done it yet download the SPARQL Corese engine.

On Window double-click the file ".jar". If it does not work or on other platforms, run the command "java - jar" followed by the name of the ".jar" archive. Notice that you need java on your machine and proper path configuration

This interface provides two tabs: (1) one to load input files and see traces of execution, and (2) the default tab to start loading or writing queries and see their result.

If you haven't done it yet download the following file of annotations and save it as "human.rdf":

http://wimmics.inria.fr/doc/tutorial/human_2013.rdf

Load the file human.rdf as RDF data in corese.

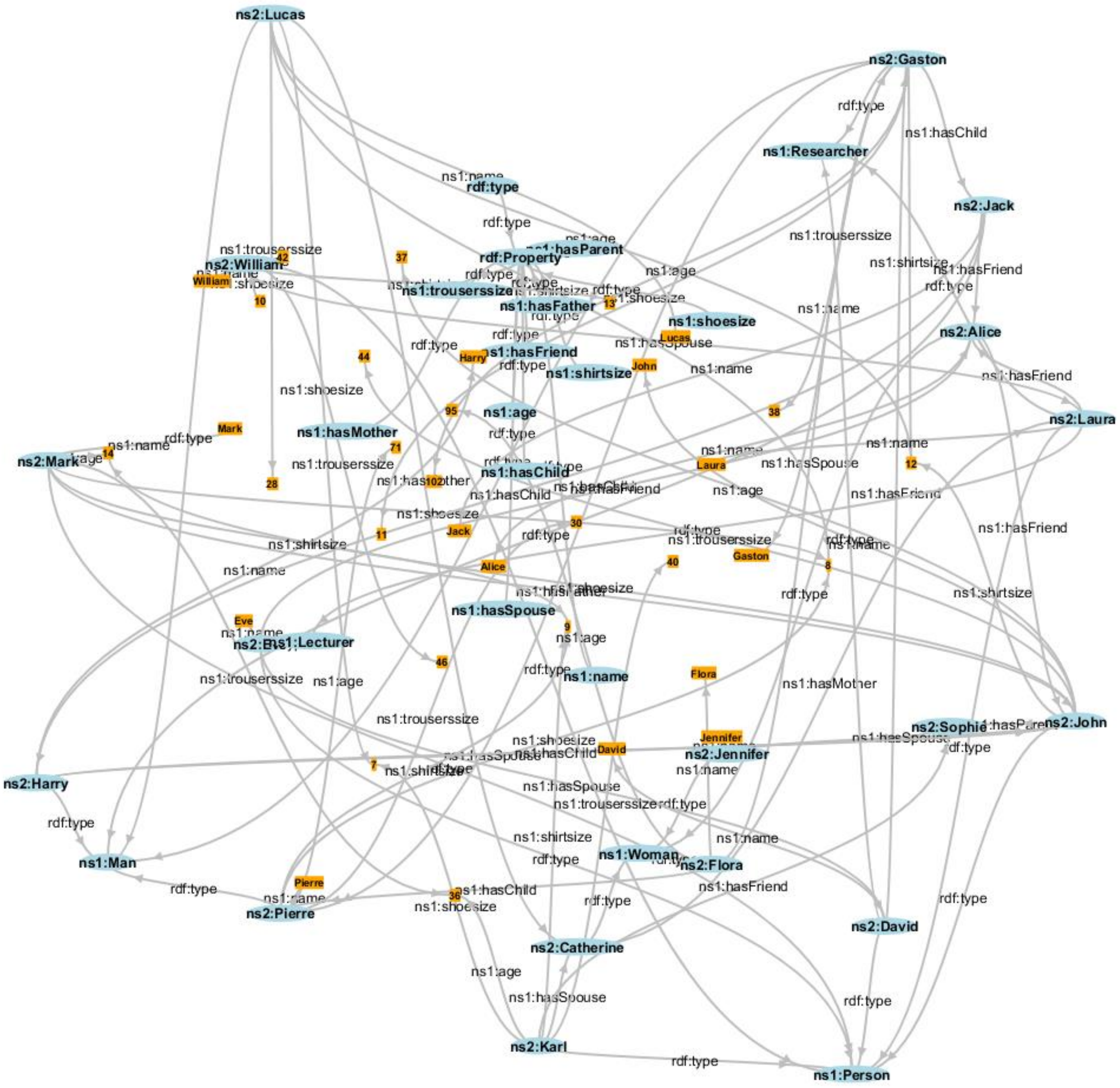
Question 1:

Create a new tab to enter the following query and explain what it does and the results you get:

```
CONSTRUCT { ?s ?p ?o } WHERE { ?s ?p ?o }
```

This is a good way to familiarize yourself with the data.

Screenshot:



Question 2:

Create a new tab to enter the following query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>

select * where { ?x a ?t . filter(strstarts(?t, h:)) }
```

Translate this query in one sentence. Run this query. How many answers do you get?

Find John and his types.

This query retrieves all resources that have a property that start with “http://www.inria.fr/2007/09/11/humans.rdfs”

http://www.inria.fr/2007/09/11/humans.rdfs-instances#John is of type http://www.inria.fr/2007/09/11/humans.rdfs#Person

Question 3:

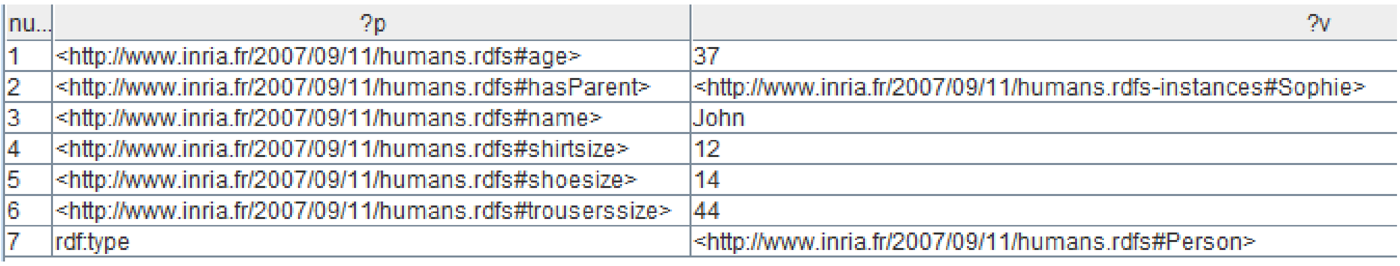
In the previous answer, locate the URI of John.

1. formulate a SELECT query to find all the properties of John
   Query

Assuming that properties and values are expected:

```
select * where { <http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
?p ?v }
```

Results:



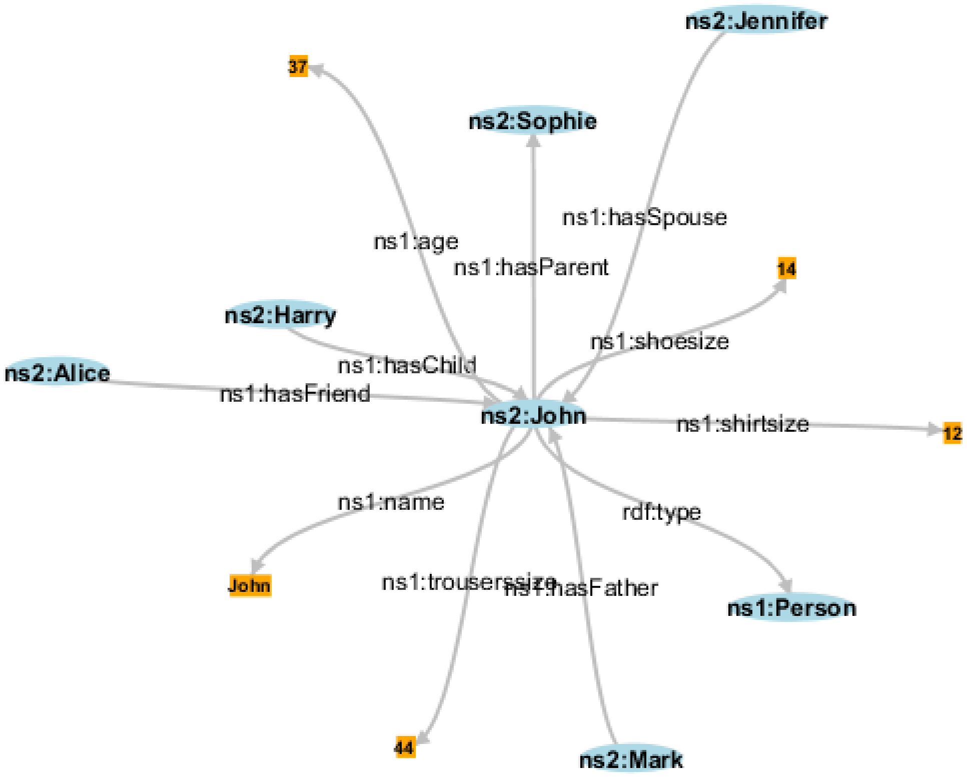
nu...	?p	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs#age>	37
2	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
3	<http://www.inria.fr/2007/09/11/humans.rdfs#name>	John
4	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>	12
5	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	14
6	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>	44
7	rdf:type	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>

2. request a description of John using the SPARQL clause for this.

Query

```
DESCRIBE <http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
```

Results:



Question 4

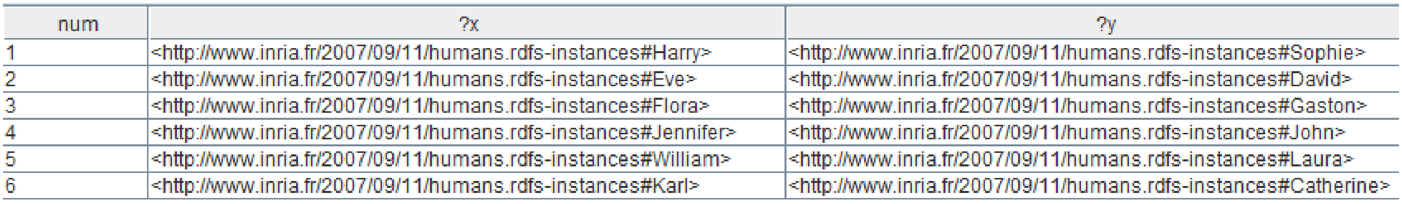
Create a new tab to enter the following query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>

select * where { ?x h:hasSpouse ?y }
```

Translate this query in one sentence.

This query retrieves all the resources that has a spouse, along with their spouse

Run this query. How many answers do you get?



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>

6 answers are returned.

Question 5:

In the RDF file, find the name of the property that is used to give the shoe size of a person.

1. Deduce a query to extract all the persons (h:Person) with their shoe size.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a h:Person ; h:shoesize ?y }
```

2. Change this query to retrieve all persons and, if available, their shoe size.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a h:Person . OPTIONAL { ?x h:shoesize ?y } }
```

3. Change this query to retrieve all persons whose shoesize is greater than 8 or whose shirt size is greater than 12.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a h:Person ; h:shoesize ?y ; h:shirtsize ?z
FILTER((?y >8) || (?z >12))}
```

Question 6:

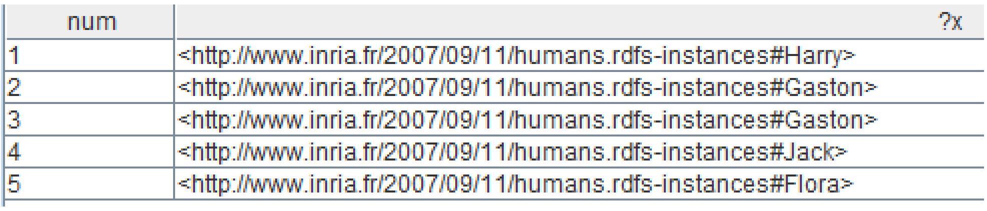
In the RDF file, find the name of the property that is used to indicate the children of a person.

1. Formulate a query to find people who have at least one child.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?x where { ?x h:hasChild ?y }
```

How many answers do you get? How many duplicates do you identify in these responses?



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>

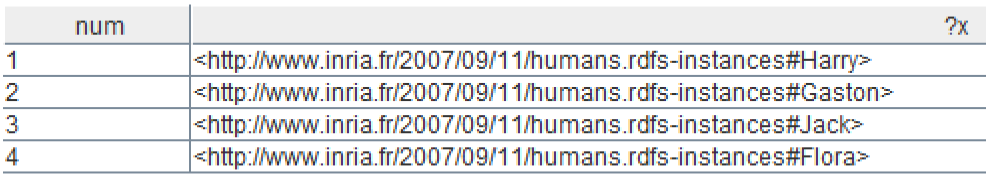
5 answers are returned, with one duplicate (#Gaston)

2. Find a way to avoid duplicates.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select distinct ?x where { ?x h:hasChild ?y }
```

How many answers do you get then?



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>

4 answers are returned, the duplicate one has been removed.

3. Rewrite a query to find the Persons who have no child.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?x where { { ?x a h:Person } minus { ?x h:hasChild ?y } }
```

Question 7

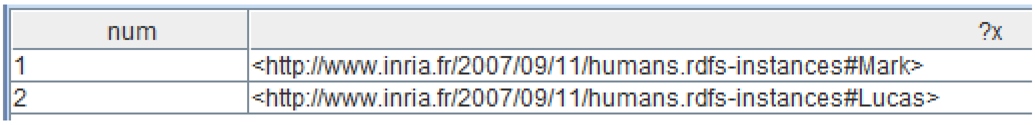
In the RDF file, find the name of the property that is used to give the age of a person.

1. Formulate a query to find people who are not adults.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?x where { ?x h:age ?a FILTER (?a < 18) }
```

How many answers do you get?



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>

2 answers are returned

2. Use the appropriate query clause to check if Mark is an adult; use the proper clause statement for this type of query to get a true or false answer.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>
ASK WHERE { b:Mark h:age ?a FILTER (?a < 18) }
```

3. Write a query that indicates for each person if her age is even (true or false).

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
SELECT ?x ( if (floor(?a/2) = ?a/2, true, false) as ?p)
WHERE { ?x a h:Person ; h:age ?a  }
```

Question 8

1. **Construct** the symmetric of all hasFriend relations using the good SPARQL statement

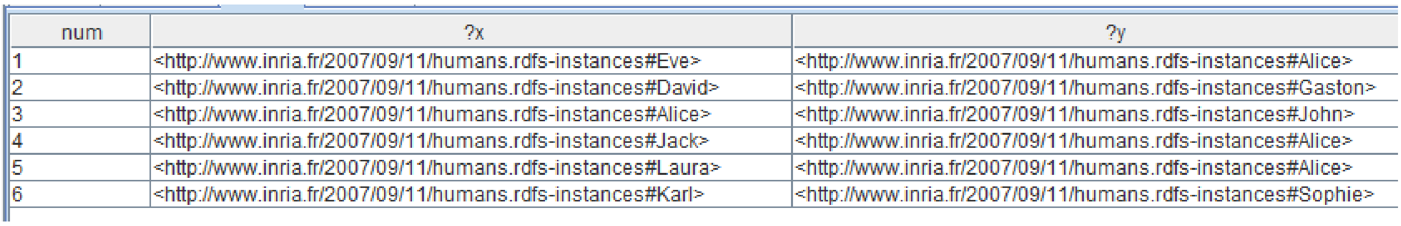
Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT { ?y h:hasFriend ?x } WHERE { ?x h:hasFriend ?y }
```

2. **Insert** the symmetric of all hasFriend relations using the adequate SPARQL statement but check the results with a select before and after.

Query:

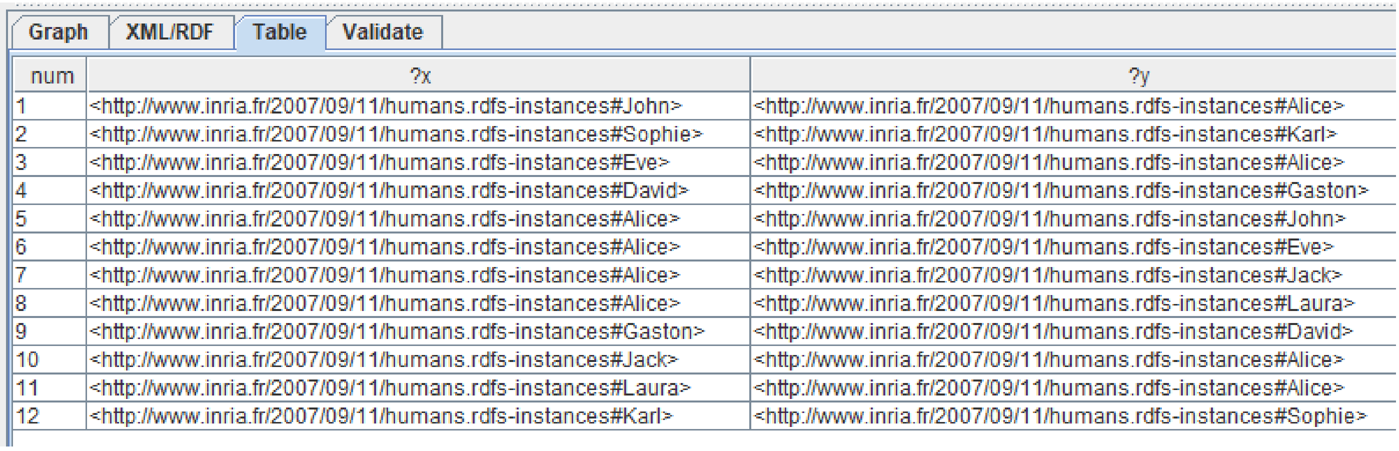
```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
SELECT * WHERE { ?x h:hasFriend ?y }
```



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
INSERT { ?y h:hasFriend ?x } WHERE { ?x h:hasFriend ?y }
```

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



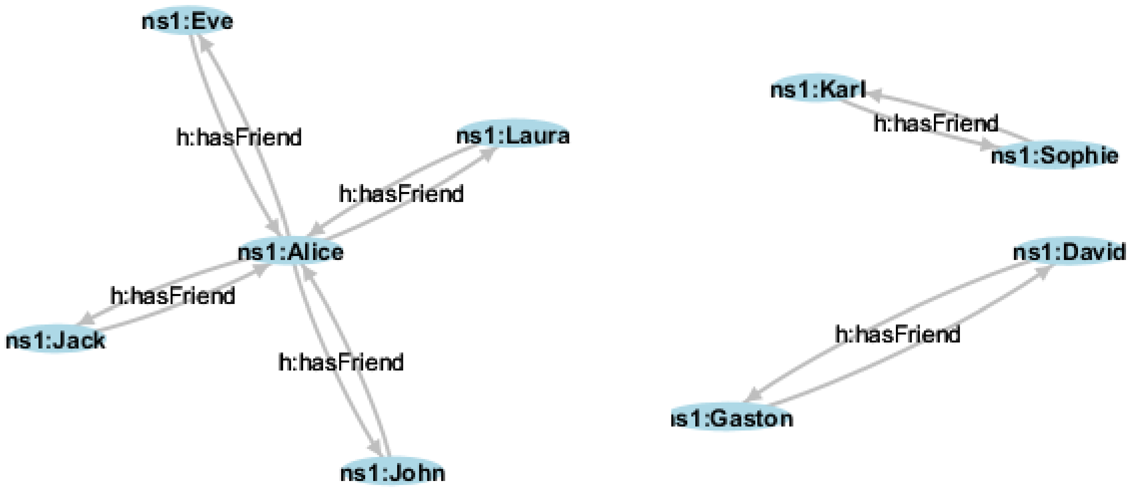
num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>

Question 9

Choose and edit one of the SELECT WHERE queries previously written to transform them into a CONSTRUCT WHERE query (retaining the same WHERE clause) in order to visualize the results as a graph.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
# using the short form (see 16.2.4 CONSTRUCT WHERE in [SPARQL11-QUERY])
CONSTRUCT WHERE { ?x h:hasFriend ?y }
```

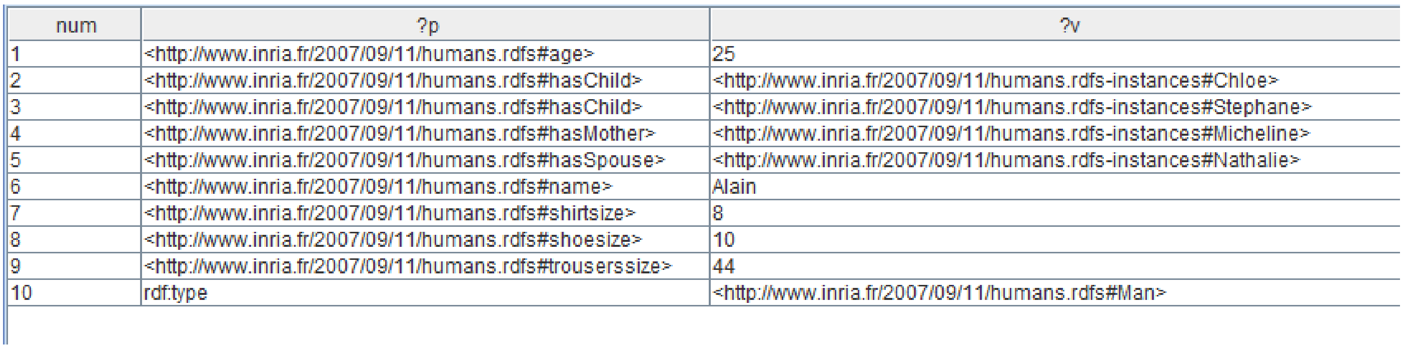


Question 10

Edit the file to add your own annotation (about you) to the RDF file reusing the properties of the file. Build queries to verify and visualize the annotations you added.

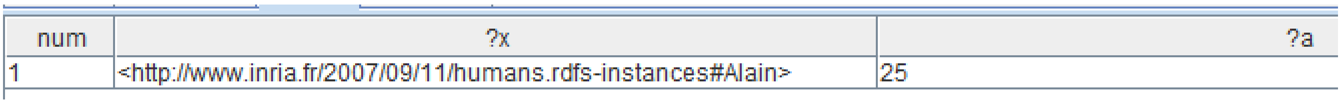
screenshot:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>
SELECT * WHERE {b:Alain ?p ?v }
```



num	?p	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs#age>	25
2	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Chloe>
3	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Stephane>
4	<http://www.inria.fr/2007/09/11/humans.rdfs#hasMother>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Micheline>
5	<http://www.inria.fr/2007/09/11/humans.rdfs#hasSpouse>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Nathalie>
6	<http://www.inria.fr/2007/09/11/humans.rdfs#name>	Alain
7	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>	8
8	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	10
9	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>	44
10	rdf:type	<http://www.inria.fr/2007/09/11/humans.rdfs#Man>

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
SELECT * WHERE {?x h:age ?a FILTER (?a = 25) }
```



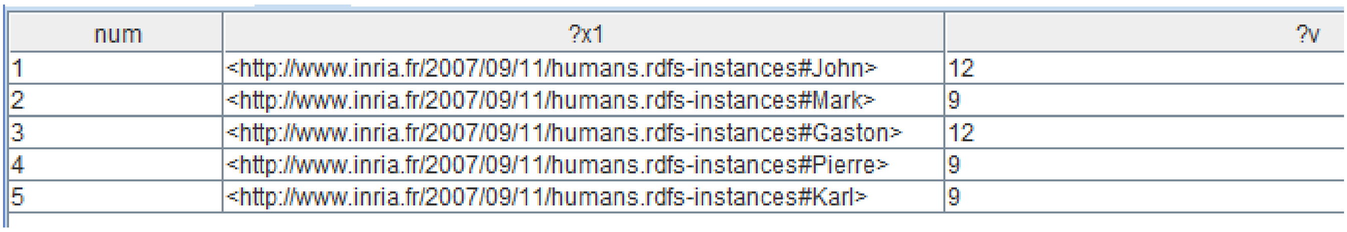
num	?x	?a
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alain>	25

Question 11

1. Formulate a query to find the persons who share the same shirt size.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
SELECT distinct ?x1 ?v
  WHERE {?x1 h:shirtsize ?v . ?x2 h:shirtsize ?v . FILTER (?x1 != ?x2)}
```



num	?x1	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	12
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	9
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	12
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	9
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	9

2. Find the persons who have the same size shirt (shirtsize) and construct a seeAlso relationship between them.

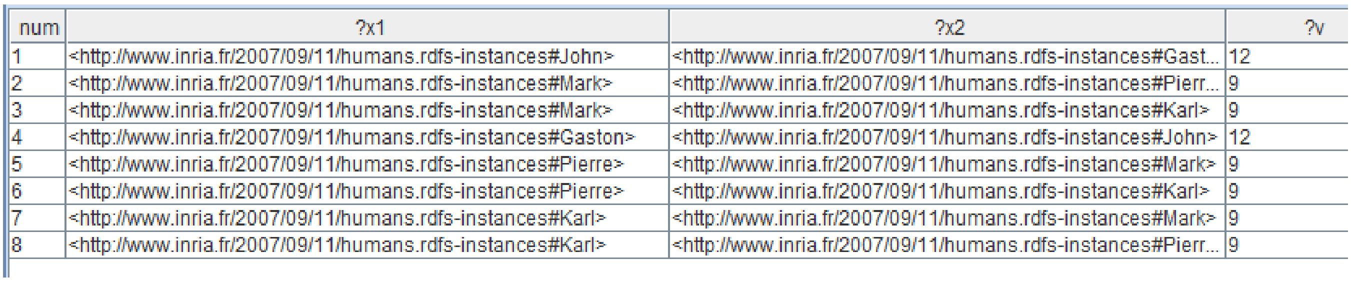
Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT { ?x1 h:seeAlso ?x2 . ?x2 h:seeAlso ?x1 . }
SELECT ?x1 ?x2
  WHERE {?x1 h:shirtsize ?v . ?x2 h:shirtsize ?v . FILTER (?x1 != ?x2)}
```

3. Change the query into an insert.
4. Visualize the resources connected by seeAlso.

insert:

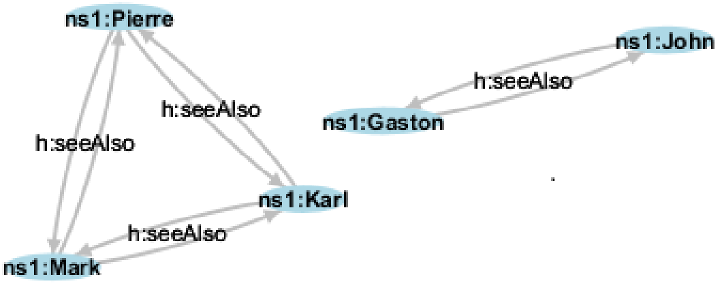
```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
INSERT  { ?x1 h:seeAlso ?x2 . ?x2 h:seeAlso ?x1 . }
WHERE {?x1 h:shirtsize ?v . ?x2 h:shirtsize ?v . FILTER (?x1 != ?x2)}
```



num	?x1	?x2	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gast...	12
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierr...	9
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	9
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	12
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	9
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	9
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	9
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierr...	9

screenshot:

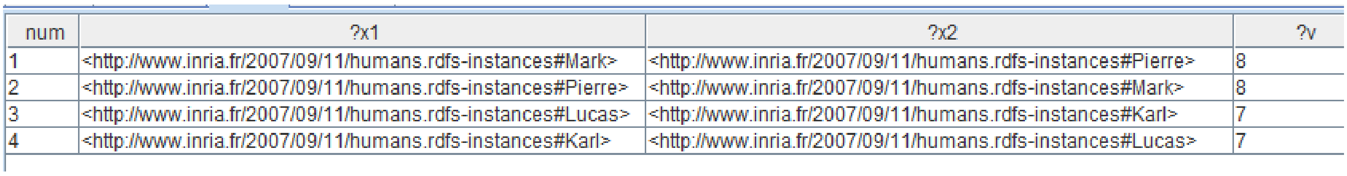
```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT { ?x1 h:seeAlso ?x2 } SELECT * WHERE { ?x1 h:seeAlso ?x2 }
```



5. Adapt the first query to find persons who have same size shoe (shoesize) and insert a seeAlso relationship between them.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
INSERT  { ?x1 h:seeAlso ?x2 . ?x2 h:seeAlso ?x1 . }
WHERE {?x1 h:shoesize ?v . ?x2 h:shoesize ?v . FILTER (?x1 != ?x2)}
```

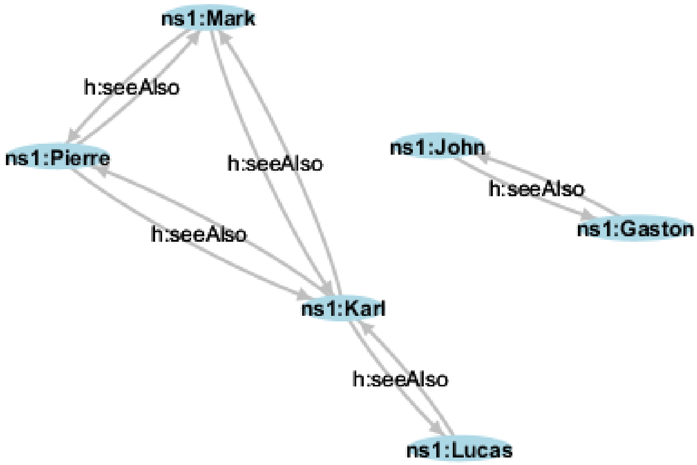


num	?x1	?x2	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	8
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	8
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	7
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	7

6. Visualize the resources connected by seeAlso

screenshot:

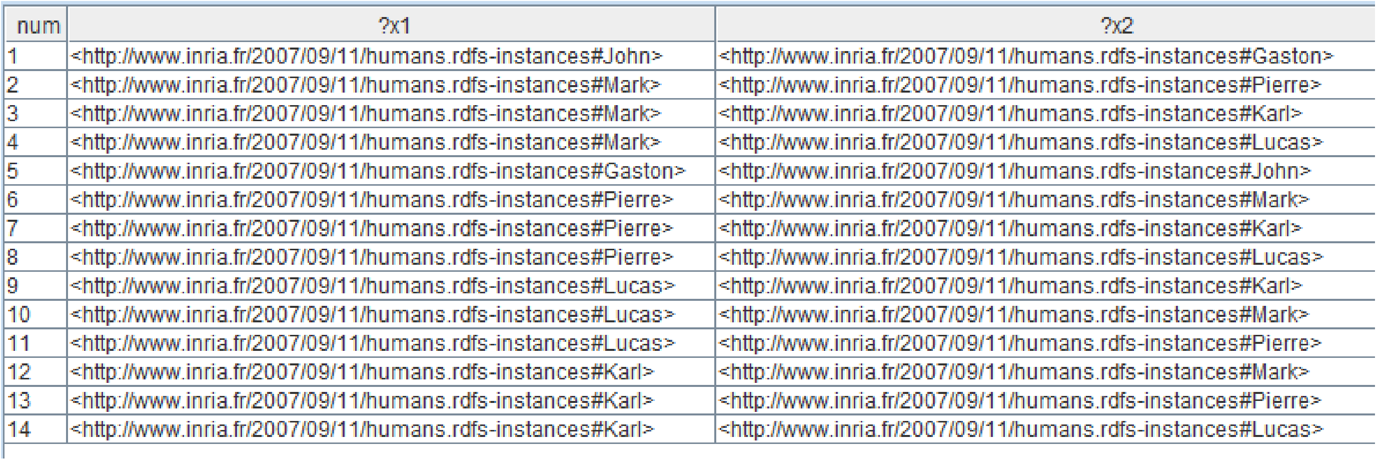
```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT { ?x1 h:seeAlso ?x2 } SELECT * WHERE { ?x1 h:seeAlso ?x2 }
```



7. Change the query to find the resources connected by a path consisting of one or several seeAlso.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
SELECT * WHERE { ?x1 (h:seeAlso)+ ?x2 }
```



num	?x1	?x2
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
13	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
14	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>

8. Reload the engine (option reload in the menu) and rerun the last visualization query.

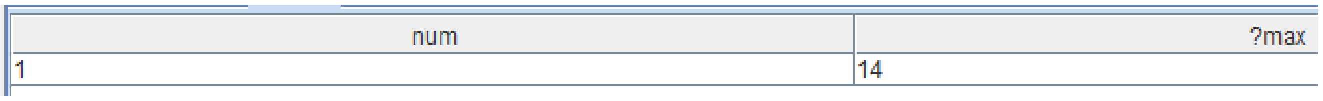
After reloading, nothing is displayed as there is no seeAlso anymore

Question 12

1. Find the largest shoe size

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select (max(?y) as ?max) where { ?x h:shoesize ?y }
```

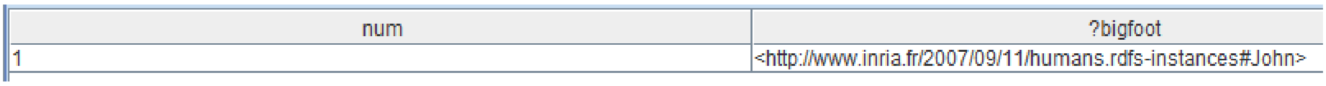


num	?max
1	14

2. Find people who have the biggest size of shoe (subquery + aggregate)

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?bigfoot where {
  { select (max(?y) as ?max) where { ?x h:shoesize ?y } }
?bigfoot h:shoesize  ?max
}
```

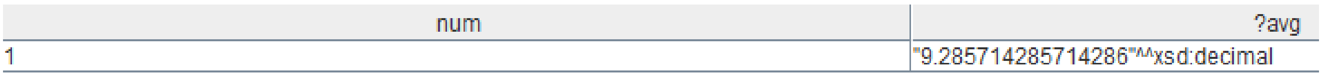


num	?bigfoot
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>

3. Calculate the average shoe size using the appropriate aggregation operator

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select (avg(?y) as ?avg) where { ?x h:shoesize ?y }
```

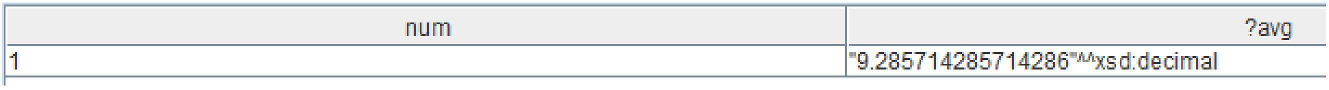


num	?avg
1	"9.285714285714286"^^xsd:decimal

4. Check the average with your own calculation using `sum()` and `count()`

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select (sum(?y)  / count(?x) as ?avg) where { ?x h:shoesize ?y }
```



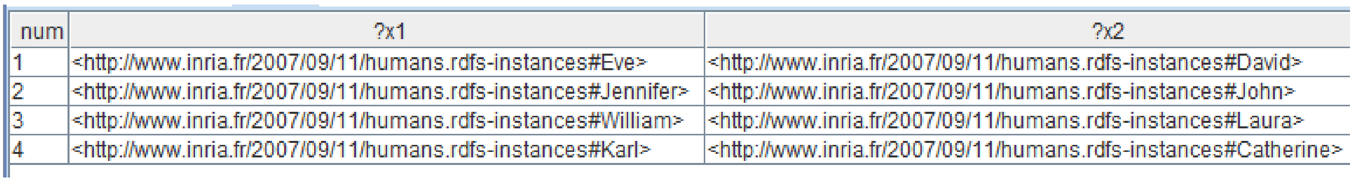
num	?avg
1	"9.285714285714286"^^xsd:decimal

Question 13

Find couples without children

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?x1 ?x2 where { ?x1 h:hasSpouse ?x2 . filter (! (exists { ?x1
h:hasChild ?c1} || exists { ?x2 h:hasChild ?c2} )) }
```



num	?x1	?x2
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>

Question 14

Using insert, create a new person with its properties. Then, check that it has been created.

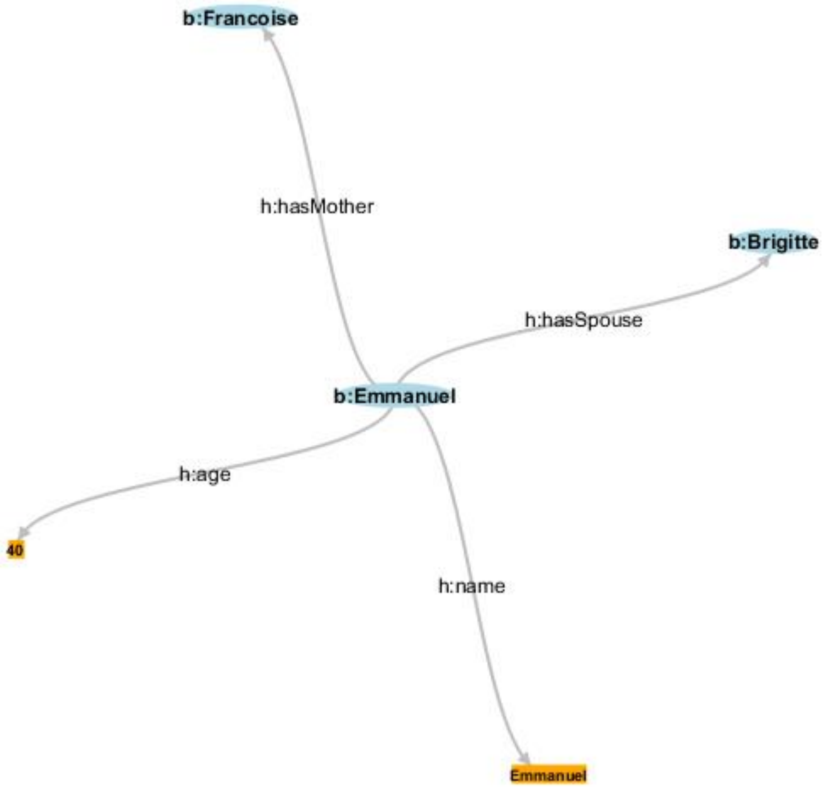
Insert:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>

INSERT DATA {
b:Emmanuel h:name "Emmanuel" ;
h:age 40 ;
h:hasMother b:Francoise ;
h:hasSpouse b:Brigitte.
}
```

Checking that it has been created :

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>

CONSTRUCT WHERE  {
  b:Emmanuel ?p ?y
}
```

Screenshot result:



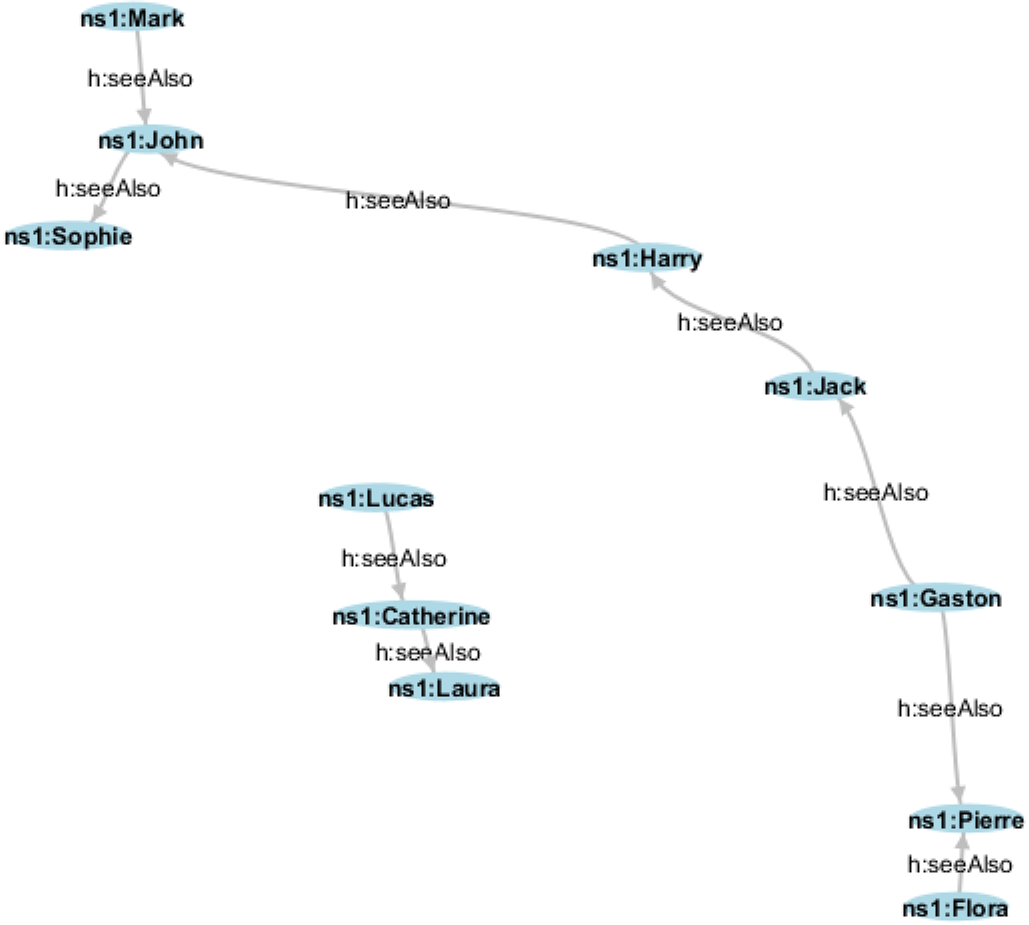
Question 15

Find the people connected by paths of any family links. Construct an arc seeAlso between them to visualize the result.

query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT { ?x h:seeAlso ?y }
SELECT ?x ?y WHERE {
        { ?x h:hasChild ?y }
  UNION { ?x h:hasFather ?y }
  UNION { ?x h:hasMother ?y }
  UNION { ?x h:hasParent ?y }
  UNION { ?x h:Spouse ?y }
}
```

screenshot:



Question 16

Run the following query:

```
prefix db: <http://dbpedia.org/ontology/>
prefix foaf: <http://xmlns.com/foaf/0.1/>
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
construct  { ?x h:name ?nx . ?y h:name ?ny . ?x h:hasSpouse ?y }
where {
  service <http://fr.dbpedia.org/sparql/> {
     select * where {
       ?x db:spouse ?y .
       ?x foaf:name ?nx .
       ?y foaf:name ?ny .
     }
     limit 20
  }
}
```

Explain what it does

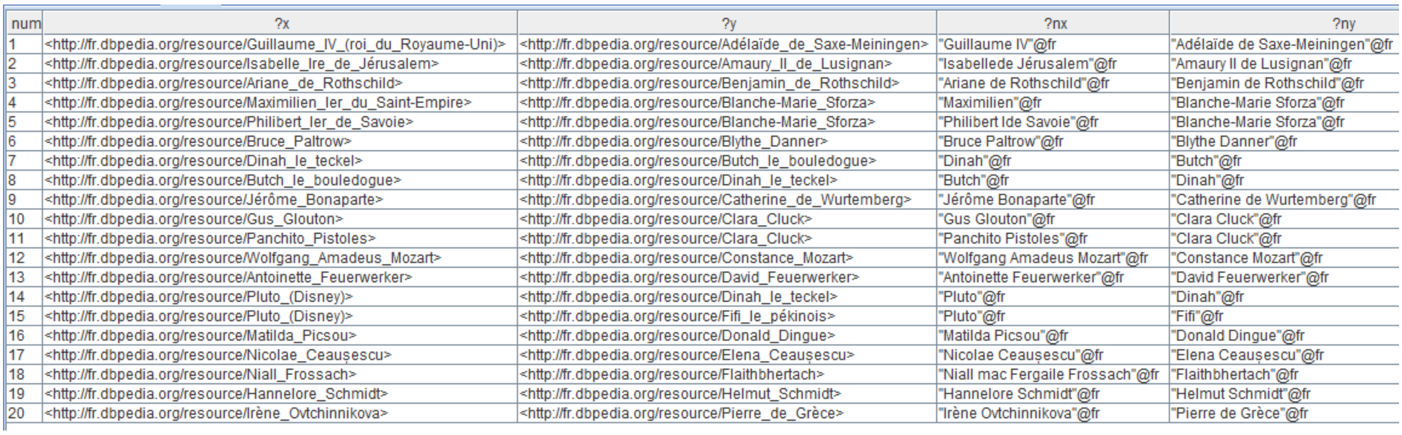
This request performs a remote access to the SPARQL endpoint http://fr.dbpedia.org, querying for 20 couples and retrieving the name of each spouse.

modify it to insert new persons in the base and check the results.

query:

If the “construct” keyword is replaced by “insert” in the request above, the persons retrieved from http://fr.dbpedia.org are inserted in the local database. They can be retrieved as follows :

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where {
       ?x h:hasSpouse ?y .
       ?x h:name ?nx .
       ?y h:name ?ny .
}
```



num	?x	?y	?nx	?ny
1	<http://fr.dbpedia.org/resource/Guillaume_IV_(roi_du_Royaume-Uni)>	<http://fr.dbpedia.org/resource/Adélaïde_de_Saxe-Meiningen>	"Guillaume IV"@fr	"Adélaïde de Saxe-Meiningen"@fr
2	<http://fr.dbpedia.org/resource/Isabelle_Ire_de_Jérusalem>	<http://fr.dbpedia.org/resource/Amaury_II_de_Lusignan>	"Isabellede Jérusalem"@fr	"Amaury II de Lusignan"@fr
3	<http://fr.dbpedia.org/resource/Ariane_de_Rothschild>	<http://fr.dbpedia.org/resource/Benjamin_de_Rothschild>	"Ariane de Rothschild"@fr	"Benjamin de Rothschild"@fr
4	<http://fr.dbpedia.org/resource/Maximilien_Ier_du_Saint-Empire>	<http://fr.dbpedia.org/resource/Blanche-Marie_Sforza>	"Maximilien"@fr	"Blanche-Marie Sforza"@fr
5	<http://fr.dbpedia.org/resource/Philibert_Ier_de_Savoie>	<http://fr.dbpedia.org/resource/Blanche-Marie_Sforza>	"Philibert Ide Savoie"@fr	"Blanche-Marie Sforza"@fr
6	<http://fr.dbpedia.org/resource/Bruce_Paltrow>	<http://fr.dbpedia.org/resource/Blythe_Danner>	"Bruce Paltrow"@fr	"Blythe Danner"@fr
7	<http://fr.dbpedia.org/resource/Dinah_le_teckel>	<http://fr.dbpedia.org/resource/Butch_le_bouledogue>	"Dinah"@fr	"Butch"@fr
8	<http://fr.dbpedia.org/resource/Butch_le_bouledogue>	<http://fr.dbpedia.org/resource/Dinah_le_teckel>	"Butch"@fr	"Dinah"@fr
9	<http://fr.dbpedia.org/resource/Jérôme_Bonaparte>	<http://fr.dbpedia.org/resource/Catherine_de_Wurtemberg>	"Jérôme Bonaparte"@fr	"Catherine de Wurtemberg"@fr
10	<http://fr.dbpedia.org/resource/Gus_Glouton>	<http://fr.dbpedia.org/resource/Clara_Cluck>	"Gus Glouton"@fr	"Clara Cluck"@fr
11	<http://fr.dbpedia.org/resource/Panchito_Pistoles>	<http://fr.dbpedia.org/resource/Clara_Cluck>	"Panchito Pistoles"@fr	"Clara Cluck"@fr
12	<http://fr.dbpedia.org/resource/Wolfgang_Amadeus_Mozart>	<http://fr.dbpedia.org/resource/Constance_Mozart>	"Wolfgang Amadeus Mozart"@fr	"Constance Mozart"@fr
13	<http://fr.dbpedia.org/resource/Antoinette_Feuerwerker>	<http://fr.dbpedia.org/resource/David_Feuerwerker>	"Antoinette Feuerwerker"@fr	"David Feuerwerker"@fr
14	<http://fr.dbpedia.org/resource/Pluto_(Disney)>	<http://fr.dbpedia.org/resource/Dinah_le_teckel>	"Pluto"@fr	"Dinah"@fr
15	<http://fr.dbpedia.org/resource/Pluto_(Disney)>	<http://fr.dbpedia.org/resource/Fifi_le_pékinois>	"Pluto"@fr	"Fifi"@fr
16	<http://fr.dbpedia.org/resource/Matilda_Picsou>	<http://fr.dbpedia.org/resource/Donald_Dingue>	"Matilda Picsou"@fr	"Donald Dingue"@fr
17	<http://fr.dbpedia.org/resource/Nicolae_Ceaușescu>	<http://fr.dbpedia.org/resource/Elena_Ceaușescu>	"Nicolae Ceaușescu"@fr	"Elena Ceaușescu"@fr
18	<http://fr.dbpedia.org/resource/Niall_Frossach>	<http://fr.dbpedia.org/resource/Flaithbhertach>	"Niall mac Fergaile Frossach"@fr	"Flaithbhertach"@fr
19	<http://fr.dbpedia.org/resource/Hannelore_Schmidt>	<http://fr.dbpedia.org/resource/Helmut_Schmidt>	"Hannelore Schmidt"@fr	"Helmut Schmidt"@fr
20	<http://fr.dbpedia.org/resource/Irène_Ovtchinnikova>	<http://fr.dbpedia.org/resource/Pierre_de_Grèce>	"Irène Ovtchinnikova"@fr	"Pierre de Grèce"@fr

Day 04: questions from the course on RDFS.

Q4.1 Choose among the following assertions one or more you consider to be true:

- ~~□ an ontology is necessarily formalized in first-order logic~~
- □ an ontology may allow inferences on data that uses it
- □ conceptual graphs can represent an ontology
- □ a shared ontology promotes interoperability
- □ description logics can represent an ontology

all true, except the one strikethrough above

Q4.2 RDFS contains primitives to (several answers possible)…

- □ describe classes of resources
- ~~□ describe formulas of calculation for values of properties~~
- □ describe types of properties of resources
- □ document definitions in natural language
- ~~□ sign and authenticate the authors of the definitions of classes and properties~~

all true, except the ones strikethrough above

Q4.3. What is defined and derived from these definitions?

```
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
@base  <http://inria.fr/2005/humans.rdfs>
 <B> rdfs:subClassOf  <A> .
 <C> rdfs:subClassOf  <A> .
 <D> rdfs:subClassOf  <B> .
 <D> rdfs:subClassOf  <C> .
```

```
<B> rdfs:subClassOf  <A> .
<C> rdfs:subClassOf  <A> .
<D> rdfs:subClassOf  <B> .
<D> rdfs:subClassOf  <C> .
<D> rdfs:subClassOf  <A> .
```

Q4.4. What is defined and derived from these definitions?

```
@prefix rdfs:  < http://www.w3.org/2000/01/rdf-schema# >
@base < http://inria.fr/2005/humans.rdfs >
<P2> rdfs:subPropertyOf  <P1> .
<P3> rdfs:subPropertyOf  <P1>.
<P4> rdfs:subPropertyOf  <P2>,  <P3>.
```

```
<P2> rdfs:subPropertyOf  <P1> .
<P3> rdfs:subPropertyOf  <P1> .
<P4> rdfs:subPropertyOf  <P2>,  <P3>, <P1> .
```

Q4.5. What is defined and derived from these definitions?

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
@base <http://inria.fr/2005/humans.rdfs>
<P1>  rdfs:subPropertyOf  <P2> .
<P2>  rdfs:domain  <B> ;  rdfs:range  <C> .
<P1>  rdfs:domain  <A> .
```

```
<P1>  rdfs:subPropertyOf  <P2> .
<P2>  rdfs:domain  <B> ;  rdfs:range  <C> .
<P1>  rdfs:domain  <A> .
<P1>  rdfs:domain  <B> ;  rdfs:range  <C> .
```

Q4.6. What could we add to this schema (several answers are possible)?

```
@prefix rdf:   <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
@prefix rdfs:  <http://www.w3.org/2000/01/rdf-schema#>
@base  <http://inria.fr/2005/humans.rdfs>
<p1>  a rdf:Property ;  rdfs:label  "age"@fr .
<c1>  a rdfs:Class; rdfs:comment "un être humain"@fr .
```

~~□ <p1> rdfs:label "prénom"@fr .~~

~~□ <c1> rdfs:comment "a human being"@fr .~~

~~□~~ <c1> rdfs:label "personne"@fr .

□ <p1> rdfs:label "age"@en .

~~□ <c1> rdfs:label "woman"@en .~~

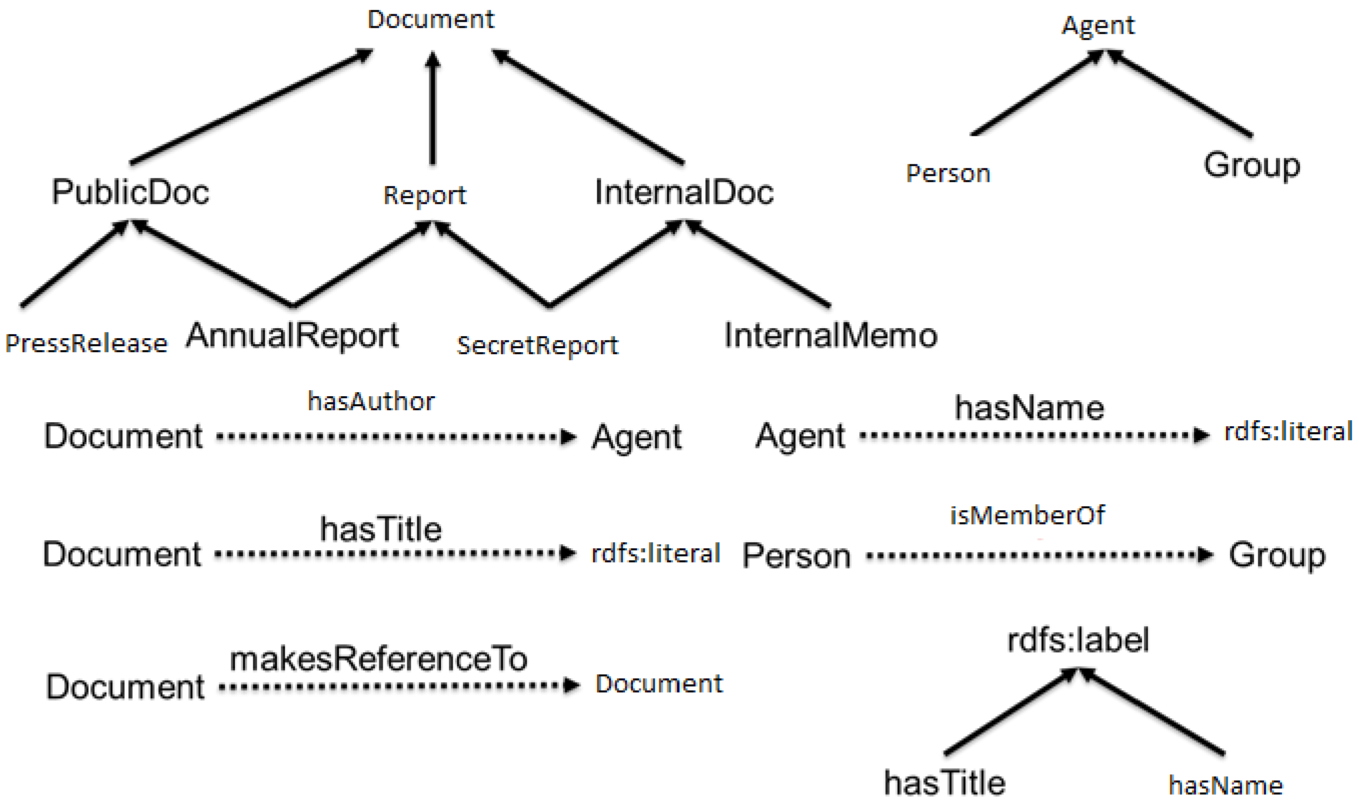
□ <c1> rdfs:label "persona"@es .

all true, except the ones strikethrough above

Q4.7. (a) Fill the blanks with: `Document`, `PublicDoc`, `PressRelease`, `Report`, `AnnualReport`, `InternalDoc`, `SecretReport`, `InternalMemo`, `Agent`, `Person`, `Group`, `hasTitle`, `hasAuthor`, `makesReferenceTo`, `hasName`, `isMemberOf` + rdf / rdfs primitives.

(b) Write it in RDFS and validate the RDF.

(a)



(b)

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
    <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
    <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
]>

<rdf:RDF
  xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://alain.tholon.dsti.institute/20180222/q4.7.rdfs"
>
  <rdfs:Class rdf:about="#PublicDocument">
    <rdfs:subClassOf rdf:resource="#Document"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#Report">
    <rdfs:subClassOf rdf:resource="#Document"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#InternalDoc">
    <rdfs:subClassOf rdf:resource="#Document"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#PressRelease">
    <rdfs:subClassOf rdf:resource="#PublicDocument"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#InternalMemo">
    <rdfs:subClassOf rdf:resource="#InternalDoc"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#AnnualReport">
    <rdfs:subClassOf rdf:resource="#PublicDocument"/>
    <rdfs:subClassOf rdf:resource="#Report"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#SecretReport">
    <rdfs:subClassOf rdf:resource="#InternalDoc"/>
    <rdfs:subClassOf rdf:resource="#Report"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#Group">
    <rdfs:subClassOf rdf:resource="#Agent"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#hasTitle">
    <rdfs:subClassOf rdf:resource="&rdfs;label"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#hasName">
    <rdfs:subClassOf rdf:resource="&rdfs;label"/>
  </rdfs:Class>

  <rdfs:Class rdf:about="#Person">
```

```
    <rdfs:subClassOf rdf:resource="#Agent"/>
  </rdfs:Class>

  <rdf:Property rdf:ID="hasAuthor">
    <rdfs:domain rdf:resource="#Document"/>
    <rdfs:range  rdf:resource="#Agent"/>
  </rdf:Property>

  <rdf:Property rdf:ID="hasTitle">
    <rdfs:domain rdf:resource="#Document"/>
    <rdfs:range  rdf:resource="&rdfs;Literal"/>
  </rdf:Property>

  <rdf:Property rdf:ID="makesReferenceTo">
    <rdfs:domain rdf:resource="#Document"/>
    <rdfs:range  rdf:resource="#Document"/>
  </rdf:Property>

  <rdf:Property rdf:ID="hasName">
    <rdfs:domain rdf:resource="#Agent"/>
    <rdfs:range  rdf:resource="&rdfs;Literal"/>
  </rdf:Property>

  <rdf:Property rdf:ID="isMemberOf">
    <rdfs:domain rdf:resource="#Person"/>
    <rdfs:range  rdf:resource="#Group"/>
  </rdf:Property>

</rdf:RDF>
```

Day 04: Answers to the practical session on RDFS.

Software requirements

- The RDF XML online validation service by W3C: https://www.w3.org/RDF/Validator/
- The RDF online translator: http://rdf-translator.appspot.com/
- The SPARQL Corese engine: http://wimmics.inria.fr/corese

Create your own schema Family.rdfs

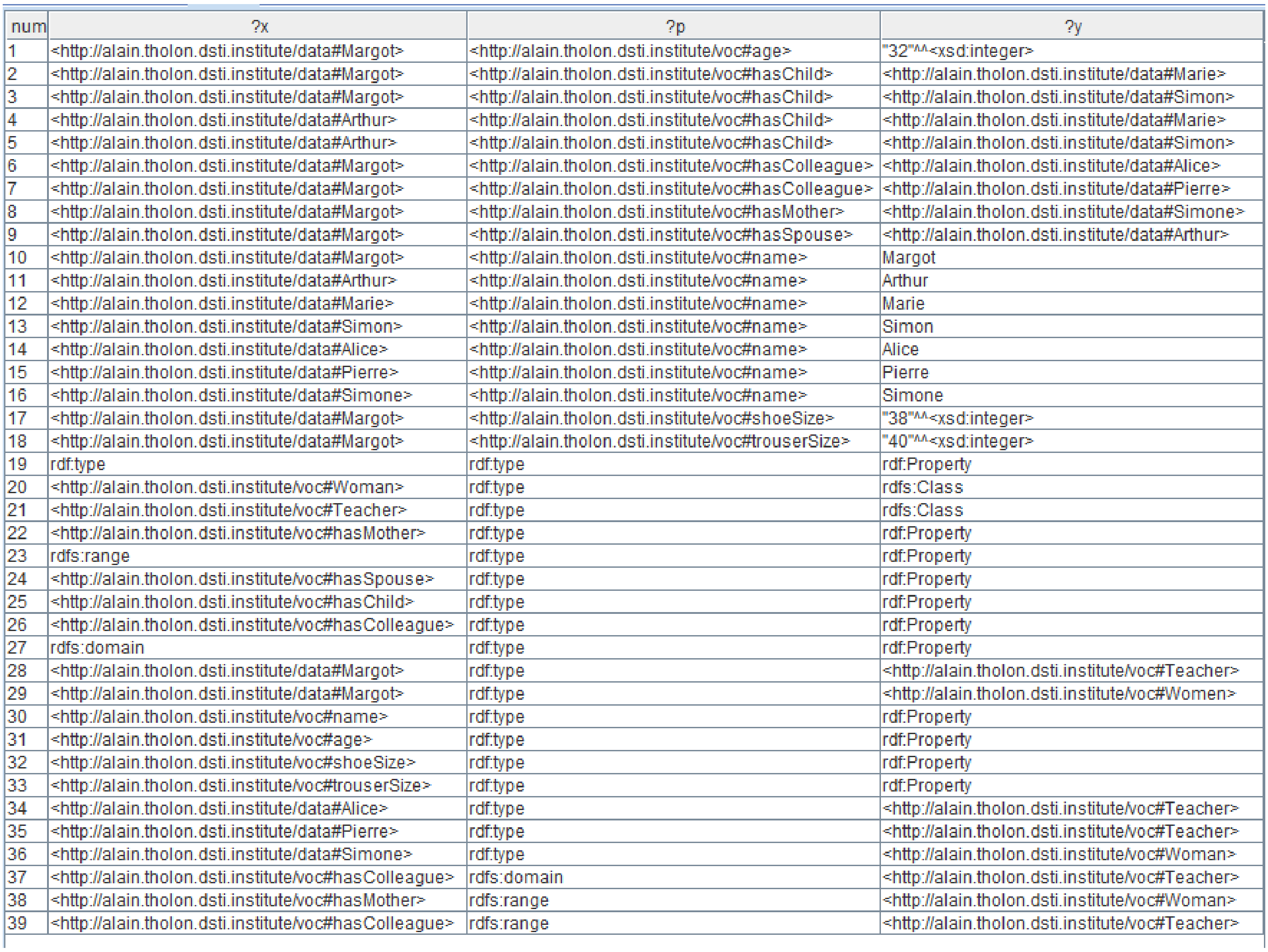
- Write the the RDF schema that you used in the description of Margot in a RDF/XML (or in turtle and then translate it) and save the RDF/XML in a file called “Family.rdfs”. Of course, this assumes that the URIs for the classes and properties declared/used must match in both files. You mays have to update the files Margot.rdf and Margot.ttl to use your ontology.

Your schema:

```
<?xml version="1.0"?>

<!DOCTYPE rdf:RDF [
    <!ENTITY rdf "http://www.w3.org/1999/02/22-rdf-syntax-ns#" >
    <!ENTITY rdfs "http://www.w3.org/2000/01/rdf-schema#" >
]>

<rdf:RDF
  xmlns:rdf ="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
  xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
  xmlns="http://www.w3.org/2000/01/rdf-schema#"
  xml:base="http://alain.tholon.dsti.institute/voc"
>

<Class rdf:ID="Woman"></Class>

<Class rdf:ID="Teacher"></Class>

<rdf:Property rdf:ID="hasMother">
  <range rdf:resource="#Woman"/>
</rdf:Property>

<rdf:Property rdf:ID="hasSpouse">
</rdf:Property>

<rdf:Property rdf:ID="hasChild">
</rdf:Property>

<rdf:Property rdf:ID="hasColleague">
  <domain rdf:resource="#Teacher"/>
  <range  rdf:resource="#Teacher"/>
</rdf:Property>

</rdf:RDF>
```

- Check that your RDF schema and RDF files are valid using the W3C's RDF validation service.
- Launch the standalone interface of Corese and load your files Family.rdfs and Margot.rdf
- The interface contains a default SPARQL query:
  `Select? x? t where {? x RDF: type? t}`
  Launch the query and examine the results.

Screenshot:

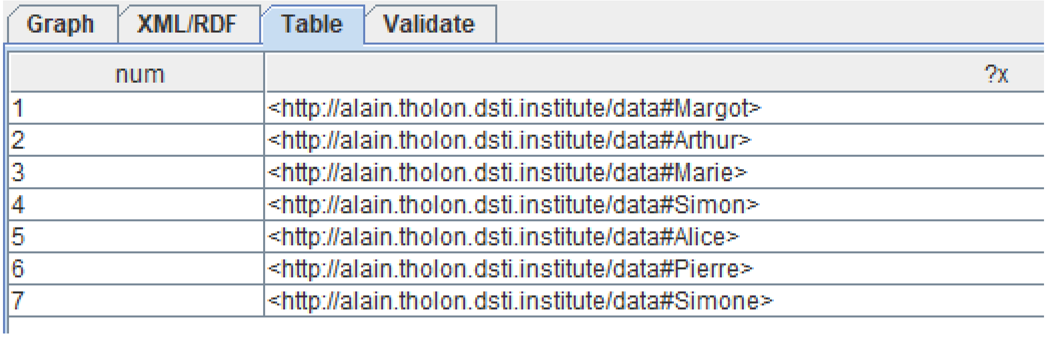


num	?x	?p	?y
1	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#age>	"32"^^<xsd:integer>
2	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasChild>	<http://alain.tholon.dsti.institute/data#Marie>
3	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasChild>	<http://alain.tholon.dsti.institute/data#Simon>
4	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/voc#hasChild>	<http://alain.tholon.dsti.institute/data#Marie>
5	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/voc#hasChild>	<http://alain.tholon.dsti.institute/data#Simon>
6	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasColleague>	<http://alain.tholon.dsti.institute/data#Alice>
7	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasColleague>	<http://alain.tholon.dsti.institute/data#Pierre>
8	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasMother>	<http://alain.tholon.dsti.institute/data#Simone>
9	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#hasSpouse>	<http://alain.tholon.dsti.institute/data#Arthur>
10	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#name>	Margot
11	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/voc#name>	Arthur
12	<http://alain.tholon.dsti.institute/data#Marie>	<http://alain.tholon.dsti.institute/voc#name>	Marie
13	<http://alain.tholon.dsti.institute/data#Simon>	<http://alain.tholon.dsti.institute/voc#name>	Simon
14	<http://alain.tholon.dsti.institute/data#Alice>	<http://alain.tholon.dsti.institute/voc#name>	Alice
15	<http://alain.tholon.dsti.institute/data#Pierre>	<http://alain.tholon.dsti.institute/voc#name>	Pierre
16	<http://alain.tholon.dsti.institute/data#Simone>	<http://alain.tholon.dsti.institute/voc#name>	Simone
17	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#shoeSize>	"38"^^<xsd:integer>
18	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/voc#trouserSize>	"40"^^<xsd:integer>
19	rdf:type	rdf:type	rdf:Property
20	<http://alain.tholon.dsti.institute/voc#Woman>	rdf:type	rdfs:Class
21	<http://alain.tholon.dsti.institute/voc#Teacher>	rdf:type	rdfs:Class
22	<http://alain.tholon.dsti.institute/voc#hasMother>	rdf:type	rdf:Property
23	rdfs:range	rdf:type	rdf:Property
24	<http://alain.tholon.dsti.institute/voc#hasSpouse>	rdf:type	rdf:Property
25	<http://alain.tholon.dsti.institute/voc#hasChild>	rdf:type	rdf:Property
26	<http://alain.tholon.dsti.institute/voc#hasColleague>	rdf:type	rdf:Property
27	rdfs:domain	rdf:type	rdf:Property
28	<http://alain.tholon.dsti.institute/data#Margot>	rdf:type	<http://alain.tholon.dsti.institute/voc#Teacher>
29	<http://alain.tholon.dsti.institute/data#Margot>	rdf:type	<http://alain.tholon.dsti.institute/voc#Women>
30	<http://alain.tholon.dsti.institute/voc#name>	rdf:type	rdf:Property
31	<http://alain.tholon.dsti.institute/voc#age>	rdf:type	rdf:Property
32	<http://alain.tholon.dsti.institute/voc#shoeSize>	rdf:type	rdf:Property
33	<http://alain.tholon.dsti.institute/voc#trouserSize>	rdf:type	rdf:Property
34	<http://alain.tholon.dsti.institute/data#Alice>	rdf:type	<http://alain.tholon.dsti.institute/voc#Teacher>
35	<http://alain.tholon.dsti.institute/data#Pierre>	rdf:type	<http://alain.tholon.dsti.institute/voc#Teacher>
36	<http://alain.tholon.dsti.institute/data#Simone>	rdf:type	<http://alain.tholon.dsti.institute/voc#Woman>
37	<http://alain.tholon.dsti.institute/voc#hasColleague>	rdfs:domain	<http://alain.tholon.dsti.institute/voc#Teacher>
38	<http://alain.tholon.dsti.institute/voc#hasMother>	rdfs:range	<http://alain.tholon.dsti.institute/voc#Woman>
39	<http://alain.tholon.dsti.institute/voc#hasColleague>	rdfs:range	<http://alain.tholon.dsti.institute/voc#Teacher>

- Modify your ontology to declare the class of Humans as a super class of Man and Woman (don’t change the data), reload the schemas and data and search for the humans to see the results

Screenshot:

Added Human as the superclass of Woman, Teacher and as domain and range properties when appropriate

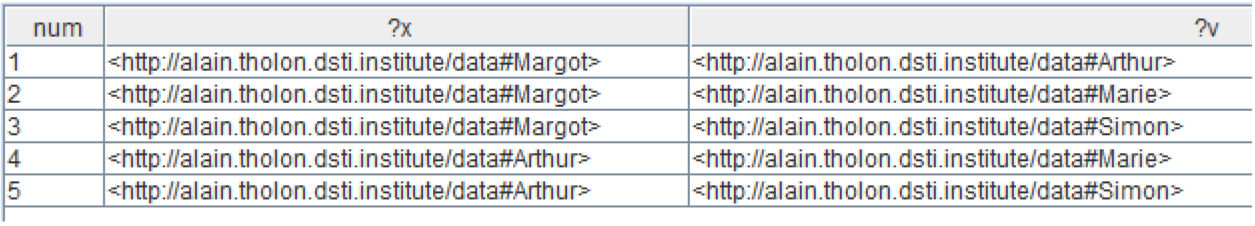


num	?x
1	<http://alain.tholon.dsti.institute/data#Margot>
2	<http://alain.tholon.dsti.institute/data#Arthur>
3	<http://alain.tholon.dsti.institute/data#Marie>
4	<http://alain.tholon.dsti.institute/data#Simon>
5	<http://alain.tholon.dsti.institute/data#Alice>
6	<http://alain.tholon.dsti.institute/data#Pierre>
7	<http://alain.tholon.dsti.institute/data#Simone>

- Modify your ontology to declare the property of familyLink as a super property of hasChild and hasSpouse (don’t change the data), reload the schemas and data and search for the family links to see the results.

Screenshot:

```
prefix voc: <http://alain.tholon.dsti.institute/voc#>
select * where { ?x voc:familyLink ?v }
```

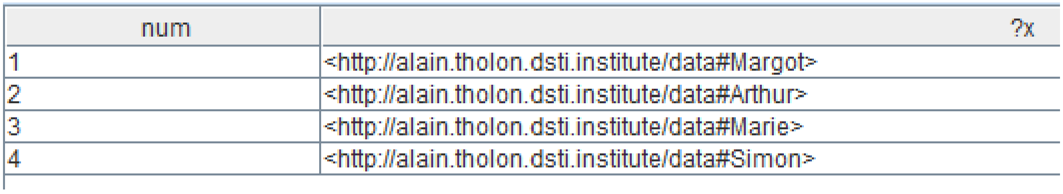


num	?x	?v
1	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Arthur>
2	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Marie>
3	<http://alain.tholon.dsti.institute/data#Margot>	<http://alain.tholon.dsti.institute/data#Simon>
4	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/data#Marie>
5	<http://alain.tholon.dsti.institute/data#Arthur>	<http://alain.tholon.dsti.institute/data#Simon>

- Modify your ontology to declare the class FamilyMember and use it to specify the signature of the property familyLink (don’t change the data) then reload the schemas and data and search for the family members.

Screenshot:

```
prefix voc: <http://alain.tholon.dsti.institute/voc#>
select * where { ?x rdf:type voc:FamilyMember }
```



num	?x
1	<http://alain.tholon.dsti.institute/data#Margot>
2	<http://alain.tholon.dsti.institute/data#Arthur>
3	<http://alain.tholon.dsti.institute/data#Marie>
4	<http://alain.tholon.dsti.institute/data#Simon>

About the human.rdfs schema

1. Download the RDF schema available at this address and save it as “human.rdfs”: http://wimmics.inria.fr/doc/tutorial/human_2013.rdfs
2. What is the namespace associated with this ontology? How was it associated?

The namespace associated with this ontology is “http://www.inria.fr/2007/09/11/humans.rdfs”
It is associated to this ontology through xml:base="http://www.inria.fr/2007/09/11/humans.rdfs because ID attribute are relative URI.

3. Look at the XML structure of this file and locate different syntactic properties: the different possible uses of the markup (ex: opening tag and closing, single tag), the use of namespaces for qualified names, the use of entities, etc.
4. Locate the use of the terms of the RDF (S) language: Class, Property, label, how, range, domain, subClassOf, subPropertyOf, etc. To what namespaces are they associated?

All XML tags that do not have a prefix are automatically prefixed by the default namespace that is defined by xmlns="…". In this case it is http://www.w3.org/2000/01/rdf-schema#

Class, label, range, comment, domain, subClassOf, subPropertyOf are associated to http://www.w3.org/2000/01/rdf-schema#

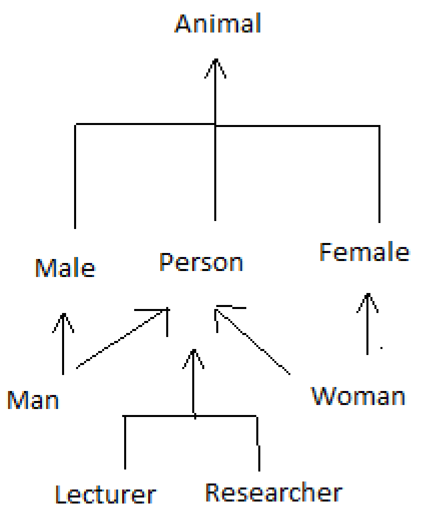
Property is associated to http://www.w3.org/1999/02/22-rdf-syntax-ns#

5. What are the classes of resources that can have the age property?

As the age property has no domain and range property, it can be applied from any resource to any resource.

6. Look at the beginning of the file and draw the subgraph of the hierarchy containing the classes `Animal`, `Man` and `Woman`.

Drawing of hierarchy:



Query the schema itself

Reset or relaunch the standalone Corese search engine interface and load the file `human`**`.rdfs`** (and only this one).

1. Write a query to find all the classes of the ontology.

   query:

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

select * where {
  ?x rdf:type rdfs:Class
}
```

2. Write a query to find all the links subClassOf in the ontology.

   query:

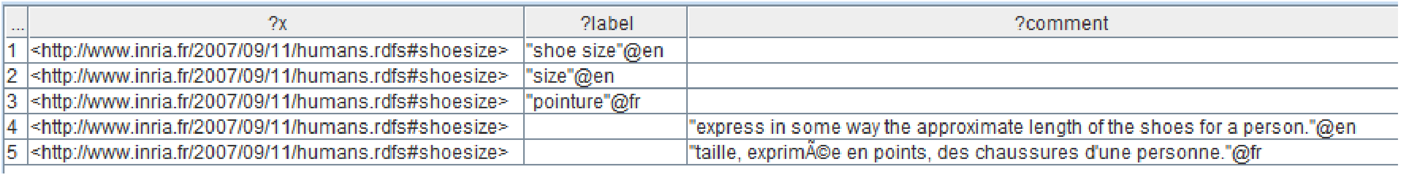
```
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>

select * where {
  ?x rdfs:subClassOf ?y
}
```

3. Write a query to find the definitions and translations of "shoe size" (other labels and comments in different languages for the resource labeled "shoe size").

   query:

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

select * where {
  ?x rdfs:label "shoe size"@en
  { ?x rdfs:label ?label . }
  UNION
 { ?x rdfs:comment ?comment . }
}
```

   answers:



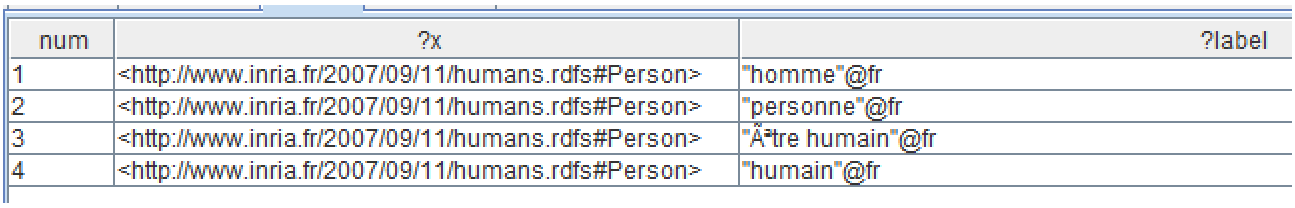
...	?x	?label	?comment
1	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	"shoe size"@en	
2	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	"size"@en	
3	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	"pointure"@fr	
4	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>		"express in some way the approximate length of the shoes for a person."@en
5	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>		"taille, exprimÃ©e en points, des chaussures d'une personne."@fr

4. Write a query to find the synonyms in French of the word 'personne' in French (other labels in the same language for the same resource/class/property). What are the answers?

query:

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

select * where {
  ?x rdfs:label "personne"@fr
  { ?x rdfs:label ?label filter ( lang(?label)="fr") . }
}
```

answers:



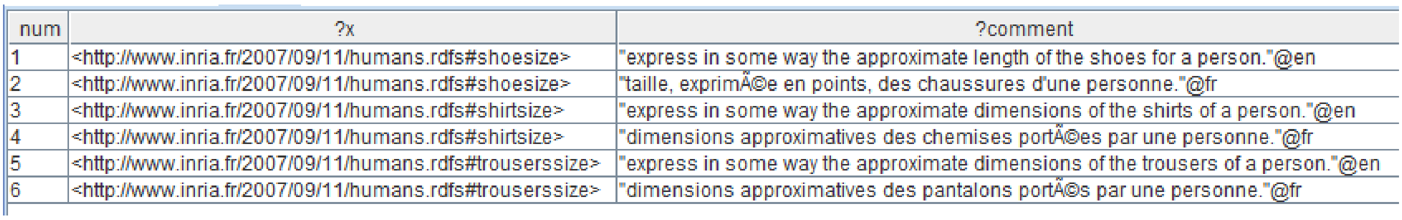
num	?x	?label
1	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>	"homme"@fr
2	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>	"personne"@fr
3	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>	"Ãªtre humain"@fr
4	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>	"humain"@fr

5. Write a query to find the different meaning of the term "size" (disambiguation using the different comments attached to different resources/classes/properties having the label "size"). What are the answers?

query:

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

select * where {
  ?x rdfs:label "size"@en
 { ?x rdfs:comment ?comment . }
}
```

answers:

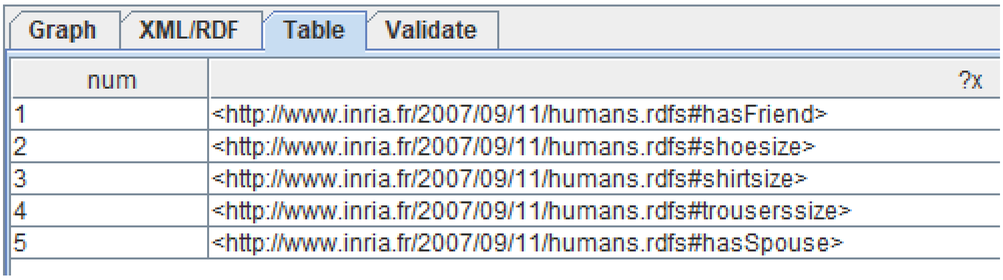


num	?x	?comment
1	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	"express in some way the approximate length of the shoes for a person."@en
2	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	"taille, exprimÃ©e en points, des chaussures d'une personne."@fr
3	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>	"express in some way the approximate dimensions of the shirts of a person."@en
4	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>	"dimensions approximatives des chemises portÃ©es par une personne."@fr
5	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>	"express in some way the approximate dimensions of the trousers of a person."@en
6	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>	"dimensions approximatives des pantalons portÃ©s par une personne."@fr

6. Write a query to find the properties that use the class Person in their signatures?

query:

```
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs#>

select distinct * where {
  ?x rdf:type rdf:Property .
  { ?x rdfs:domain b:Person . }
  UNION
  { ?x rdfs:range b:Person . }
}
```



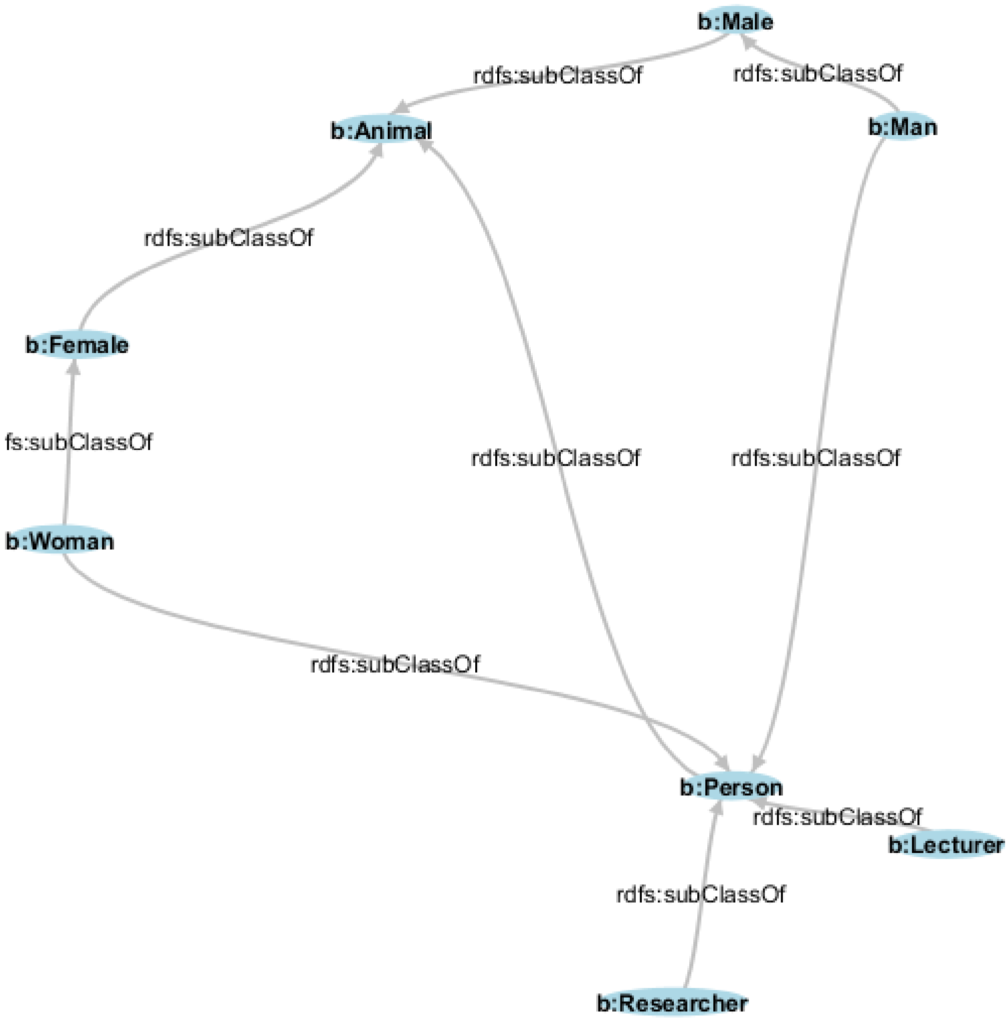
num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend>
2	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>
3	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>
4	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>
5	<http://www.inria.fr/2007/09/11/humans.rdfs#hasSpouse>

7. Rebuild the hierarchy of Classes (CONSTRUCT) considering only the classes in the humans.rdfs schema

query:

```
prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix b: <http://www.inria.fr/2007/09/11/humans.rdfs#>

CONSTRUCT {  ?y rdfs:subClassOf ?x }
select * where {
  ?y rdfs:subClassOf ?x .
}
```

screenshot:



8. To the previous CONSTRUCT add the signatures of the relations.

query:

<ANSWER HERE/>

screenshot:

<ANSWER HERE/>

You now know how to query schemas on the semantic Web!

Query data augmented by an RDFS schema

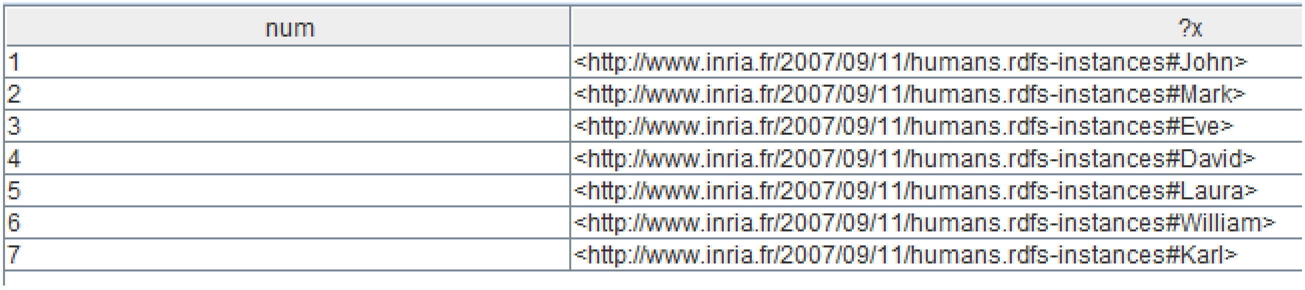
Question 1

1. Reset the Corese engine and load only the annotations (.rdf)
2. Write a query to find the Persons.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?x where { ?x a h:Person }
```

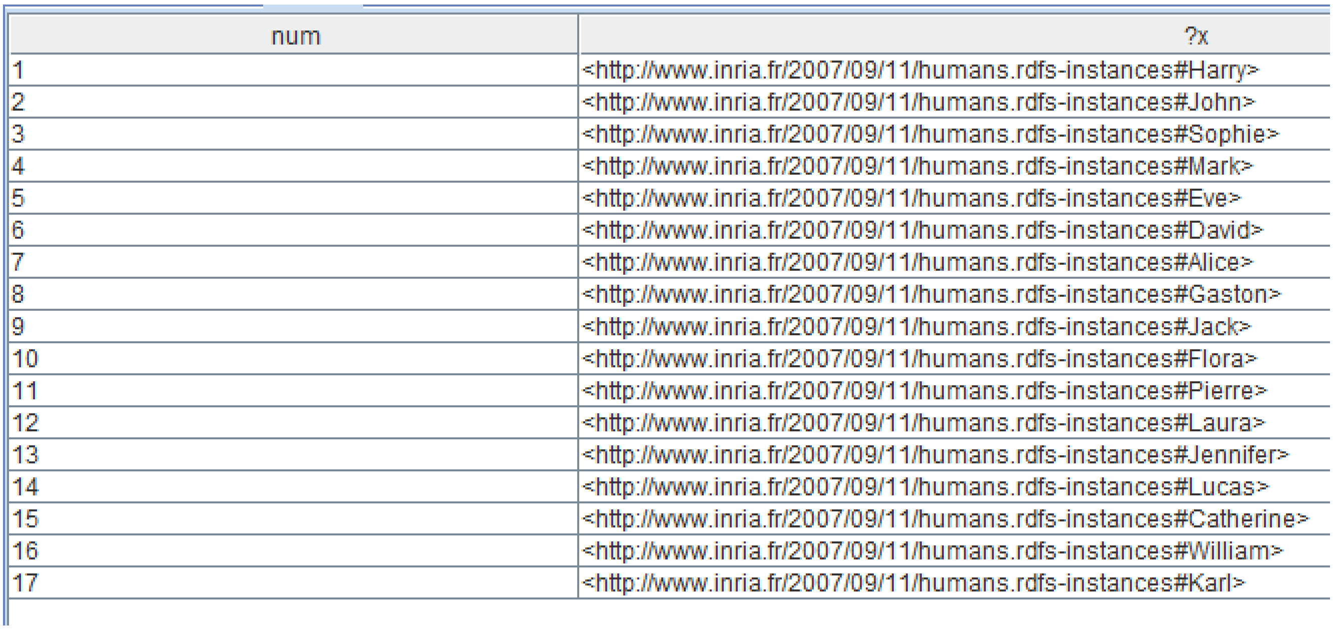
Number of results:



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>

3. Load the schema (.rdfs)
4. Rerun the query to find the Persons and explain the result.

New number of results and your explanation:



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
13	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer>
14	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
15	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
16	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>
17	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>

The RDFS schema is adding resources due to the definition of subclasses of Person

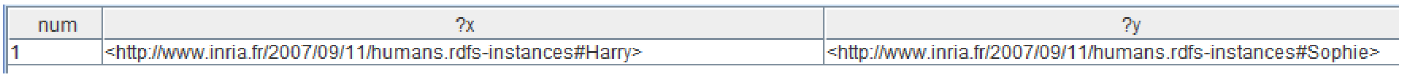
Question 2

1. Write a query to find Males and their wives. How many answers do you get? Explain this result.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a h:Male ; h:hasSpouse ?y }
```

Number of results:



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>

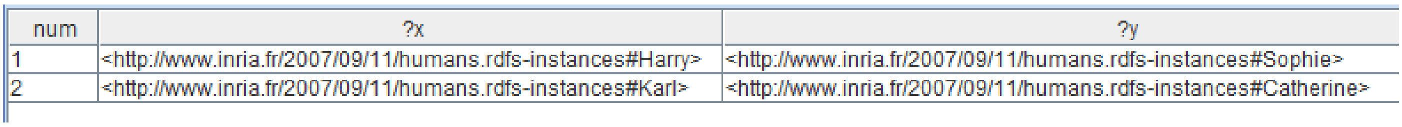
We are missing all the Person that are not Male or a subclass of Male.

2. In the data declare that Lucas has to father Karl. Reset Corese, reload the ontology and the data, and then rerun the query to find Males and their wives. Explain the new result.

Line added in RDF:

```
<hasFather rdf:resource="#Karl"/>
```

Number of results and your explanation:



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>

As Karl is now defined as a Father it automatically becomes a Male due to range property of the hasFather property, and now matches the request.

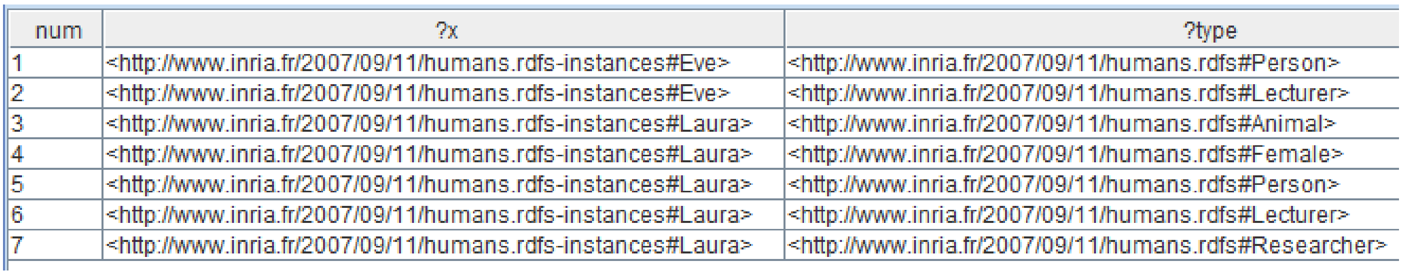
Question 3

1. Write a query to find the Lecturers and their types. How many answers do you get? See how this typing is declared in the data and explain the result.

Query:

```
prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a h:Lecturer ; rdf:type ?type }
```

Number of results and your explanation:



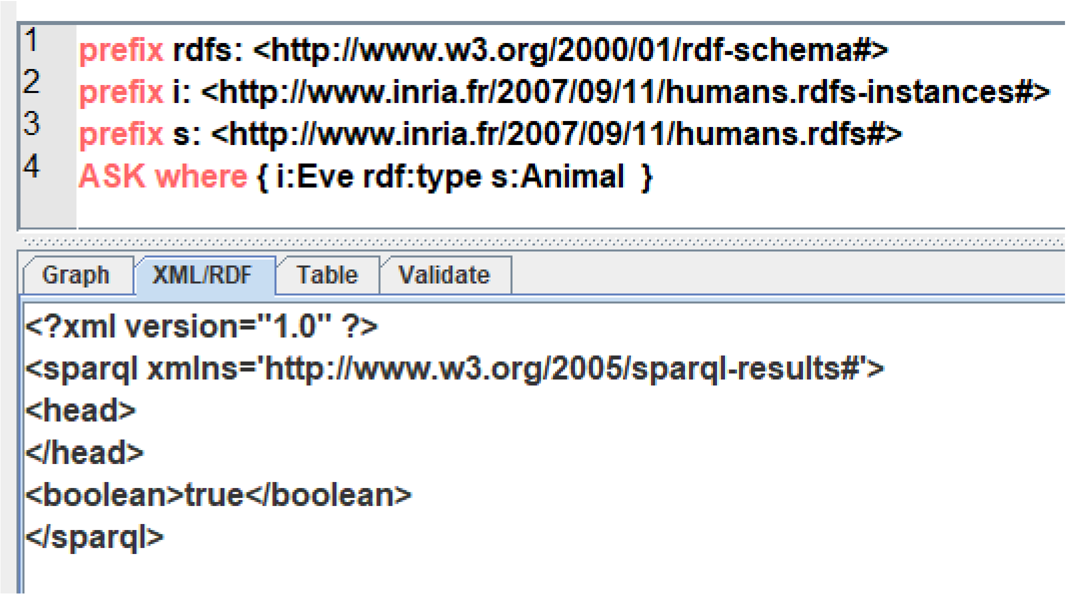
num	?x	?type
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#Animal>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#Female>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#Person>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#Researcher>

This is because of the Lecturer class hierarchy that is defined in the schema.

However, in the case of Eve, it is unclear to me why Animal is not listed, as Person is simply defined as subclass of Animal.

The fact that Eve is of type Animal can be confirmed by the query below :

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
prefix i: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>
ASK where { i:Eve rdf:type s:Animal }
```



```
prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
prefix i: <http://www.inria.fr/2007/09/11/humans.rdfs-instances#>
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
ASK where { i:Eve rdf:type s:Animal  }
```

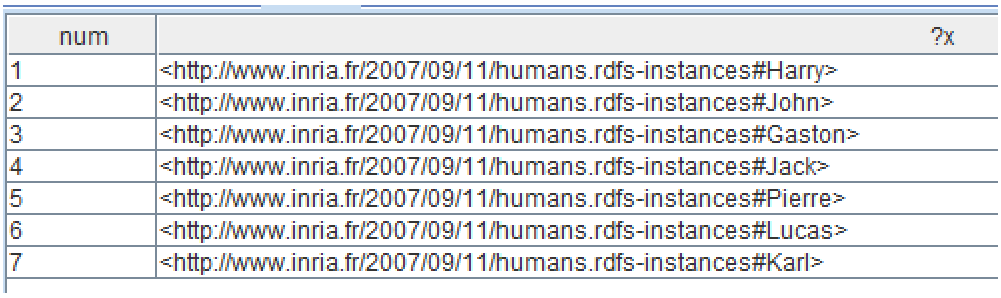
```
<?xml version="1.0" ?>
<sparql xmlns='http://www.w3.org/2005/sparql-results#'>
<head>
</head>
<boolean>true</boolean>
</sparql>
```

2. Write a query to find common instances of the classes Person and Male. See how this typing is declared in the data and explain the presence of Jack.

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x rdf:type s:Person ; rdf:type s:Male }
```

Your explanation of the result:



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>

Jack is present because it is declared as Man that is a subclass of both Person and Male

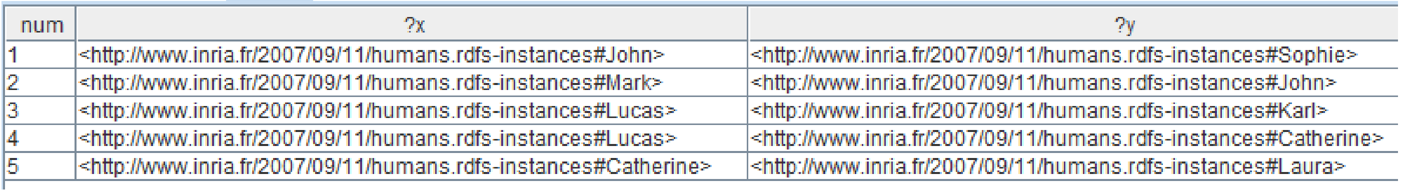
Question 4

Write a query to find the hasAncestor relations. Explain the result after checking where this property is used in the data.

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x s:hasAncestor ?y }
```

Your explanation of the result:



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>

hasAncestor is not used in the data.

However, in the schema the properties hasFather, hasMother and hasParent are defined as subproperty of has Ancestor, which explains the results found.

Question 5

1. Write a query to find the family cores (couples and their children) using a SELECT

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?parent1 ?parent2 ?child where
{
  { ?parent1 s:hasSpouse ?parent2
    {
      { ?child s:hasParent ?parent1 }
      UNION
      { ?parent1 s:hasChild ?child }
    }
    UNION
    {
      { ?child s:hasParent ?parent2 }
      UNION
      { ?parent2 s:hasChild ?child }
    }
  }
}
order by ?parent1
```

2. Modify it to display the result with a CONSTRUCT query

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
CONSTRUCT {
  ?parent1 s:family ?parent2 .
  ?parent2 s:family ?parent1 .
  ?parent1 s:family ?child .
  ?parent2 s:family ?child .
  ?child s:family ?parent1 .
  ?child s:family ?parent2 .
  }
select ?parent1 ?parent2 ?child where
{
  { ?parent1 s:hasSpouse ?parent2
    {
      { ?child s:hasParent ?parent1 }
      UNION
      { ?parent1 s:hasChild ?child }
    }
    UNION
    {
      { ?child s:hasParent ?parent2 }
      UNION
      { ?parent2 s:hasChild ?child }
    }
  }
}
order by ?parent1
```

Question 6

1. Declare the olderThan relationship in the schema to indicate between two people which is eldest and construct the arcs between peoples with a SPARQL query

Addition to schema:

```
<rdf:Property rdf:ID="olderThan">
  <domain rdf:resource="#Animal"/>
  <range rdf:resource="#Animal"/>
</rdf:Property>
```

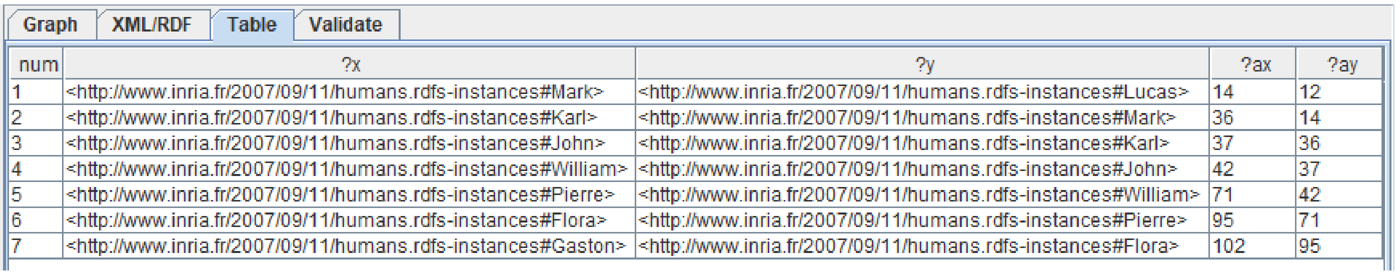
Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
insert { ?x s:olderThan ?y }
where {
  ?x s:age ?ax .
  ?y s:age ?ay .
  filter(?ax > ?ay)
}
```

2. Find a query that generates only the minimum number of link without redundancy with olderThan transitivity.

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where {
  { ?x s:olderThan ?y . ?x s:age ?ax . ?y s:age ?ay }
  minus
  { ?x s:olderThan ?z ?z s:olderThan ?y}
}
order by ?ax ?ay
```



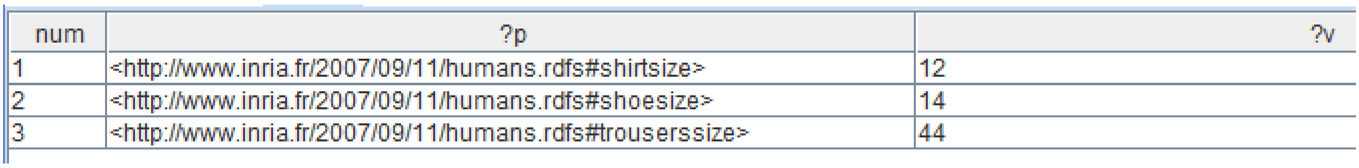
num	?x	?y	?ax	?ay
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	14	12
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	36	14
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	37	36
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	42	37
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>	71	42
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	95	71
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>	102	95

Question 7

Write a query to find for John the properties which label contains the string "size" and the value of these properties.

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select distinct ?p ?v where {
  ?x s:name "John" .
  ?x ?p ?v .
  ?p rdfs:label ?l .
  filter (contains(?l, "size"))
}
```



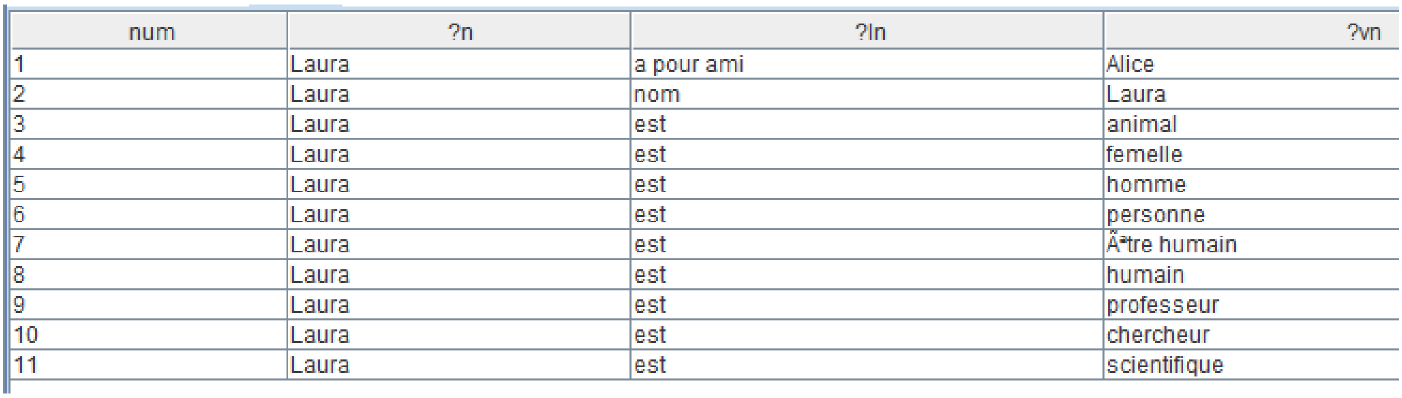
num	?p	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize>	12
2	<http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>	14
3	<http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize>	44

Question 8

Use the ontology to document your answers in natural language: write a query to find the types and properties of Laura in French.

Query:

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select distinct ?n ?ln ?vn where {
  ?x s:name "Laura" .
  ?x s:name ?n .
  {
    ?x ?p ?v .
    ?p rdfs:label ?l .
    filter (lang(?l) = "fr")
    bind(str(?l) as ?ln)
    optional { ?v s:name ?z }
    bind( if (bound(?z), ?z, ?v) as ?vn)
  }
  union
  {
    ?x rdf:type ?t .
    ?t rdfs:label ?l .
    filter (lang(?l) = "fr")
    bind("est" as ?ln)
    bind(str(?l) as ?vn)
  }
}
```



num	?n	?ln	?vn
1	Laura	a pour ami	Alice
2	Laura	nom	Laura
3	Laura	est	animal
4	Laura	est	femelle
5	Laura	est	homme
6	Laura	est	personne
7	Laura	est	Ãªtre humain
8	Laura	est	humain
9	Laura	est	professeur
10	Laura	est	chercheur
11	Laura	est	scientifique

Unfortunately, Laura is qualified as “homme” because of :

```
<Class rdf:ID="Person">
  …
  <label xml:lang="fr">homme</label>
  …
</Class>
```

Day 04: questions from the course on OWL.

Q5.1 What can we deduce?

```
ex:Man owl:intersectionOf (ex:Male ex:Human) .
ex:Woman owl:intersectionOf (ex:Female ex:Human) .
ex:Human owl:unionOf (ex:Man ex:Woman) .
ex:Jane a ex:Human .
ex:John a ex:Man .
ex:James a ex:Male .
ex:Jane a ex:Female .
```

```
Jane a Woman .
John a Male .
John a Human .
```

Q5.2 What are we defining and inferring?

```
@prefix ex: <http://example.org/>

ex:GrandFather rdfs:subClassOf [
   a owl:Class ;
   owl:intersectionOf ( ex:Parent ex:Man )
] .

ex:Jim a ex:Man, ex:Parent .
ex:Jack a ex:GrandFather .
```

```
ex:Jack a ex:Man, ex:Parent .
```

Q5.3 What can we deduce?

```
ex:hasSpouse a owl:SymmetricProperty .
ex:hasChild  owl:inverseOf ex:hasParent .
ex:hasParent rdfs:subPropertyOf ex:hasAncestor .
ex:hasAncestor a owl:TransitiveProperty .
ex:Jim  ex:hasChild  ex:Jane .
ex:Jane ex:hasSpouse ex:John .
ex:Jim  ex:hasParent ex:James .
```

```
ex:John ex:hasSpouse ex:Jane .
Ex:Jane ex:hasParent ex:Jim .
Ex:James ex:hasChild ex:Jim .
Ex:Jane ex:ex:hasAncestor ex:Jim .
Ex:Jane ex:ex:hasAncestor ex:James .
Ex:Jim  ex:ex:hasAncestor ex:James .
```

Q5.4 What can we deduce?

```
ex:Human owl:equivalentClass foaf:Person .
foaf:name owl:equivalentProperty ex:name .
ex:JimmyPage a ex:Human ;
             owl:sameAs ex:JamesPatrickPage .
ex:JimmyHendrix owl:differentFrom ex:JimmyPage .
```

```
Ex:JimmyPage a foaf:Person.
Ex:JamesPatrickPage a foaf:Person ; a ex:Person .
Ex:JimmyHendrix owl:differentFrom ex:JamesPatrickPage .
```

Q5.5 What are we defining and inferring?

```
ex:UnluckyPerson owl:equivalentClass [
  a owl:Class ;
  owl:intersectionOf (
    ex:Person
    [ a owl:Class ; owl:complementOf ex:Lucky ]
  )
] .
```

An ex:UnluckyPerson is an ex:Person that is not ex:Lucky

Q5.6 What can we deduce?

```
ex:Human rdfs:subClassOf
  [ a owl:Restriction ;
    owl:onProperty ex:hasParent ;
    owl:allValuesFrom ex:Human ] .
ex:John a ex:Human .
ex:John ex:hasParent ex:James, ex:Jane.
```

The rdfs:subClassOf says that the parents of all humans are also humans

```
Ex:James a ex:Human .
Ex:Jane a ex:Human .
```

Q5.7 What are we defining and inferring?

```
@prefix ex: <http://example.org/>
ex:PersonList rdfs:subClassOf
  [
    a owl:Restriction ;
    owl:onProperty rdf:first ;
    owl:allValuesFrom ex:Person
  ] ,  [
    a owl:Restriction ;
    owl:onProperty rdf:rest ;
    owl:allValuesFrom ex:PersonList
  ] .

ex:value rdfs:range ex:PersonList .
ex:abc ex:value (ex:a ex:b ex:c) .
```

PersonList is defined to be a list of Person (recursively) and ex:abc is a ex:PersonList (through rdfs:range

```
ex:a a ex:Person .
ex:b a ex:Person .
ex:c a ex:Person .
```

Q5.8 What are we defining and inferring?

```
@prefix ex: <http://example.org/>
ex:Human rdfs:subClassOf [
  owl:intersectionOf (
    [
      a owl:Restriction ;
        owl:onProperty ex:hasBiologicalFather ;
        owl:maxCardinality 1
    ] , [
      a owl:Restriction ;
        owl:onProperty ex:hasBiologicalMother ;
        owl:maxCardinality 1
    ] )
] .
ex:John a ex:Human ;
        ex:hasBiologicalFather ex:James , ex:Jimmy .
```

The code above causes an error if the engine can infer that ex:James and ex:Jimmy are different resources. This is due the maxCardinality of 1 of ex:hasBiologicalFather.

Q5.9 What are we defining and inferring?

```
@prefix ex: <http://example.org/>
ex:Wealthy a owl:Class ;
     owl:equivalentClass  [
       a owl:Class ; owl:intersectionOf  (
         [ a owl:Restriction ;
           owl:onProperty ex:hasChild ;
           owl:allValuesFrom ex:Wealthy
         ] ,
         [ a owl:Restriction ;
           owl:onProperty ex:hasChild ;
           owl:someValuesFrom ex:Wealthy
         ]
     )] .
ex:John a ex:Wealthy ; ex:hasChild ex:Jim .
```

The first restriction says there all children become wealthy if one of their parent is wealthy.
The second restriction says their parents becomes wealthy if there exists at least one child that is wealthy.
As a consequence all others siblings will also become wealthy.

```
ex:Jim a ex:Wealthy
```

Day 04: Answers to the practical session on OWL.

Software requirements

- The RDF XML online validation service by W3C: https://www.w3.org/RDF/Validator/
- The RDF online translator: http://rdf-translator.appspot.com/
- The SPARQL Corese engine: http://wimmics.inria.fr/corese

A, Query data augmented by an OWL schema

Make a copy of the human.rdfs file, name it humans.owl and use it for the rest of the session. For each of the following statements, specify a SPARQL query that shows that the difference before and after running the OWL inferences: you will find that answers to these queries are different depending on whether you load the ontology humans.rdfs or the humans.owl you modified.

1. Declare that `hasSpouse` and `hasFriend` are symmetrical properties.

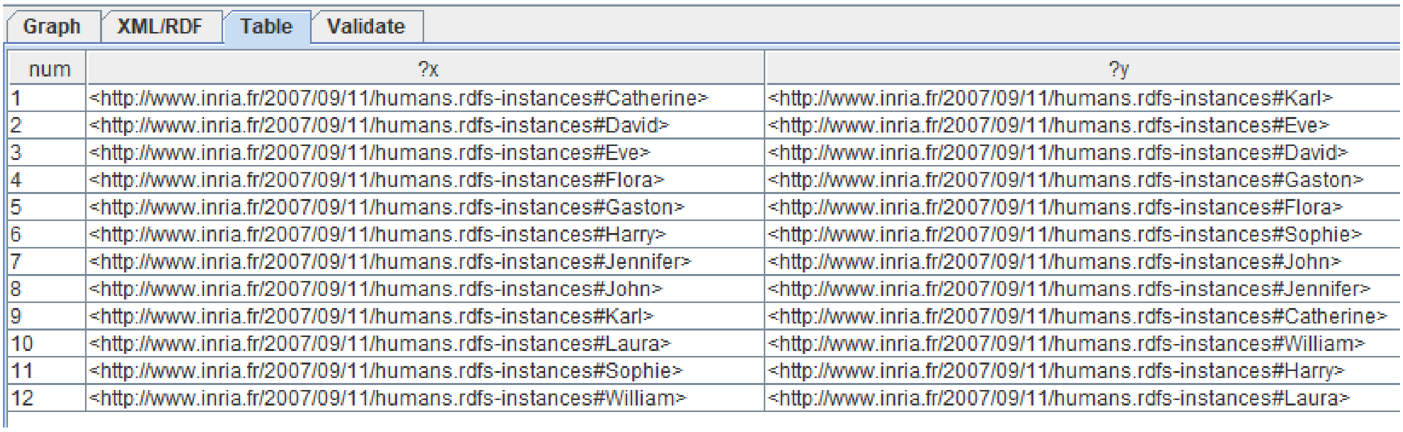
Code added to the schema:

Global change made applicable to tbe entire section :

- **In human.owl, added to root RDF tag : `xmlns:owl="http://www.w3.org/2002/07/owl#"`**
- **in human.owl, changed all rdfs:Class to owl:class**
- **in Corese, activate the engine option "OWL RL (to get owl type reasoning).**

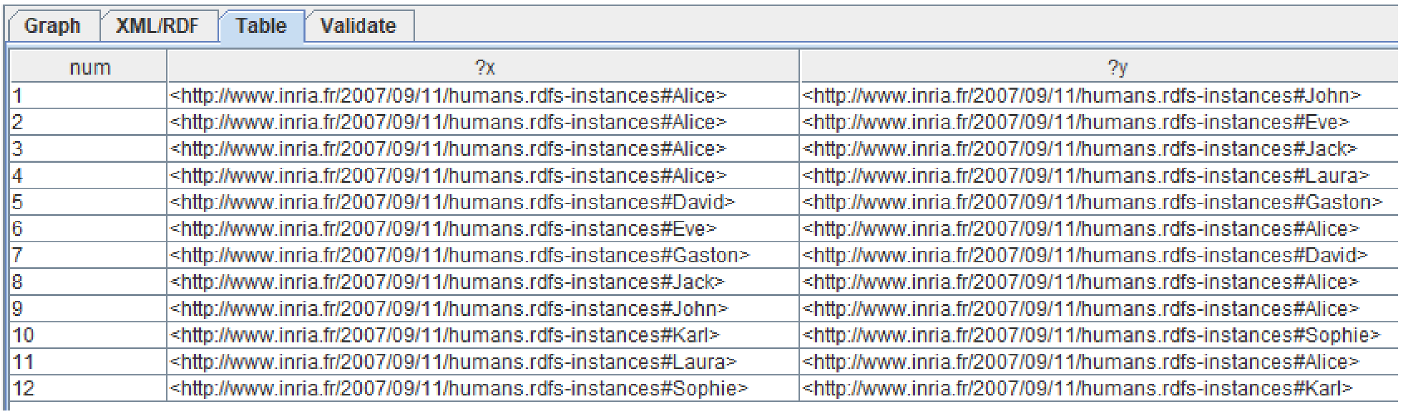
```
<owl:SymmetricProperty rdf:ID="hasSpouse" />
<owl:SymmetricProperty rdf:ID="hasFriend" />
```

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x s:hasSpouse ?y } order by ?x
```



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#William>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x s:hasFriend ?y } order by ?x
```



num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>

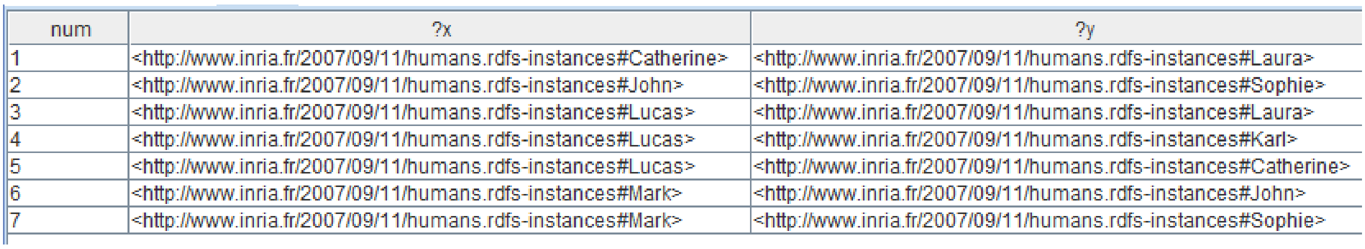
The symmetry of the property can now be observed

2. Declare `hasAncestor` as transitive property.

Code added to the schema:

```
<owl:TransitiveProperty rdf:ID="hasAncestor" />
```

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x s:hasAncestor ?y } order by ?x
```



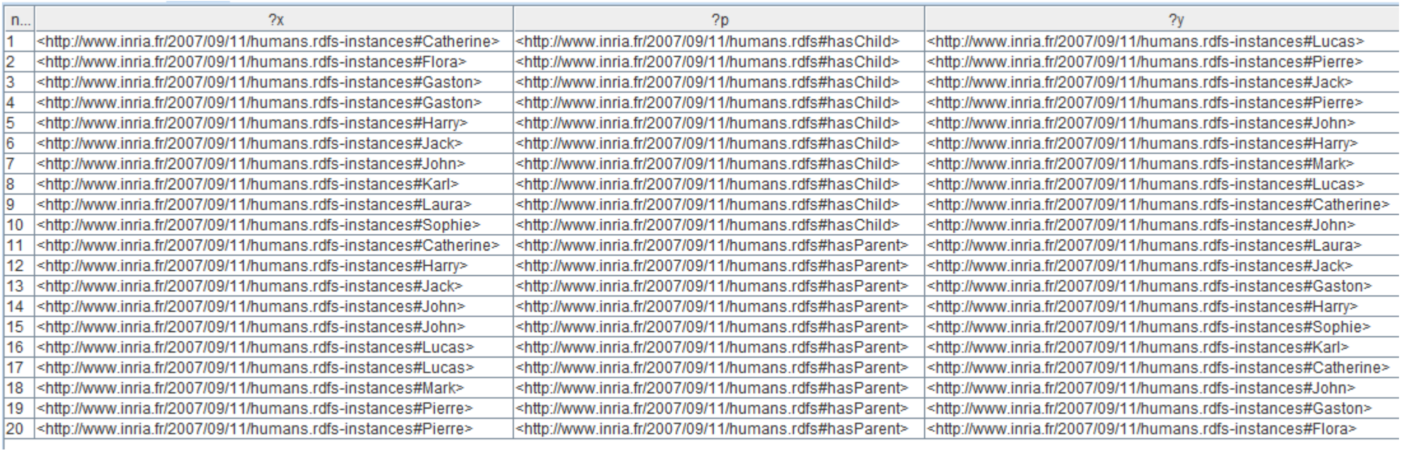
num	?x	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>

3. Declare that `hasChild` is the inverse `hasParent` property.

Code added to the schema:

```
<rdf:Property rdf:ID="hasChild">
  <owl:inverseOf rdf:resource="#hasParent" />
</rdf:Property>
```

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where {
  ?x ?p ?y .
  filter(?p = s:hasChild || ?p = s:hasParent)
} order by ?p ?x
```



n...	?x	?p	?y
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>
5	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
6	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
7	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>
8	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>
9	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
10	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasChild>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
11	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>
12	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>
13	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
14	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
15	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie>
16	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl>
17	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine>
18	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#John>
19	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
20	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre>	<http://www.inria.fr/2007/09/11/humans.rdfs#hasParent>	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora>

It is worth noted that the addition “hasChild inverseOf hasParent” allows the engine to discover ancestors to children indicated only through the hasChild relationship : the previous request about hasAncestor now returns 12 answers instead of 7.

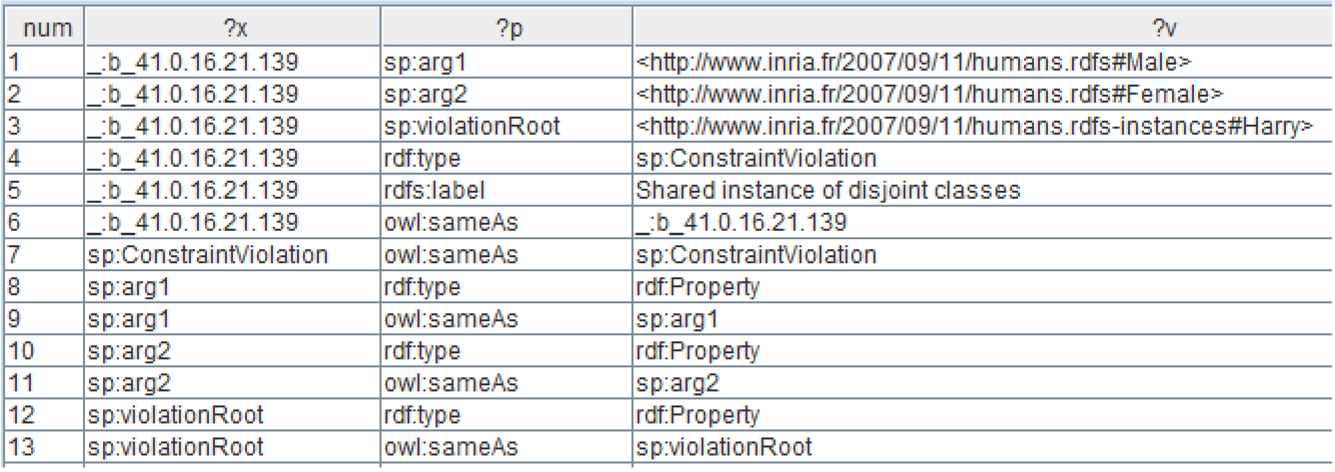
4. Declare the disconnection between `Male` and `Female`. Violate the constraint in the data, check the results and then remove the violation you created.

Code added to the schema:

```
<owl:Class rdf:ID="Male">
<owl:disjointWith rdf:resource="#Female"/>
</owl:Class>
```

Harry is declared as a Man in human.rds, when adding `<Woman rdf:ID="Harry"/>`
The request below returns :

```
select * where { ?x ?p ?v } order by ?x ?p
```



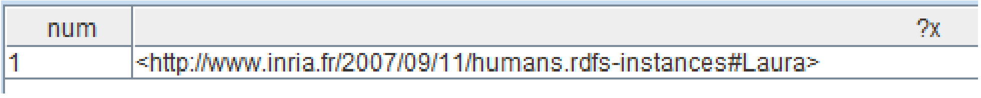
num	?x	?p	?v
1	_:b_41.0.16.21.139	sp:arg1	<http://www.inria.fr/2007/09/11/humans.rdfs#Male>
2	_:b_41.0.16.21.139	sp:arg2	<http://www.inria.fr/2007/09/11/humans.rdfs#Female>
3	_:b_41.0.16.21.139	sp:violationRoot	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry>
4	_:b_41.0.16.21.139	rdf:type	sp:ConstraintViolation
5	_:b_41.0.16.21.139	rdfs:label	Shared instance of disjoint classes
6	_:b_41.0.16.21.139	owl:sameAs	_:b_41.0.16.21.139
7	sp:ConstraintViolation	owl:sameAs	sp:ConstraintViolation
8	sp:arg1	rdf:type	rdf:Property
9	sp:arg1	owl:sameAs	sp:arg1
10	sp:arg2	rdf:type	rdf:Property
11	sp:arg2	owl:sameAs	sp:arg2
12	sp:violationRoot	rdf:type	rdf:Property
13	sp:violationRoot	owl:sameAs	sp:violationRoot

5. Declare that the class `Professor` is the intersection of the class `Lecturer` and `Researcher` class.

Code added to the schema:

```
<owl:Class rdf:ID="Professor">
  <owl:intersectionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Lecturer"/>
    <owl:Class rdf:about="#Researcher"/>
  </owl:intersectionOf>
</owl:Class>
```

```
prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a s:Professor . }
```

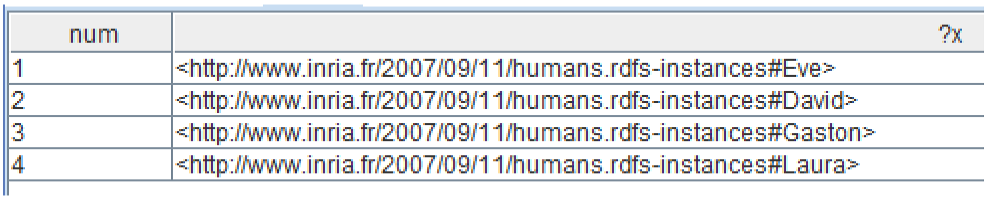


num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>

6. Declare that the `Academic` class is the union of classes `Lecturer` and `Researcher`.

Code added to the schema:

```
<owl:Class rdf:ID="Academic">
  <owl:unionOf rdf:parseType="Collection">
    <owl:Class rdf:about="#Lecturer"/>
    <owl:Class rdf:about="#Researcher"/>
  </owl:unionOf>
</owl:Class>

prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select * where { ?x a s:Academic . }
```



num	?x
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#David>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston>
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura>

7. Create a class `Organization` and its sub class `University`. Create a new property `mainEmployer`, with domain `Person` and range `Organization`. Use a restriction to declare that any `Professor` has for main employer a `University`.

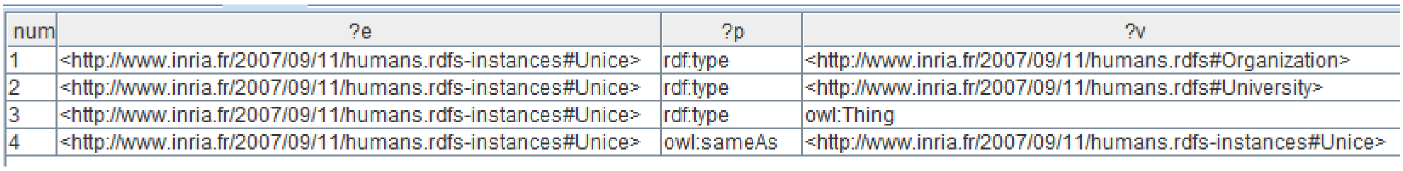
Code added to the schema (new property, new classes and new restriction):

```
<owl:Class rdf:ID="Professor">
  <subClassOf>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#mainEmployer"/>
      <owl:allValuesFrom rdf:resource="#University" />
    </owl:Restriction>
  </subClassOf>
</owl:Class>
```

Code added to the data (just declare the main employer of a Professor):

From previous questions, we know that Laura is a Professor. So we add the line in **bold green**:

```
<Researcher rdf:about="#Laura">
  <name>Laura</name>
  <mainEmployer rdf:resource="#Unice"/>
</Researcher>

prefix s: <http://www.inria.fr/2007/09/11/humans.rdfs#>
select ?e ?p ?v  where { ?x a s:Professor ; s:mainEmployer ?e . ?e ?p ?v  }
```



num	?e	?p	?v
1	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Unice>	rdf:type	<http://www.inria.fr/2007/09/11/humans.rdfs#Organization>
2	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Unice>	rdf:type	<http://www.inria.fr/2007/09/11/humans.rdfs#University>
3	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Unice>	rdf:type	owl:Thing
4	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Unice>	owl:sameAs	<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Unice>

8. Use a restriction to declare that any person must have a parent who is a woman. For this last statement, you need to run the rule engine after loading the ontology and data.

Code added to the schema:

```
<owl:Class rdf:ID="Person">
  <owl:equivalentClass>
    <owl:Restriction>
      <owl:onProperty rdf:resource="#hasParent"/>
      <owl:someValuesFrom rdf:resource="#Woman" />
    </owl:Restriction>
  </owl:equivalentClass>
</owl:Class>
```

B, Make your own OWL models:

For each one of the following OWL primitives imagine a definition that could use it and provide that definition in OWL using your preferred syntax (RDF/XML or N3/Turtle). For instance a possible definition using owl:TransitiveProperty would be a definition of the Ancestor property. For each primitive in the following list you imagine the definition of a class or property that was not given in the course and you give that definition in English and in OWL.

1. owl:one of <YOUR EXAMPLE HERE/>
2. owl:unionOf <YOUR EXAMPLE HERE/>
3. owl:intersectionOf <YOUR EXAMPLE HERE/>
4. owl:complementOf <YOUR EXAMPLE HERE/>
5. owl:disjointWith <YOUR EXAMPLE HERE/>
   or owl:AllDisjointClasses
   or owl:disjointUnionOf
6. owl:ObjectProperty <YOUR EXAMPLE HERE/>
7. owl:DatatypeProperty <YOUR EXAMPLE HERE/>
8. owl:SymmetricProperty <YOUR EXAMPLE HERE/>
   or owl:AsymmetricProperty
9. owl:inverseOf <YOUR EXAMPLE HERE/>
10. owl:TransitiveProperty <YOUR EXAMPLE HERE/>
11. owl:propertyDisjointWith <YOUR EXAMPLE HERE/>
12. owl:ReflexiveProperty <YOUR EXAMPLE HERE/>
    or owl:IrreflexiveProperty
13. owl:propertyChainAxiom <YOUR EXAMPLE HERE/>
14. owl:FunctionalProperty <YOUR EXAMPLE HERE/>
15. owl:InverseFunctionalProperty <YOUR EXAMPLE HERE/>
16. owl:hasKey <YOUR EXAMPLE HERE/>
17. owl:allValuesFrom <YOUR EXAMPLE HERE/>
18. owl:someValuesFrom <YOUR EXAMPLE HERE/>
19. owl:hasValue <YOUR EXAMPLE HERE/>
20. owl:maxCardinality <YOUR EXAMPLE HERE/>
    or owl:minCardinality
21. owl:qualifiedCardinality <YOUR EXAMPLE HERE/>

Day 05: questions from the course on Vocabularies.

Q6.1 What do you think of the annotation?

```
@prefix skos: <http://www.w3.org/2004/02/skos/core#>.
<#B-A-Ba> a skos:Concept ;
  skos:prefLabel   "B.A.-BA"@en , "b.a.-ba"@en ;
  skos:altLabel    "B-A-BA"@en ,  "b-a-ba"@en ;
  skos:hiddenLabel "BABA"@en , "baba"@en .
```

There is an error in this annotation as there are 2 preferred labels in English.

Q6.2 practice:

1. Using the site prefix.cc find back the namespace usually associated to the SKOS prefix
2. Access the URL of the namespace and find the RDF source file defining the SKOS vocabulary
3. Find the definition of the property `narrowMatch` and give all the relations it has with other properties

1) http://www.w3.org/2004/02/skos/core#

2) http://www.w3.org/TR/skos-reference/skos.rdf

3) `narrowMatch` is related to the following properties:

```
rdfs:subPropertyOf:   skos:mappingRelation, skos:narrower
owl:inverseOf:        skos:broadMatch
```

Q6.3 practice:

1. Open the source file of Dublin Core Terms: http://dublincore.org/2012/06/14/dcterms.rdf
   Look at the definition of the class `FileFormat` and find the class it inherits from.
2. Choose your preferred book on Amazon, Fnac, etc. and describe it in an RDF annotation using as many DC primitives as necessary .
3. Add the most restrictive CC license to your preferred book ; is this license appropriate?

1) `<rdfs:subClassOf rdf:resource="http://purl.org/dc/terms/MediaType"/>`

2)

```
<http://www.isbn.fr/ISBN-13/978-2100496358>
dc:creator <http://ns.inria.fr/fabien.gandon#me> ,
<http://ns.inria.fr/rose.dieng#me> ,
<http://ns.inria.fr/alain.giboin#me> ,
<http://ns.inria.fr/olivier.corby#me> ;
dc:title "Knowledge management : Méthodes et outils pour la gestion des
connaissances" ;
dc:language "fr" ;
dc:subject "RDF, RDFS, SPARQL, OWL, SKOS" ;
dc:date "2005-10-05" ;
dc:publisher <http://dunod.fr> ;
dc:format "text/html" ; dc:type dcterms:Text ;
dc:license <https://creativecommons.org/licenses/by-nc-nd/4.0> .
```

3) Licence added about in **green bold**.
   That license does not allow commercial use of the book nor to share adaptation of the book.

Q6.4 practice:

1. Get the source of the FoaF schema: http://xmlns.com/foaf/spec/index.rdf
2. Find the property `weblog`
3. What are the types of this property?
4. Does it inherit from other properties?
5. What is its signature?

2)

```
  <rdf:Property rdf:about="http://xmlns.com/foaf/0.1/weblog"
vs:term_status="stable" rdfs:label="weblog" rdfs:comment="A weblog of some
thing (whether person, group, company etc.).">
    <rdf:type rdf:resource="http://www.w3.org/2002/07/owl#ObjectProperty"/>
    <rdfs:subPropertyOf rdf:resource="http://xmlns.com/foaf/0.1/page"/>
    <rdf:type
rdf:resource="http://www.w3.org/2002/07/owl#InverseFunctionalProperty"/>
    <rdfs:domain rdf:resource="http://xmlns.com/foaf/0.1/Agent"/>
    <rdfs:range rdf:resource="http://xmlns.com/foaf/0.1/Document"/>
    <rdfs:isDefinedBy rdf:resource="http://xmlns.com/foaf/0.1/"/>
  </rdf:Property>
```

3) The type is `http://www.w3.org/2002/07/owl#InverseFunctionalProperty`
4) It inherits from `http://xmlns.com/foaf/0.1/page`
5) Its signature is :
   - Domain : `http://xmlns.com/foaf/0.1/Agent`
   - Range : `http://xmlns.com/foaf/0.1/Document`

Q6.5 practice:

1. Find the FOAF-a-Matic web page
2. Use this tool to generate your FOAF profile in RDF/XML
3. Translate it into Turtle, save and give the result in your answers.
4. Add five specific relationships to your FOAF file using RELATIONSHIPS: http://purl.org/vocab/relationship/

1) http://www.ldodds.com/foaf/foaf-a-matic.html
2)
3) http://rdf-translator.appspot.com/ :

```
@prefix admin: <http://webns.net/mvcb/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix vocab:<http://purl.org/vocab/relationship/> .

<> a foaf:PersonalProfileDocument ;
    admin:errorReportsTo <mailto:leigh@ldodds.com> ;
    admin:generatorAgent <http://www.ldodds.com/foaf/foaf-a-matic> ;
    foaf:maker <#me> ;
    foaf:primaryTopic <#me> .

<#me> a foaf:Person ;
    foaf:family_name "Tholon" ;
    foaf:givenname "Alain" ;
    foaf:knows [ a foaf:Person ;
          foaf:mbox_sha1sum "d2f304c4fda0038ee688d45b8fb5a700cc734c2c" ;
          foaf:name "Seb" ] ;
    foaf:mbox_sha1sum "0c192e6a3e710c36253e2e2932cef0b79ec7099d" ;
    foaf:name "Alain Tholon" ;
    foaf:schoolHomepage <https://www.datasciencetech.institute/> ;
    foaf:title "Mr" ;
    vocab:Relationship <http://ns.inria.fr/fabien.gandon#me> .
```

4) Added the 2 lines in **bold green** above.

Q6.6 What does this mean?

```
:BioRDF2DBLP a void:Linkset;
               void:target :BioRDF;
               void:target :DBLP;
               void:linkPredicate skos:exactMatch;
               void:triples 8936 .
```

BioRDF2DBLP is a linkeset between the dataset bioRDF DBLP.
It contains 8936 RDF triples that are of kind skos:exactMatch.

Q6.7 practice:

1. Connect to the Void Store SPARQL endpoint: http://void.rkbexplorer.com/sparql/
2. What is the meaning of the default SPARQL query in the interface, run it and look at the results.
3. Write a SPARQL query to find the dataset that has for label "DBpedia-fr" and all its properties.

1) http://void.rkbexplorer.com/sparql/

2)
```
SELECT DISTINCT ?endpoint
  WHERE { ?ds a void:Dataset . ?ds void:sparqlEndpoint ?endpoint }
```

This request retrieves all sparql endpoints associated to all dataset known by the engine.
It returns 118 sparql endpoints

3)
```
SELECT ?p ?v
WHERE { ?ds a void:Dataset . ?ds rdfs:label "DBpedia-fr" ?ds ?p ?v }
```

P	v
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	_:b1
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	_:b2
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/ns/dcat#Dataset>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://rdfs.org/ns/void#Dataset>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/ns/sparql-service-description#Dataset>
<http://www.w3.org/1999/02/22-rdf-syntax-ns#type>	<http://www.w3.org/ns/prov#Entity>
<http://purl.org/dc/terms/title>	DBpedia French
<http://www.w3.org/2000/01/rdf-schema#label>	**DBpedia-fr**
<http://purl.org/dc/terms/description>	DBpedia is a crowd-sourced … (snip) … itself.
<http://www.w3.org/ns/dcat#keyword>	lod
<http://www.w3.org/ns/dcat#keyword>	crossdomain
<http://www.w3.org/ns/dcat#keyword>	rdf
<http://www.w3.org/ns/dcat#keyword>	access-api
<http://www.w3.org/ns/dcat#keyword>	access-bulk
<http://www.w3.org/ns/dcat#keyword>	access-www
<http://www.w3.org/ns/dcat#keyword>	wikipedia
<http://purl.org/dc/terms/publisher>	<http://wiki.dbpedia.org/Association>
<http://rdfs.org/ns/void#exampleResource>	<http://fr.dbpedia.org/data/Berlin.rdf>
<http://purl.org/dc/terms/language>	fr
<http://www.w3.org/ns/dcat#distribution>	<http://fr.dbpedia.org/sparql>
<http://rdfs.org/ns/void#sparqlEndpoint>	<http://fr.dbpedia.org/sparql>
<http://www.w3.org/ns/dcat#landingPage>	<http://fr.dbpedia.org/>
<http://www.w3.org/ns/prov#wasDerivedFrom>	<http://dumps.wikimedia.org/frwiki/20130420>
<http://www.w3.org/ns/prov#wasGeneratedBy>	<http://dbpedia.org/dataid.ttl#Extraction>
<http://purl.org/dc/terms/license>	<http://creativecommons.org/licenses/by-sa/4.0/rdf>
<http://purl.org/dc/terms/rights>	Data comprising DBpedia release 3.4 … (snip) … License.
<http://dataid.dbpedia.org/ns#licenseName>	Creative Commons Attribution-ShareAlike 4.0
<http://dataid.dbpedia.org/ns#ontologyLocation>	<http://downloads.dbpedia.org/3.9/dbpedia_3.9.owl>
<http://www.w3.org/ns/dcat#contactPoint>	<http://dbpedia.org/dataid.ttl#Sylvain_Boissel>
<http://www.w3.org/ns/dcat#contactPoint>	<http://dbpedia.org/dataid.ttl#_Julien_Cojan>
<http://dataid.dbpedia.org/ns#latestVersion>	<http://dbpedia.org/dataid.ttl#DBpedia_fr_3.9>

Q6.8 What does this mean?

```
ex:plot prov:used ex:stats1998 .
ex:bar-chart prov:wasGeneratedBy ex:plot .
ex:stats1998 a dcat:Distribution ;
             dcat:format [ rdfs:label "CSV" ] ;
             dcat:mediaType "text/csv" .
```

A bar-chart was generated by plot using stats1998 that is a CSV file.

Q6.9 What does this mean?

```
@prefix dcat: <http://www.w3.org/ns/dcat#> .
@prefix void: <http://rdfs.org/ns/void#> .
@prefix rdf:  <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix prov: <http://www.w3.org/ns/prov#> .
@prefix dct:  <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@base <http://inria.fr/data> .

:db-employ
  a dcat:Distribution ;
  dcat:downloadURL <http://wimmics.inria.fr/docs/employ-2014.sql> ;
  dct:title "SQL Dump of the employees" ;
  dct:spatial <http://www.geonames.org/6640252> ;
  dct:issued "2015-01-12"^^xsd:date ;
  dct:temporal <http://reference.data.gov.uk/id/year/2014> ;
  dct:publisher <http://inria.fr> ;
  dcat:mediaType "application/sql" ;
  dcat:format [ rdfs:label "SQL" ] ;
  dct:language <http://id.loc.gov/vocabulary/iso639-1/fr> ;
  dcat:byteSize "38729"^^xsd:decimal .

:R2RTransform12 prov:used :db-employ ;
                prov:used :R2R-employ-mapping ;
                prov:used <http://xmlns.com/foaf/0.1/> .

:FoaFDump a void:Dataset;
          void:feature <http://www.w3.org/ns/formats/RDF_XML>;
          void:dataDump <http://wimmics.inria.fr/docs/employ-2014.rdf>;
          void:exampleResource <http://ns.inria.fr/fabien.gandon#me> ;
          void:vocabulary <http://xmlns.com/foaf/0.1/>;
          void:triples 12875;
          dct:title "RDF Dump of the employees" ;
          prov:wasGeneratedBy :R2RTransform12 ;
          prov:generatedAtTime "2015-01-14T11:38:27"^^xsd:dateTime ;
          prov:wasDerivedFrom :db-employ .
```

SQL dump of the INRIA employees done on January 12th, 2015.

The SQL dump was transformed in RDS using FOAF.

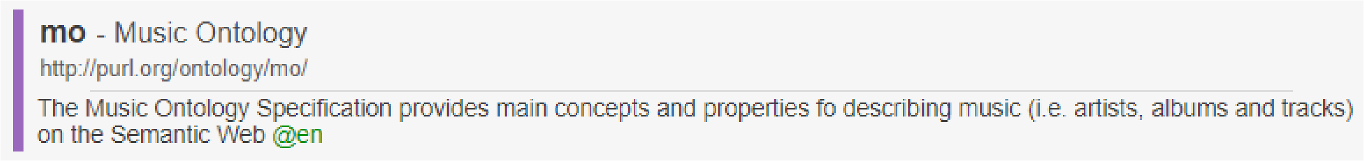
Q6.10 practice:

1. Connect to the LOV directory: http://lov.okfn.org/
2. Search for schemas talking about “music artist”.
3. What is the top ontology you find?
4. What is its version number?
5. Is it reused by other ontologies?
6. How many classes and properties does it have?
7. What expressivity does it use? (RDFS, OWL)

1) http://lov.okfn.org/

2) music artist

3) the top ontology found is :

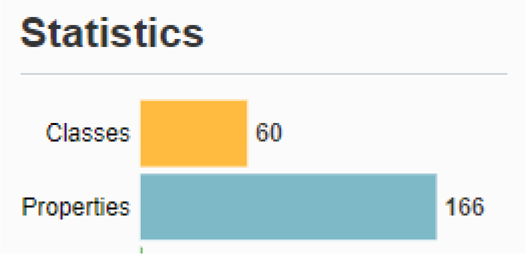


mo - Music Ontology
http://purl.org/ontology/mo/
The Music Ontology Specification provides main concepts and properties fo describing music (i.e. artists, albums and tracks) on the Semantic Web @en

4) Its version number is v2.15, from 2013.

5) It is reused by other ontologies such the playlist ontology (plo).

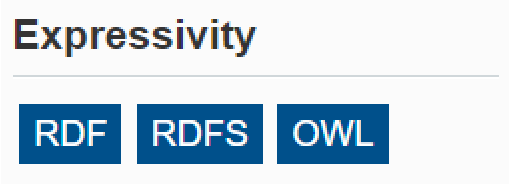
6)



Statistics

Classes	60
Properties	166

7)



Expressivity

RDF RDFS OWL

Day 05: questions from the course on other data formats.

Q7.1 What are the triples produced with this mapping and this table?

```
:My_Table rdf:type rr:TriplesMap ;
  rr:subjectMap [  rr:template
"https://www.ietf.org/rfc/rfc{NUM}.txt"; ];
  rr:predicateObjectMap [
    rr:predicateMap [  rr:predicate dc:title  ];
    rr:objectMap [     rr:column "ttl"     ]
  ].
```

ID	NUM	Ttl
87	2616	Hypertext Transfer Protocol -- HTTP/1.1
88	2396	Uniform Resource Identifiers (URI): Generic Syntax

```
@prefix dc: <http://purl.org/dc/terms/> .

<https://www.ietf.org/rfc/rfc2616.txt>
  dc:title "Hypertext Transfer Protocol -- HTTP/1.1" .
<https://www.ietf.org/rfc/rfc2396.txt>
  dc:title "Uniform Resource Identifiers (URI): Generic Syntax" .
```

Q7.2 What are the triples encoded in this HTML?

```
<div vocab="http://xmlns.com/foaf/0.1/" resource="#cathy"
typeof="Person">
 <p> <span property="name">Catherine Faron</span>
  (mail: <span property="mbox">faron@i3s.unice.fr</span>) is a
friend of
  <span property="knows"
resource="http://ns.inria.fr/fabien.gandon#me">Fabien Gandon</span>
 </p>
</div>
```

Assuming a base of http://www.example.com/ :

```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .

<http://www.example.com/#cathy>
   rdf:type foaf:Person;
   foaf:name "Catherine Faron";
   foaf:mbox "faron@i3s.unice.fr";
   foaf:knows <http://ns.inria.fr/fabien.gandon#me> .
```

Q7.3 practice:

1. Look at the Web Page
   https://www.w3.org/TR/xhtml-rdfa-scenarios/scenario-2.html
2. Call the translator on this Web page to get Turtle:
   http://rdf-translator.appspot.com/
3. What does the extracted triple say?
4. Do the same with:
   http://schema.org/docs/schema_org_rdfa.html
   What kind of data is represented in that page?
5. Again, what are the different subjects described in RDFa in this page:
   http://iricelino.org/rdfa/sample-annotated-page.html

3)

```
@prefix dc: <http://purl.org/dc/terms/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<https://www.w3.org/TR/xhtml-rdfa-scenarios/scenario-2.html> dc:creator
"Paul"@en .
```

It says that the HTML page scenario-2.html was created by Paul

4)

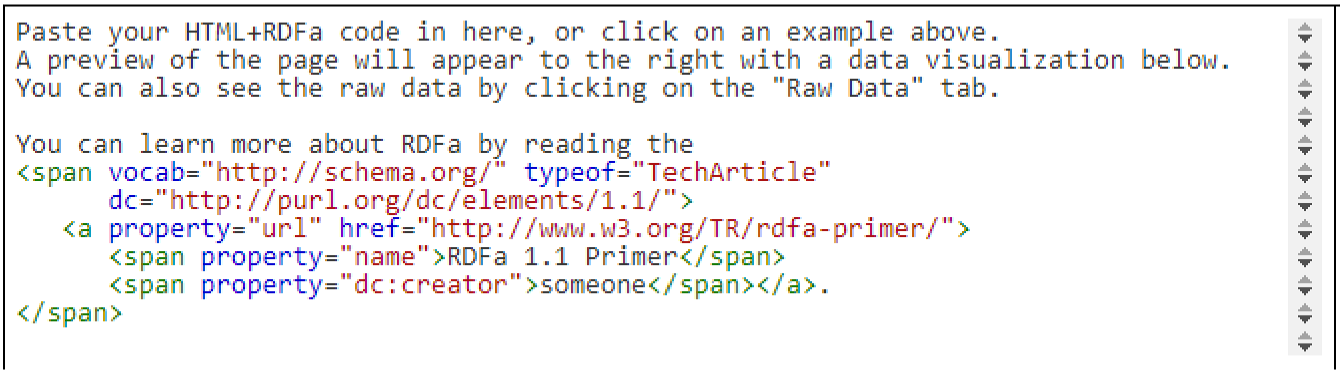
http://schema.org/docs/schema_org_rdfa.html defines classes, i.e. it defines an ontology
It times out http://rdf-translator.appspot.com/

5) http://iricelino.org/rdfa/sample-annotated-page.html

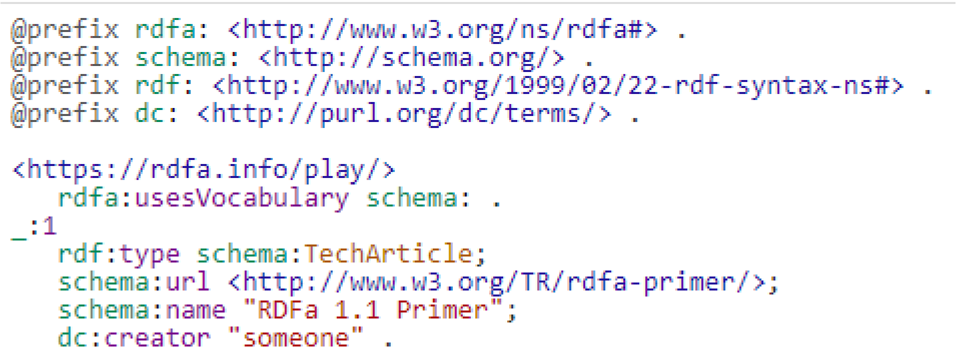
```
@prefix cc: <http://creativecommons.org/ns#> .
@prefix dc: <http://purl.org/dc/terms/> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
@prefix ns1: <dbp:> .
@prefix ns2: <cal:> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<dbr:Baruch_Spinoza> a foaf:Person ;
    ns1:influenced <dbr:Albert_Einstein>,
        <dbr:Arthur_Schopenhauer> ;
    foaf:familyname "Spinoza"@en .

<http://iricelino.org/rdfa/sample-annotated-page.html> cc:license
<http://creativecommons.org/licenses/by-nc-sa/3.0/> ;
    dc:creator "Irene Celino"@en ;
    dc:title "Sample page annotated with RDFa 1.1"@en .

<http://iricelino.org/rdfa/sample-annotated-page.html#me> a foaf:Person ;
    foaf:givenname "Giovanni"@it ;
    foaf:homepage <http://example.org/blog/> ;
    foaf:knows [ a foaf:Person ;
            foaf:mbox <mailto:sue@example.org> ;
            foaf:name "Sue"@en ] ;
    foaf:mbox <mailto:john@example.org> ;
    foaf:name "John Doe"@en .

<urn:ISBN:0091808189> a <biblio:book> ;
    dc:title "Canteen Cuisine"@en .

<urn:ISBN:1596913614> a <biblio:book> ;
    dc:description "White's autobiography"@en .

<dbr:Albert_Einstein> ns1:birthPlace <dbr:Germany> ;
    ns1:citizenship <dbr:Germany>,
        <dbr:United_States> ;
    ns1:dateOfBirth "1879-03-14"^^xsd:date ;
    foaf:depiction
<http://upload.wikimedia.org/wikipedia/commons/thumb/d/d3/Albert_Einstein_Head.
jpg/460px-Albert_Einstein_Head.jpg> ;
    foaf:name "Albert Einstein"@en .

<dbr:Arthur_Schopenhauer> a foaf:Person .

<http://example.org/blog/> dc:title "Understanding Semantics"@en ;
    foaf:primaryTopic <http://iricelino.org/rdfa/sample-annotated-
page.html#semantics> .

<dbr:Germany> ns1:conventionalLongName "Federal Republic of Germany"@en .

[] a ns2:Vevent ;
    ns2:dtstart "2015-09-16T16:00:00-05:00"^^xsd:dateTime ;
    ns2:summary "one last summer Barbecue"@en .
```

This page is about mainly about books and person.

Q7.4 Use the online tool to play with RDFa adding for instance a “creator” property https://rdfa.info/play/



```
Paste your HTML+RDFa code in here, or click on an example above.
A preview of the page will appear to the right with a data visualization below.
You can also see the raw data by clicking on the "Raw Data" tab.

You can learn more about RDFa by reading the
<span vocab="http://schema.org/" typeof="TechArticle"
      dc="http://purl.org/dc/elements/1.1/">
   <a property="url" href="http://www.w3.org/TR/rdfa-primer/">
      <span property="name">RDFa 1.1 Primer</span>
      <span property="dc:creator">someone</span></a>.
</span>
```



```
@prefix rdfa: <http://www.w3.org/ns/rdfa#> .
@prefix schema: <http://schema.org/> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix dc: <http://purl.org/dc/terms/> .

<https://rdfa.info/play/>
   rdfa:usesVocabulary schema: .
_:1
   rdf:type schema:TechArticle;
   schema:url <http://www.w3.org/TR/rdfa-primer/>;
   schema:name "RDFa 1.1 Primer";
   dc:creator "someone" .
```

Added dc prefix and creator property.

Q7.5 IMDB uses RDFa – OGP for the I like button

1. Choose a movie on IMDB http://www.imdb.com
2. Copy the URL of the page of the movie
3. Go to the RDFa 1.0 RDFa Distiller and Parser: https://www.w3.org/2007/08/pyRdfa/
4. Open the URI option, past the URL of the movie page and configure and perform the extraction to get Turtle
5. Try also the transformation on the translator: http://rdf-translator.appspot.com/

1) “Downton Abbey”

2) http://www.imdb.com/title/tt1606375/?ref_=nv_sr_1

4)

```
@prefix ns1: <http://www.w3.org/1999/xhtml/vocab#> .
@prefix ns2: <http://www.facebook.com/2008/> .
@prefix og: <http://ogp.me/ns#> .

<http://www.imdb.com/title/tt1606375/?ref_=nv_sr_1> og:description "Created by Julian
Fellowes.  With Hugh Bonneville, Phyllis Logan, Elizabeth McGovern, Brendan Coyle. A
chronicle of the lives of the British aristocratic Crawley family and their servants in
the early 20th Century.";
    og:image "https://images-na.ssl-images-
amazon.com/images/M/MV5BY2U1NmIwYzgtNjFkOS00YWUxLTg0YTMtZmE5NTA3YjRmY2NlXkEyXkFqcGdeQXV
yNTA4NzY1MzY@._V1_UY1200_CR106,0,630,1200_AL_.jpg";
    og:site_name "IMDb";
    og:title "Downton Abbey (TV Series 2010-2015)";
    og:type "video.tv_show";
    og:url "http://www.imdb.com/title/tt1606375/";
    ns2:fbmlapp_id "115109575169727" .

<http://www.imdb.com/title/tt1606375/?ref_=nv_sr_1#twitterIframe> ns1:role
ns1:presentation .

[] ns1:role ns1:button .
```

5)

```
@prefix fb: <http://www.facebook.com/2008/fbml> .
@prefix ns1: <http://www.facebook.com/2008/> .
@prefix og: <http://ogp.me/ns#> .
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix xhv: <http://www.w3.org/1999/xhtml/vocab#> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<http://www.imdb.com/title/tt1606375/?ref_=nv_sr_1> og:description "Created by
Julian Fellowes.  With Hugh Bonneville, Phyllis Logan, Elizabeth McGovern,
Brendan Coyle. A chronicle of the lives of the British aristocratic Crawley
family and their servants in the early 20th Century." ;
    og:image "https://images-na.ssl-images-
amazon.com/images/M/MV5BY2U1NmIwYzgtNjFkOS00YWUxLTg0YTMtZmE5NTA3YjRmY2NlXkEyXkF
qcGdeQXVyNTA4NzY1MzY@._V1_UY1200_CR106,0,630,1200_AL_.jpg" ;
    og:site_name "IMDb" ;
    og:title "Downton Abbey (TV Series 2010–2015)" ;
    og:type "video.tv_show" ;
    og:url "http://www.imdb.com/title/tt1606375/" ;
    ns1:fbmlapp_id "115109575169727" .

<http://www.imdb.com/title/tt1606375/?ref_=nv_sr_1#twitterIframe> xhv:role
xhv:presentation .

[] xhv:role xhv:button .
```

Q7.6 Test JSON-LD online

1. Transform your FOAF profile in JSON-LD with the translator: http://rdf-translator.appspot.com/
2. Use the following online tool to generate different variations of JSON-LD of your profile (expanded, collapsed, flattened, etc.) http://json-ld.org/playground/

```
{
  "@graph": [
    {
      "@id": "file:///base/data/home/apps/s%7Erdf-
translator/1.380697414950152317/",
      "@type": "http://xmlns.com/foaf/0.1/PersonalProfileDocument",
      "http://webns.net/mvcb/errorReportsTo": {
        "@id": "mailto:leigh@ldodds.com"
      },
      "http://webns.net/mvcb/generatorAgent": {
        "@id": "http://www.ldodds.com/foaf/foaf-a-matic"
      },
      "http://xmlns.com/foaf/0.1/maker": {
        "@id": "file:///base/data/home/apps/s%7Erdf-
translator/1.380697414950152317/#me"
      },
      "http://xmlns.com/foaf/0.1/primaryTopic": {
        "@id": "file:///base/data/home/apps/s%7Erdf-
translator/1.380697414950152317/#me"
      }
    },
    {
      "@id": "file:///base/data/home/apps/s%7Erdf-
translator/1.380697414950152317/#me",
      "@type": "http://xmlns.com/foaf/0.1/Person",
      "http://purl.org/vocab/relationship/Relationship": {
        "@id": "http://ns.inria.fr/fabien.gandon#me"
      },
      "http://xmlns.com/foaf/0.1/family_name": "Tholon",
      "http://xmlns.com/foaf/0.1/givenname": "Alain",
      "http://xmlns.com/foaf/0.1/knows": {
        "@id": "_:ub20bL17C16"
      },
      "http://xmlns.com/foaf/0.1/mbox_sha1sum":
"0c192e6a3e710c36253e2e2932cef0b79ec7099d",
      "http://xmlns.com/foaf/0.1/name": "Alain Tholon",
      "http://xmlns.com/foaf/0.1/schoolHomepage": {
        "@id": "https://www.datasciencetech.institute/"
      },
      "http://xmlns.com/foaf/0.1/title": "Mr"
    },
    {
      "@id": "_:ub20bL17C16",
      "@type": "http://xmlns.com/foaf/0.1/Person",
      "http://xmlns.com/foaf/0.1/mbox_sha1sum":
"d2f304c4fda0038ee688d45b8fb5a700cc734c2c",
      "http://xmlns.com/foaf/0.1/name": "Seb"
    }
  ]
}
```

Q7.7 To provide the metadata of a CSV file I can...

- ~~□ include them in a special column of the CSV.~~
- □ put them in a file with the same name plus “-metadata.json”.
- ~~□ put them in the first line of my CSV file.~~
- □ put them in a file called “csv-metadata.json” in the same directory.
- ~~□ add the URL of the metadata file to the content of my CSV file.~~

all true, except the ones strikethrough above

Q7.8 TV Catalog : Imagine we submit the following call to an LDP platform

```
GET /catalog/tv/ HTTP/1.1
Host: example.org
Accept: text/turtle; charset=UTF-8
```

and we receive the following answer:

```
HTTP/1.1 200 OK
Content-Type: text/turtle; charset=UTF-8
Link: <http://www.w3.org/ns/ldp#Resource>; rel="type",
<http://www.w3.org/ns/ldp#DirectContainer>; rel="type"
Allow: OPTIONS,HEAD,GET,POST,PUT
Accept-Post: text/turtle, application/ld+json
Content-Length: 232
ETag: W/"90231678"
@prefix ldp: <http://www.w3.org/ns/ldp#> .
@prefix dcterms: <http://purl.org/dc/terms/> .
@prefix cat: <http://example.org/vocab/catalog#> .
<> a ldp:DirectContainer;   ldp:membershipResource <#cat>;
ldp:hasMemberRelation cat:hasProduct;
  dcterms:title "Container of the TV descriptions";
  ldp:contains <tv1>, <tv2> .
<#cat> a cat:Catalog;   dcterms:title "Catalog of TVs";   cat:hasProduct <tv1>,
<tv2> .
```

Which ones of the following statements are true?

- □ the container is just a basic container.
- □ the container is a direct container.
- □ the container is an indirect container.
- □ the platform accepts the GET calls.
- □ the platform accepts the PATCH calls.
- □ the platform accepts RDF/XML format.
- □ the platform accepts RDF Turtle.
- □ the platform accepts RDF JSON-LD.
- □ a link `hasProduct` is automatically created between the resource #cat and the resources of this container

Skipped during the course.