

Semantic Web handout including: 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. You can use screenshots when appropriate as an answer to a 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.

First name: Maher

Family name: SEBAI

Email: maher.sebai@edu.dsti.institute

Day 01: questions from the course.

Q1.1 Practice XML replace missing parts

```
<art_book>
  <short_title>Architecture Now</short_title>
  <first_author>Jodidio, Philip</first_author>
  <ID isbn10="3822840912" />
</art_book>
```

Q1.2 Provide 10 first lines

Get 10 first lines of the five results for:

<http://www.wikidata.org/entity/Q23014205>
<http://www.wikidata.org/entity/Q23014205.json>
<http://www.wikidata.org/entity/Q23014205.rdf>
<http://www.wikidata.org/entity/Q23014205.ttl>
<http://www.wikidata.org/entity/Q23014205.nt>

<ANSWER HERE/>

<http://www.wikidata.org/entity/Q23014205>:

```
<!DOCTYPE html>

<html class="client-nojs" lang="en" dir="ltr">

<head>

<meta charset="UTF-8"/>

<title>Fabien Gandon - Wikidata</title>

<script>document.documentElement.className =
document.documentElement.className.replace( /(^|\s)client-nojs(\s|$)/,
"$1client-js$2" );</script>

<script>(window.RLQ=window.RLQ||[]).push(function()
```

<https://www.wikidata.org/wiki/Special:EntityData/Q23014205.json>

```
{"entities": {"Q23014205": {"pageid": 25028548, "ns": 0, "title": "Q23014205", "lastrevid": 818593402, "modified": "2018-12-24T07:42:39Z", "type": "item", "id": "Q23014205", "labels": {"fr": {"language": "fr", "value": "Fabien Gandon"}, "en": {"language": "en", "value": "Fabien Gandon"}, "br": {"language": "br", "value": "Fabien Gandon"}, "de": {"language": "de", "value": "Fabien Gandon"}, "af": {"language": "af", "value": "Fabien Gandon"}, "an": {"language": "an", "value": "Fabien Gandon"}, "ast": {"language": "ast", "value": "Fabien Gandon"}, "bar": {"language": "bar", "value": "Fabien Gandon"}, "bm": {"language": "bm", "value": "Fabien Gandon"}, "ca": {"language": "ca", "value": "Fabien Gandon"}, "co": {"language": "co", "value": "Fabien Gandon"}, "cs": {"language": "cs", "value": "Fabien Gandon"}, "cy": {"language": "cy", "value": "Fabien Gandon"}, "da": {"language": "da", "value": "Fabien Gandon"}, "de-at": {"language": "de-at", "value": "Fabien Gandon"}, "de-ch": {"language": "de-ch", "value": "Fabien Gandon"}, "en-ca": {"language": "en-ca", "value": "Fabien Gandon"}, "en-gb": {"language": "en-gb", "value": "Fabien Gandon"}, "eo": {"language": "eo", "value": "Fabien Gandon"}}}
```

<http://www.wikidata.org/entity/Q23014205.rdf>

```
<?xml version="1.0"?>
<rdf:RDF xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:xsd="http://www.w3.org/2001/XMLSchema#"
xmlns:ontolex="http://www.w3.org/ns/lemon/ontolex#"
xmlns:dct="http://purl.org/dc/terms/" xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#" xmlns:owl="http://www.w3.org/2002/07/owl#"
xmlns:wikibase="http://wikiba.se/ontology#"
xmlns:wds="http://www.wikidata.org/entity/statement/"
xmlns:wdata="https://www.wikidata.org/wiki/Special:EntityData/"
xmlns:skos="http://www.w3.org/2004/02/skos/core#" xmlns:schema="http://schema.org/"
xmlns:ccc="http://creativecommons.org/ns#"
xmlns:geo="http://www.opengis.net/ont/geosparql#"
xmlns:prov="http://www.w3.org/ns/prov#"
xmlns:wdref="http://www.wikidata.org/reference/"
xmlns:wdv="http://www.wikidata.org/value/" xmlns:wd="http://www.wikidata.org/entity/"
xmlns:wdt="http://www.wikidata.org/prop/direct/"
xmlns:wdtn="http://www.wikidata.org/prop/direct-normalized/"
xmlns:p="http://www.wikidata.org/prop/"
xmlns:ps="http://www.wikidata.org/prop/statement/"
xmlns:psv="http://www.wikidata.org/prop/statement/value/"
xmlns:psn="http://www.wikidata.org/prop/statement/value-normalized/"
xmlns:pq="http://www.wikidata.org/prop/qualifier/"
xmlns:pqv="http://www.wikidata.org/prop/qualifier/value/"
```

```

xmlns:pqn="http://www.wikidata.org/prop/qualifier/value-normalized/"
xmlns:pr="http://www.wikidata.org/prop/reference/"
xmlns:prv="http://www.wikidata.org/prop/reference/value/"
xmlns:prn="http://www.wikidata.org/prop/reference/value-normalized/"
xmlns:wdno="http://www.wikidata.org/prop/novalue/">
    <rdf:Description
        rdf:about="https://www.wikidata.org/wiki/Special:EntityData/Q23014205">
            <rdf:type rdf:resource="http://schema.org/Dataset"/>
            <schema:about rdf:resource="http://www.wikidata.org/entity/Q23014205"/>
            <cc:license
                rdf:resource="http://creativecommons.org/publicdomain/zero/1.0/"/>
                <schema:softwareVersion>1.0.0</schema:softwareVersion>
                <schema:version
                    rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">818593402</schema:version>
                    <schema:dateModified
                        rdf:datatype="http://www.w3.org/2001/XMLSchema#dateTime">2018-12-
                        24T07:42:39Z</schema:dateModified>
                        <wikibase:statements
                            rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">29</wikibase:statements>

```

<http://www.wikidata.org/entity/Q23014205.ttl>

```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
@prefix ontolex: <http://www.w3.org/ns/lemon/ontolex#> .
@prefix dct: <http://purl.org/dc/terms/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
@prefix wikibase: <http://wikiba.se/ontology#> .
@prefix wds: <http://www.wikidata.org/entity/statement/> .
@prefix wdata: <https://www.wikidata.org/wiki/Special:EntityData/> .
@prefix skos: <http://www.w3.org/2004/02/skos/core#> .

```

<http://www.wikidata.org/entity/Q23014205.nt>

Failed – Network Error

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?

<ANSWER HERE/>

- <http://dbpedia.org/page/London>
- dbp:populationDemonym: Londoner
- <http://dbpedia.org/page/Vienna>
- Vindobona

Q1.4 WHO.IS?

1. contact for inria.fr

2. contact for fabien.info
3. contact for lemonde.fr

<ANSWER HERE/>

1. florian.dufour@inria.fr
2. REDACTED FOR PRIVACY
3. domain_names@lemonde.fr

Q1.5 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 the “URI of the name of Victor Hugo” in the Library of Congress:
<http://id.loc.gov/authorities/names/n79091479>
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

<ANSWER HERE/>

1.a.

```
<?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"
```

1.b.

```
<?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.a

```

<?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>

```

2.b

```

<?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.a

```

<?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">
<
<!-- header -->
<head profile="http://www.w3.org/1999/xhtml/vocab">
    <meta charset="utf-8" />
    <meta http-equiv="X-UA-Compatible" content="IE=edge" />
    <meta name="viewport" content="width=device-width, initial-scale=1.0" />
    <title>About: Vienna</title>

```

3.b

```

<?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#"
    xmlns:geo="http://www.w3.org/2003/01/geo/wgs84_pos#"
    xmlns:prov="http://www.w3.org/ns/prov#"
    xmlns:ns10="http://dbpedia.org/ontology/PopulatedPlace/"
    xmlns:georss="http://www.georss.org/georss/"
    xmlns:ns12="http://purl.org/linguistics/gold/" >
    <rdf:Description rdf:about="http://dbpedia.org/resource/243_Ida">
```

```

<dbp:discoverySite rdf:resource="http://dbpedia.org/resource/Vienna" />
</rdf>Description>
```

4.

```

<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML+RDFa 1.0//EN"
"http://www.w3.org/MarkUp/DTD/xhtml-rdfa-1.dtd">
<html version="XHTML+RDFa 1.0" xmlns="http://www.w3.org/1999/xhtml"
xmlns:madsrdf="http://www.loc.gov/mads/rdf/v1#"
xmlns:ri="http://id.loc.gov/ontologies/RecordInfo#"
xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"
xmlns:skos="http://www.w3.org/2004/02/skos/core#"
xmlns:skosxl="http://www.w3.org/2008/05/skos-xl#"
xmlns:owl="http://www.w3.org/2002/07/owl#" xmlns:cs="http://www.w3.org/2003/06/sw-
vocab-status/ns#" xmlns:dcterms="http://purl.org/dc/terms/">
<head>
<title>Hugo, Victor, 1802-1885 - LC Linked Data Service: Authorities and Vocabularies
| Library of Congress</title>
<meta name="description" content=" The Linked Data Service provides access to
commonly found standards and vocabularies promulgated by the Library of Congress.
This includes data values and the controlled vocabularies that house them. Datasets
available include LCSH, BIBFRAME, LC Name Authorities, LC Classification, MARC codes,
PREMIS vocabularies, ISO language codes, and more."/>
<link rel="schema.DC" href="http://purl.org/dc/elements/1.1/" />
<link rel="dc.relation.isPartOf" href="//www.loc.gov/" title="Library of Congress"/>
```

5.a

```

<!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="https://creativecommons.org/licenses/by/4.0/" rel="license"/><link type="image/vnd.microsoft.icon" href="/favicon.ico" rel="shortcut
icon"/><link href="/uniprot.min.css2019_02" type="text/css" rel="stylesheet"/><script
type="text/javascript">
    var BASE = '/';
    var ua = window.navigator.userAgent;
    var directory = (~ua.indexOf('MSIE ') || ~ua.indexOf('Trident/')) === 0 ? "non-ie" : "ie";
</script><script src="/scripts/frontier/d3/d3.v3.min.js" type="text/javascript"></script><script src="/js-compr.js2019_02" type="text/javascript">
    uniprot.namespace = 'uniprot';
    uniprot.releasedate = '2019_02';
</script><script type="text/javascript">
;
```

5.b

```

<?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/P43121">
<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">1995-11-01</created>
<modified rdf:datatype="http://www.w3.org/2001/XMLSchema#date">2019-02-13</modified>
<version rdf:datatype="http://www.w3.org/2001/XMLSchema#int">172</version>
<mnemonic>MUC18_HUMAN</mnemonic>
<oldMnemonic>MU18_HUMAN</oldMnemonic>

```

6. Topic: Xavier Dolan URI. Format: RDF in JSON format file
7. Topic: “Distributed Artificial Intelligence for Distributed Corporate Knowledge Management” article. Format: RDF in Turtle format

Q1.6 Recall five best practices of linked open data



<ANSWER HERE/>

1. On the web, open license
2. Machine readable data
3. Non proprietary format
4. RDF standard
5. Linked RDF

Q1.7 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.

<ANSWER HERE WITH THE COPY-PASTED ANOTATED TEXT OR SCREENSHOT/>



Confidence:

 0.5

Language: English

n-best candidates

[SELECT TYPES...](#)

[ANNOTATE](#)

In early 2005, the band, at the time named "X-Tazy", created and independently released [Double Face](#), a self-produced [fan EP](#).

In early 2006, the band got an opportunity to open for [Robert Plant](#) and [Adagio](#), playing for over 7000 music fans and meeting their future producer, [Adagio's keyboard](#) player [Kevin Codfert](#). With the support of Kevin, [Myrath](#) recorded their first full-length album [Hope](#). The band was signed by the [French](#) label Brennus-Music and [Hope](#) was released in 2007 in [France](#). The album was well-received, and led to a number of festival performances in [Europe](#). They are heavily influenced by folkloric [Tunisian music](#) and their producer, [Kevin Codfert](#) supported their [endeavour](#) in creating a distinct genre for the band, a genre that combines Metal music with Tunisian [folklore](#).^[1]

Their follow-up album [Desert Call](#) was released in early 2010. It was a second turning point in [Myrath](#)'s career in terms of increased recognition and increasing their fan base, garnering an even wider interest from the media and promoters.

The third [Myrath](#) album [Tales of the Sands](#) was released in late 2011. They supported the album by touring with bands such as [Dream Theater](#), [HIM](#), [W.A.S.P.](#), [Tarja Turunen](#) and toured around the world (including [Dubai](#), [USA](#), [India](#), [Europe](#), [North Africa](#)) over a few years. The band also made their first live appearance on US soil at [ProgPower USA](#), in 2013, and headlined [ProgPower Europe](#) in [Netherlands](#). [Myrath](#) have secured a slot on the new [Symphony X](#) tour in [Europe](#) for early 2016. Still in 2016, they released their fourth album, [Legacy](#). Also in 2016, vocalist Zaher Zorgati was cast as a guest singer in [Ayreon](#)'s album [The Source](#).^[2]

[BACK TO TEXT](#)

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)

<ANSWER HERE/>

RDF is a Model of a directed labeled multigraphs

Subject: vertex

Predicate: edge/ identifies the type of arc

Object: vertex

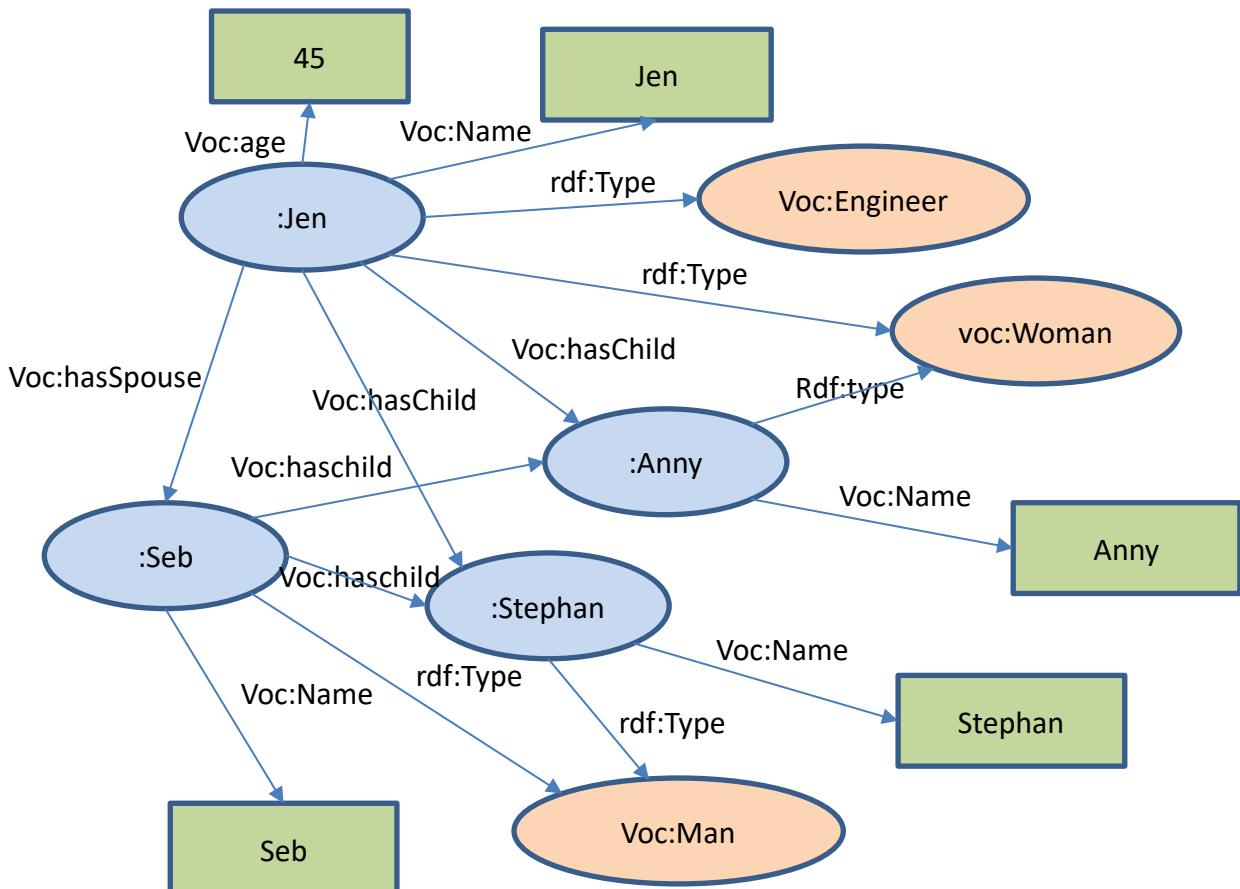
Q2.1 Fill the blanks

"Jen is an engineer woman, 45-year old, married to Seb who is a man with whom she had two children: Anny who is a woman and Stefan who is a man". For each person we also explicitly specify the name.

To fill the blanks we use the values: :Seb, :Stefan, voc:name, voc:hasChild, voc:age, voc:hasSpouse, rdf:type, voc:Engineer, voc:Man, "Jen", "Seb", "Anny", "Stefan"

For each person we also explicitly specify the name

<ANSWER HERE BY REPLACING ALL THE QUESTION MARKS/>:



Q2.2 Fill the blanks (RDF/XML)

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml version="1.0" encoding="UTF-8"?>
<!DOCTYPE rdf:RDF [    <!ENTITY vocab "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="&vocab;" xml:base="http://www.unice.fr/data">
<voc:Woman rdf:about="#Jen">
  <voc:name>Jen</voc:name>
  <voc:age
    rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">45</voc:age>
    <voc:hasSpouse rdf:resource="#Seb"></voc:hasSpouse >
    <voc:hasChild rdf:resource="#Stefan"></voc:hasChild>
    <voc:hasChild>
      <rdf:Description rdf:about="#Anny">
        <voc:name>Anny</voc:name>
        <rdf:type rdf:resource="&vocab;#Woman"></rdf:type>
      </rdf:Description>
    </voc:hasChild>
    <rdf:type rdf:resource="&vocab;#Engineer"></rdf:type>
  </voc:Woman>
<voc:Man rdf:about="#Seb">
  <voc:name>Seb</voc:name>
  <voc:hasChild rdf:resource="#Stefan"></voc:hasChild>
  <voc:hasChild rdf:resource="#Anny"></voc:hasChild>
</voc:Man>
<voc:Man rdf:about="#Stefan">
  <voc:name>Stefan</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#Jen> a voc:Engineer , voc:Woman ;
  voc:age "45"^^xsd:string ;
  voc:hasChild <http://www.unice.fr/data#Anny>,
<http://www.unice.fr/data#Stephan>;
  voc:hasSpouse <http://www.unice.fr/data#Seb> ;
  voc:name "Jen" .
<http://www.unice.fr/data#Seb> a voc:Man ;
  voc:hasChild <http://www.unice.fr/data#Anny>,
  <http://www.unice.fr/data#Stephan> ;
  voc:name "Seb" .
<http://www.unice.fr/data#Anny> a voc:Woman ;
```

```
voc:name "Anny" .  
< http://www.unice.fr/data#Stephan > a voc:Man ;  
    voc:name "Stephan".
```

Q2.4 Visit me please

Get the RDF data from: <http://ns.inria.fr/fabien.gandon#me>

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:
<http://cltl.nl/visualrdf/>
<http://www.easyrdf.org/converter> (PNG, SVG)
6. Adapt to your data and do it again

<ANSWER HERE/>

2. The retrieved RDF is in XML format

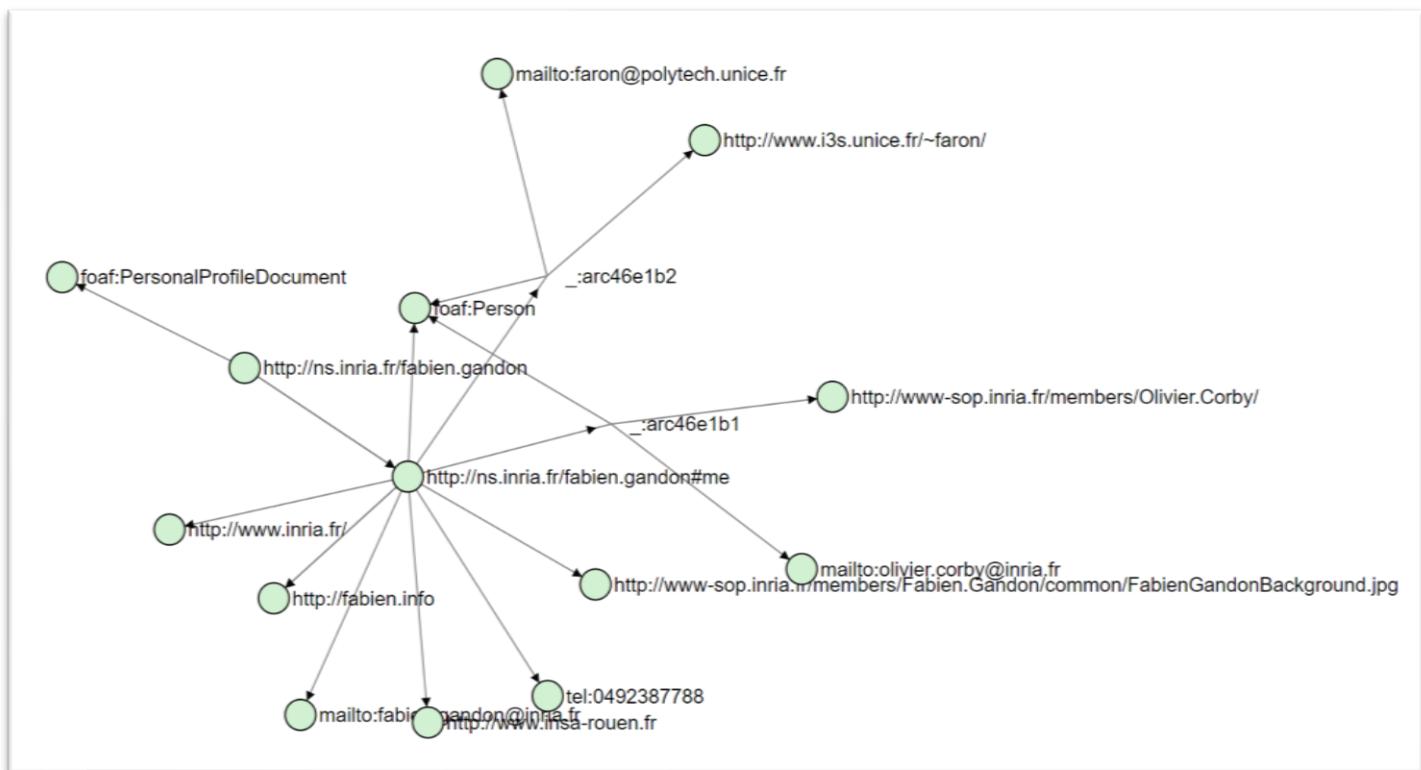
```
@prefix foaf: <http://xmlns.com/foaf/0.1/> .  
  
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
  
@prefix xml: <http://www.w3.org/XML/1998/namespace> .  
  
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .  
  
  
<http://ns.inria.fr/fabien.gandon>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

```



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>

```

<ANSWER HERE/>

This RDF describes the doc.html file: it is of type Report, and the Theme is Music + History. The Report has 23 Pages

Graph of the data model



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?

<ANSWER HERE/>

The RDF is in XML format/syntax

6. madsrdf:elementValue "، هوجو، فيكتور"@en

The error is that @en is mentioned for all the languages.

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> .
}
```

<ANSWER HERE/>

It is a TRIG syntax document: a named graph. A document `inria.fr/rr/doc.html` has a subject `data.bnf.fr/ark:/12148/cb121105993`

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?

<ANSWER HERE/>

The RDF is in XML format/syntax

The structure of this data used reification instead of named graphs

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

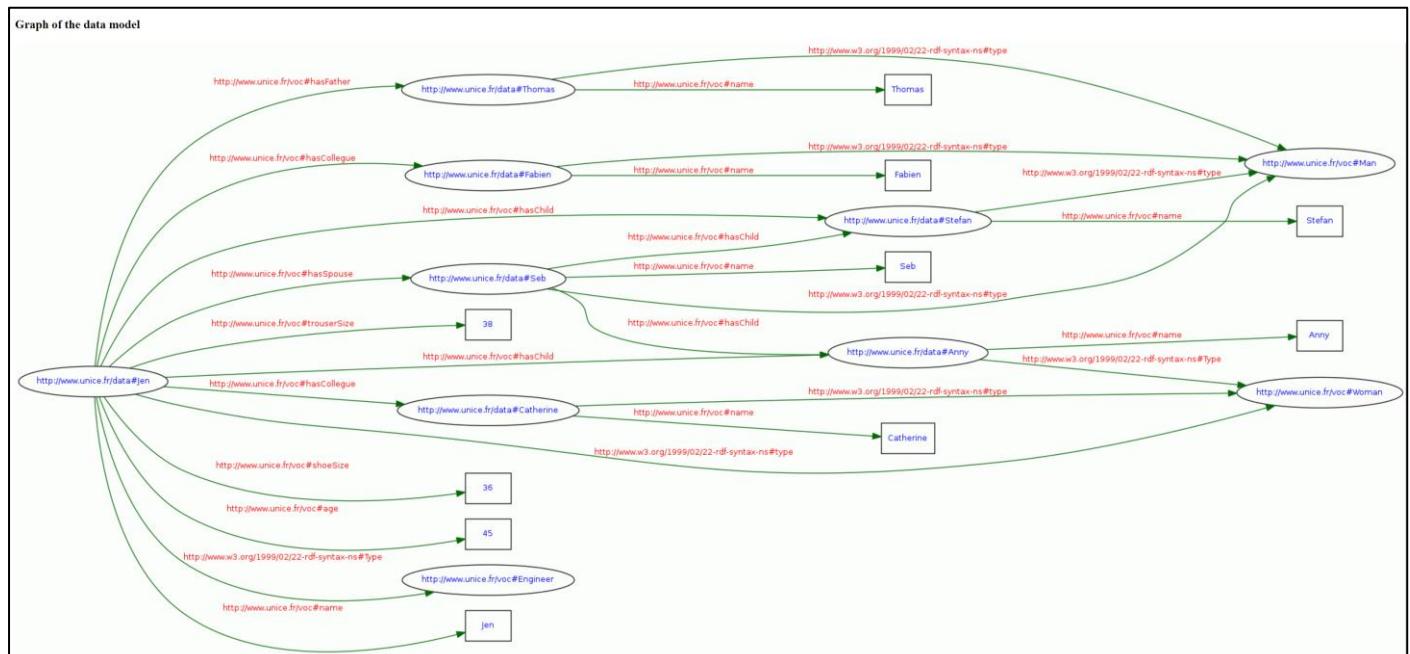
Read carefully the following statements:

"Jen is a 45-year old woman and she has a shoe size of 36 and trouser size of 38. She is, married to Seb who is a man with whom she had two children: Anny who is a woman and Stefan who is a man. Jen is also an engineer and Catherine and Fabien are her colleagues. Jen's father is a man named Thomas"

1. Use your text editor and write the above statements in RDF in N3 syntax inventing your own vocabulary. Save you file as "Jen.ttl"
2. Use your favorite text or XML editor and write the above statements in RDF in XML syntax reusing the same vocabulary "Jen.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:

<ANSWER HERE/>



```

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@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#Jen> a voc:Woman ;
    voc:age 45 ;
    voc:shoeSize 36;
    voc:trouserSize 38;
    voc:hasColleague <http://www.unice.fr/data#Fabien>,
        <http://www.unice.fr/data#Catherine>;
    voc:hasFather <http://www.unice.fr/data#Thomas>;
    voc:hasChild <http://www.unice.fr/data#Anny>,
        <http://www.unice.fr/data#Stefan> ;
    voc:hasSpouse <http://www.unice.fr/data#Seb> ;
    voc:name "Jen" ;
    rdf:type voc:Engineer .

<http://www.unice.fr/data#Seb> a voc:Man ;
    voc:hasChild <http://www.unice.fr/data#Anny>,
        <http://www.unice.fr/data#Stefan> ;
    voc:name "Seb" .

<http://www.unice.fr/data#Anny> voc:name "Anny" ;
    rdf:type voc:Woman .

<http://www.unice.fr/data#Stefan> a voc:Man ;
    voc:name "Stefan" .

<http://www.unice.fr/data#Fabien> a voc:Man ;
    voc:name "Fabien" .

<http://www.unice.fr/data#Catherine> a voc:Woman ;
    voc:name "Catherine" .

<http://www.unice.fr/data#Thomas> a voc:Man ;
    voc:name "Thomas" .

```

Code of validated RDF in XML syntax:

<ANSWER HERE/>

```

<?xml version="1.0" encoding="UTF-8"?>

<rdf:RDF

    xmlns:rdf="http://www.w3.org/1999/02/22-rdf-syntax-ns#"
    xmlns:voc="http://www.unice.fr/voc#"

>

<rdf:Description rdf:about="http://www.unice.fr/data#Thomas">
    <rdf:type rdf:resource="http://www.unice.fr/voc#Man"/>
    <voc:name>Thomas</voc:name>

```

```
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Fabien">
  <voc:name>Fabien</voc:name>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Man"/>
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Jen">
  <voc:hasColleague rdf:resource="http://www.unice.fr/data#Fabien"/>
  <voc:trouserSize rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">38</voc:trouserSize>
  <voc:hasColleague rdf:resource="http://www.unice.fr/data#Catherine"/>
  <voc:shoeSize rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">36</voc:shoeSize>
  <voc:hasSpouse rdf:resource="http://www.unice.fr/data#Seb"/>
  <voc:age rdf:datatype="http://www.w3.org/2001/XMLSchema#integer">45</voc:age>
  <voc:hasChild rdf:resource="http://www.unice.fr/data#Stefan"/>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Engineer"/>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Woman"/>
  <voc:name>Jen</voc:name>
  <voc:hasFather rdf:resource="http://www.unice.fr/data#Thomas"/>
  <voc:hasChild rdf:resource="http://www.unice.fr/data#Anny"/>
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Seb">
  <voc:hasChild rdf:resource="http://www.unice.fr/data#Anny"/>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Man"/>
  <voc:name>Seb</voc:name>
  <voc:hasChild rdf:resource="http://www.unice.fr/data#Stefan"/>
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Stefan">
  <rdf:type rdf:resource="http://www.unice.fr/voc#Man"/>
  <voc:name>Stefan</voc:name>
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Catherine">
  <voc:name>Catherine</voc:name>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Woman"/>
</rdf:Description>

<rdf:Description rdf:about="http://www.unice.fr/data#Anny">
  <voc:name>Anny</voc:name>
  <rdf:type rdf:resource="http://www.unice.fr/voc#Woman"/>
</rdf:Description>
```

```
</rdf:Description>
```

```
</rdf:RDF>
```

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 -Dfile.encoding=UTF8 " 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 "Jen.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.

The screenshot shows the Corese 3.2 interface. The title bar reads "Corese 3.2 - Wimmicks INRIA I3S - 2016-12-25". The menu bar includes File, Edit, Engine, Debug, Query, Template, Explain, ?, System, and Query2. Below the menu is a toolbar with buttons for Query, Validate, to SPIN, to SPARQL, Prove, Trace, Search, Refresh stylesheet, and Default stylesheet. The main area has tabs for Graph, XML, Table, and Validate. The validate tab is active, showing a SPARQL query:

```
1 select * where {
2   ?x ?p ?y
3 }
4
```

The results table has three columns: ?x, ?y, and ?p. The data is as follows:

?x	?y	?p
http://www.unice.fr/data#Jen	45	http://www.unice.fr/voc#age
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Stefan	http://www.unice.fr/voc#hasChild
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Anny	http://www.unice.fr/voc#hasChild
http://www.unice.fr/data#Seb	http://www.unice.fr/data#Stefan	http://www.unice.fr/voc#hasChild
http://www.unice.fr/data#Seb	http://www.unice.fr/data#Anny	http://www.unice.fr/voc#hasChild
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Fabien	http://www.unice.fr/voc#hasColleague
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Catherine	http://www.unice.fr/voc#hasColleague
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Thomas	http://www.unice.fr/voc#hasFather
http://www.unice.fr/data#Jen	http://www.unice.fr/data#Seb	http://www.unice.fr/voc#hasSpouse
http://www.unice.fr/data#Thomas	Thomas	http://www.unice.fr/voc#name
http://www.unice.fr/data#Fabien	Fabien	http://www.unice.fr/voc#name
http://www.unice.fr/data#Jen	Jen	http://www.unice.fr/voc#name
http://www.unice.fr/data#Catherine	Catherine	http://www.unice.fr/voc#name
http://www.unice.fr/data#Seb	Seb	http://www.unice.fr/voc#name
http://www.unice.fr/data#Stefan	Stefan	http://www.unice.fr/voc#name
http://www.unice.fr/data#Anny	Anny	http://www.unice.fr/voc#name
http://www.unice.fr/data#Jen	36	http://www.unice.fr/voc#shoeSize
http://www.unice.fr/data#Jen	38	http://www.unice.fr/voc#trouserSize
http://www.unice.fr/data#Jen	http://www.unice.fr/voc#Engineer	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Anny	http://www.unice.fr/voc#Woman	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Thomas	http://www.unice.fr/voc#Man	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#name	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Fabien	http://www.unice.fr/voc#Man	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Jen	http://www.unice.fr/voc#Woman	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#hasColleague	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#trouserSize	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#Catherine	http://www.unice.fr/voc#Woman	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#shoeSize	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#hasSpouse	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Seb	http://www.unice.fr/voc#Man	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#age	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/voc#hasChild	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.unice.fr/data#Stefan	http://www.unice.fr/voc#Man	http://www.w3.org/1999/02/22-rdf-syntax-ns#Type
http://www.w3.org/1999/02/22-rdf-syntax-ns#Type	http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#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-sop.inria.fr/members/Olivier.Corby/> ;
        foaf:mbox <mailto:olivier.corby@inria.fr> ;
        foaf:name "Olivier Corby" ],
        [ a foaf:Person ;
        rdfs:seeAlso <http://www.i3s.unice.fr/~faron/> ;
        foaf:mbox <mailto:faron@polytech.unice.fr> ;
        foaf:name "Catherine Faron-Zucker" ] ;
    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>

<ANSWER HERE/>

The first is the URI of the document where the **primary topic** is the person “Fabien gandan”, the second is the URI of the person 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?

<ANSWER HERE/>

BBC website does not provide white shark RDF anymore ...

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.

<ANSWER HERE/>

The triple is:

```
<http://dbpedia.org/resource/Paris>
a <http://dbpedia.org/class/yago/WikicatCapitalsInEurope>
```

4, If you don't have the human dataset file yet, 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?

<ANSWER HERE/>

```
xml:base="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?

<ANSWER HERE/>

rdf:ID . this how it is used for example

```
<Person rdf:ID="Mark">
```

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

<ANSWER HERE/>

```
xmlns=http://www.inria.fr/2007/09/11/humans.rdfs#
```

```
this is the default namespace for qualifier Person, Woman, Man Researcher
```

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

<ANSWER HERE/>

It is an Entity constant that should be replaced with what the constant was assigned to

5. Find everything about information on John in this file.

all the information:

<ANSWER HERE/>

John information:

Name = John

Shoe size = 14

Age = 37

Has child = Mark

Has friend = Alice

Has Spouse = Jennifer

Shirt size = 12

Trouser size = 44

Has parent = sophie

Has parent = Harry

6. Translate the file in turtle and save it as `human_2013.ttl`

10 first lines:

<ANSWER HERE/>

```
@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" .  
  
  
<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora> a :Woman ;  
      :age 95 ;  
      :hasChild <http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre> ;
```

```
:hasSpouse <http://www.inria.fr/2007/09/11/humans.rdf#Gaston> ;
```

```
:name "Flora" .
```

7. In the turtle version find everything about Laura.

all the information:

<ANSWER HERE/>

Laura informations:

- Has spouse = william
- Has child = Catherine
- Is lecturer
- Is a person
- Is researcher
- Has friend = Alice
- Has name = Laura

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 . }
```

<ANSWER HERE/>

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
```

<ANSWER HERE/>

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

<ANSWER HERE/>

This query returns the USA presidents list

Q30 = United States of America

P6 = Head of Government

P735 = given name

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

<ANSWER HERE/>

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

select * where { ?x a foaf:Person . } limit 20

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

<ANSWER HERE/>

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

select * where { ?x a foaf:Person . } limit 20 OFFSET 10

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

<ANSWER HERE/>

```
Select ?age where {?x c:name 'Fabien'; c:age ?age}

Select ?name where {?x c:name ?name; c:age ?age . filter (?age < 50) }

Select * where {?x c:name 'Fabien'; c:age ?age . filter (?age <50) }

Select ?p ?v where {?x ?p ?v; ?x c:name 'Fabien'; c:age ?age . filter ?age <50}

Select ?name where {?x c:name 'Fabien', ?name filter (?name != 'Fabien')}

Select ?x where {?x ?p1 ?v; ?p2 ?v filter (?p1 != ?p2) }

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 }
}
```

<ANSWER HERE/>

Yes because the minus set is ignored (no common variables)

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

<ANSWER HERE/>

```
Select ?x where {
?x c:memberOf ?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))
```

<ANSWER HERE/>

The number of book for each author sorted by the most prolific author first.

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

<ANSWER HERE/>

The first one ?x (foaf:knows | rdfs:seeAlso)+ ?y is the correct one

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 }
}
```

<ANSWER HERE/>

Try to find the name of x friend, if the name variable is bound then select the real name, otherwise print John Doe if no name is bound

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
```

<ANSWER HERE/>

A good estimate to x's age based on x relations average age.

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

<ANSWER HERE/>

```
Select * where {{?x c:study c:informatics } union {?x c:study c:mathematics}}
```

```
Select * where {{?x c:study c:informatics } union {?x c:study c:mathematics}}
```

```
Optional {?x c:name ?name} }
```

```
Select * where {{?x c:study c:informatics } union {?x c:study c:mathematics}}
```

```
Optional {graph ?g {?x c:name ?name}} }
```

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 } }
```

<ANSWER HERE/>

```
?x ?p ?y solved against g1
```

```
?y ?q ?z solved against g2
```

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

<ANSWER HERE/>

```
Insert {?x rdfs:label ?n}
```

```
Delete {?x foaf:name ?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)
  }
}
```

<ANSWER HERE/>

The query locate the age property in string format, translate the age into integer, delete the identified triple and insert a new one with age converted into integer

Q3.17 Which clauses could you use to obtained results as RDF triples following a specific pattern?

- SELECT ... WHERE {...} ...
- CONSTRUCT {} WHERE {...} ...
- DESCRIBE <...> DESCRIBE ... {...}
- ASK {...}
- DELETE { ... } INSERT { ... } WHERE {...} ...

<ANSWER HERE/>

The “construct” and the “insert” one

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 plain English.

<ANSWER HERE/>

Select the resource that has a type with prefix h: as defined in humans.rdfs, exclude all other types

Run this query. How many answers do you get?

<ANSWER HERE/>

We have 21 results in corese interface

Find John and his types.

<ANSWER HERE/>

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

<ANSWER HERE/>

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

Results:

<ANSWER HERE/>

Results	
?p	?o
http://www.inria.fr/2007/09/11/humans.rdfs#age	37
http://www.inria.fr/2007/09/11/humans.rdfs#hasParent	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs#name	John
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize	12
http://www.inria.fr/2007/09/11/humans.rdfs#shoeSize	14
http://www.inria.fr/2007/09/11/humans.rdfs#trouserSize	44
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs#Person

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

Query

<ANSWER HERE/>

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

Results:

<ANSWER HERE/>

?_ast_v_0	?_ast_p_0	?_ast_v_1	?_ast_p_1
37	http://www.inria.fr/2007/09/11/humans.rdfs#instan...	http://www.inria.fr/2007/09/11/humans.rdfs#age	
John	http://www.inria.fr/2007/09/11/humans.rdfs#name	http://www.inria.fr/2007/09/11/humans.rdfs#hasP...	
12	http://www.inria.fr/2007/09/11/humans.rdfs#shirts...		
14	http://www.inria.fr/2007/09/11/humans.rdfs#shoes...		
44	http://www.inria.fr/2007/09/11/humans.rdfs#trouse...		
http://www.inria.fr/2007/09/11/humans.rdfs#Person	http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs-instanc...	http://www.inria.fr/2007/09/11/humans.rdfs#hasC...
		http://www.inria.fr/2007/09/11/humans.rdfs-instanc...	http://www.inria.fr/2007/09/11/humans.rdfs#hasFa...
		http://www.inria.fr/2007/09/11/humans.rdfs-instanc...	http://www.inria.fr/2007/09/11/humans.rdfs#hasFr...
		http://www.inria.fr/2007/09/11/humans.rdfs-instanc...	http://www.inria.fr/2007/09/11/humans.rdfs#hasS...

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 plain English.

<ANSWER HERE/>

Find all the resources that has a property “hasSpouse” and their values

Run this query. How many answers do you get?

<ANSWER HERE/>

6 results

?x	?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine

Question 5:

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

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

Query:

```
select * where { ?x h:shoesize ?o. ?x a h:Person }
```

<ANSWER HERE/>

?x	?o
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	14
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark	8
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	10
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	7

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

Query:

```
select * where { ?x a h:Person OPTIONAL {?x h:shoesize ?o} }
```

<ANSWER HERE/>

Graph	XML	Table	Validate
		?x	?o
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			14
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark			8
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#David			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William			10
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl			7

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

Query:

```
select * where { ?x a h:Person . ?x h:shoesize ?size1. ?x h:shirtsize ?size2. filter ( (?size1 > 8) || (?size2 > 12)) }
```

<ANSWER HERE/>

Graph	XML	Table	Validate
		?x	?size1
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	14		12
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	10		13

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 the parents who have at least one child.

Query:

```
select ?x where { ?x h:hasChild ?c}
```

<ANSWER HERE/>

Graph	XML	Table	Validate
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora			

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

<ANSWER HERE/>

```
We have 5 answers, among them Gaston is duplicated
```

2. Find a way to avoid duplicates.

Query:

```
select distinct ?x where { ?x h:hasChild ?c}
```

<ANSWER HERE/>

Graph	XML	Table	Validate	?x
				http://www.inria.fr/2007/09/11/humans.rdfs#Harry
				http://www.inria.fr/2007/09/11/humans.rdfs#Gaston
				http://www.inria.fr/2007/09/11/humans.rdfs#Jack
				http://www.inria.fr/2007/09/11/humans.rdfs#Flora

How many answers do you get then?

<ANSWER HERE/>

Now we have 4 distinct answers

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

Query:

```
select ?x where { ?x a h:Person} minus {?x h:hasChild ?c} }
```

<ANSWER HERE/>

Graph	XML	Table	Validate	?x
				http://www.inria.fr/2007/09/11/humans.rdfs#John
				http://www.inria.fr/2007/09/11/humans.rdfs#Mark
				http://www.inria.fr/2007/09/11/humans.rdfs#Eve
				http://www.inria.fr/2007/09/11/humans.rdfs#David
				http://www.inria.fr/2007/09/11/humans.rdfs#Laura
				http://www.inria.fr/2007/09/11/humans.rdfs#William
				http://www.inria.fr/2007/09/11/humans.rdfs#Karl

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:

```
select * where { ?x h:age ?age filter (?age <19) }
```

<ANSWER HERE/>

Graph	XML	Table	Validate	?x	?age
				http://www.inria.fr/2007/09/11/humans.rdfs#Mark	14
				http://www.inria.fr/2007/09/11/humans.rdfs#Lucas	12

How many answers do you get?

<ANSWER HERE/>

We have 2 kids

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 i: <http://www.inria.fr/2007/09/11/humans-instances.rdfs#>
ask { i:Mark h:age ?age filter (?age >18)}
```

<ANSWER HERE/>

Graph XML Table Validate

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

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 ?age (if ( abs(floor(?age/2) - ceil(?age/2)) =0, TRUE,FALSE) as ?odd)

where { ?x h:age ?age }
```

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2 select ?x ?age (if ( abs(floor(?age/2) - ceil(?age/2)) =0, TRUE,FALSE) as ?odd)
3 where { ?x h:age ?age }
```

Graph XML Table Validate

?x	?age	?odd
http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37	false
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14	true
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Gaston	102	true
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pierre	95	false
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Lucas	71	false
http://www.inria.fr/2007/09/11/humans.rdfsInstances#William	12	true
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	42	true
	36	true

Question 8

1. **Construct** the symmetric of all hasFriend relations using the good SPARQL statement (ex. When finding Thomas hasFriend Fabien, your query should construct Fabien hasFriend Thomas)

Query:

```
construct {?y h:hasFriend ?x } where {?x h:hasFriend ?y}
```

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 construct {?y h:hasFriend ?x } where {?x h:hasFriend ?y}
```

Graph XML Table Validate

?x	?y
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Eve	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Alice
http://www.inria.fr/2007/09/11/humans.rdfsInstances#David	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Alice	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Jack	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Alice
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Laura	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Alice
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Sophie

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

Query:

```
insert {?y h:hasFriend ?x } where {?x h:hasFriend ?y}
```

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 select ?x ?y where {?x h:hasFriend ?y}
```

Graph	XML	Table	Validate																										
		<table border="1"> <thead> <tr> <th>?x</th><th>?y</th></tr> </thead> <tbody> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#John</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#David</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#John</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#David</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice</td></tr> <tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl</td><td>http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie</td></tr> </tbody> </table>	?x	?y	http://www.inria.fr/2007/09/11/humans.rdfs#instances#John	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#David	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#John	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs#instances#David	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie	
?x	?y																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#John	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#David	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#John																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Eve																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs#instances#David																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Jack	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Alice																												
http://www.inria.fr/2007/09/11/humans.rdfs#instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs#instances#Sophie																												

Question 9

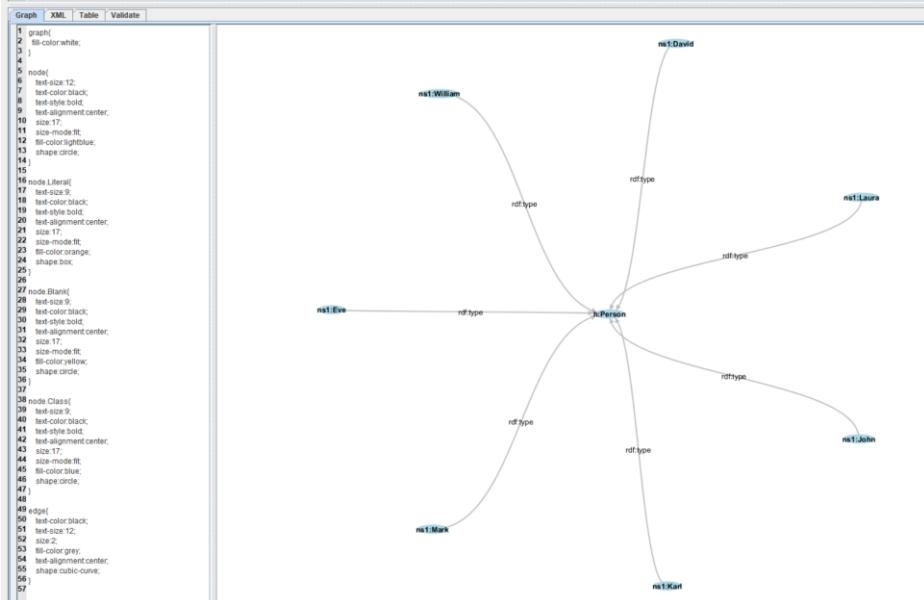
Choose and edit one of the SELECT WHERE queries previously written to transform them into a CONSTRUCT WHERE query (retaining the same WHERE clause) in order to visualize the results as a graph.

Query:

```
construct {?x a h:Person} where { {?x a h:Person} minus {?x h:hasChild ?c} }
```

<ANSWER HERE/>

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



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:

<ANSWER HERE/>

```

1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2 select ?p ?o where { <http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher> ?p ?o }

```

?p	?o
http://www.inria.fr/2007/09/11/humans.rdfs#age	38
http://www.inria.fr/2007/09/11/humans.rdfs#hasAncestor	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher
http://www.inria.fr/2007/09/11/humans.rdfs#hasFather	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher
http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mourad
http://www.inria.fr/2007/09/11/humans.rdfs#hasParent	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher
http://www.inria.fr/2007/09/11/humans.rdfs#hasSpouse	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Hanene
http://www.inria.fr/2007/09/11/humans.rdfs#name	Maher
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize	9
http://www.inria.fr/2007/09/11/humans.rdfs#shoesize	9
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize	42
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs#Person
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs#Animal
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs#Male

Question 11

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

Query:

```
select ?x ?y ?val where {?x h:shirtsize ?val. ?y h:shirtsize ?val filter (?x < ?y)}
```

<ANSWER HERE/>

```

1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 select ?x ?y ?val where {?x h:shirtsize ?val. ?y h:shirtsize ?val filter (?x < ?y)}

```

?x	?y	?val
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pierre	9
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	9
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pierre	9
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher	9
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	9
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Maher	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pierre	9

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

Query:

```
construct {?x h:seeAlso ?y} where {?x h:shirtsize ?val. ?y h:shirtsize ?val filter (?x != ?y)}
```

<ANSWER HERE/>

```

1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 construct {?x h:seeAlso ?y} where {?x h:shirtsize ?val. ?y h:shirtsize ?val filter (?x != ?y)}

```

Graph XML Table Validate

```

2 fill-color:white;
3 }
4
5 node{
6   text-size:12;
7   text-color:black;
8   text-style:bold;
9   text-alignment:center;
10  size:17;
11  size-mode:fit;
12  fill-color:lightblue;
13  shape:circle;
14 }
15
16 node.Literal{
17   text-size:9;
18   text-color:black;
19   text-style:bold;
20   text-alignment:center;
21   size:17;
22   size-mode:fit;
23   fill-color:orange;
24   shape:box;
25 }
26
27 node.Blank{
28   text-size:9;
29   text-color:black;
30   text-style:bold;
31   text-alignment:center;
32   size:17;
33   size-mode:fit;
34   fill-color:yellow;
35   shape:circle;
36 }
37
38 node.Class{
39   text-size:9;
40   text-color:black;
41   text-style:bold;
42   text-alignment:center;
43   size:17;
44   size-mode:fit;
45   fill-color:blue;
46   shape:circle;
47 }
48
49 edge{
50   text-color:black;
51   text-size:12;
52   size:2;
53   fill-color:grey;
54   text-alignment:center;
55   shape:cubic-curve;
56 }
57

```

3. Change the query into an insert.
4. Visualize the resources connected by seeAlso (use the CONSTRUCT clause).

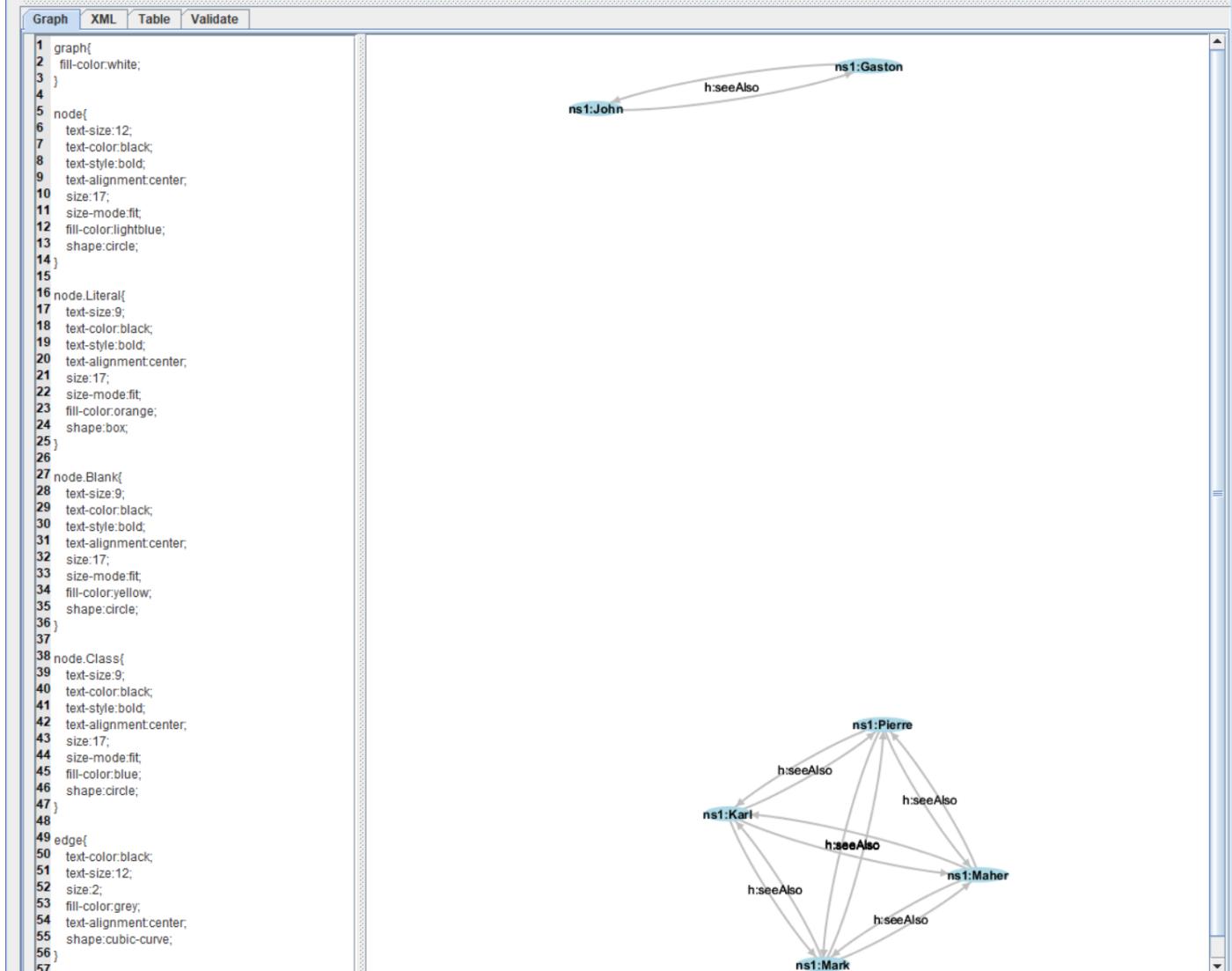
```

insert {?x h:seeAlso ?y} where {?x h:shirtsize ?val. ?y h:shirtsize ?val filter (?x != ?y)}
construct {?x h:seeAlso ?y} where {?x h:seeAlso ?y}

```

screenshot:

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 construct {?x h:seeAlso ?y} where {?x h:seeAlso ?y}
```



<ANSWER HERE/>

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

Query:

```
select ?x ?y ?val where {?x h:shoesize ?val. ?y h:shoesize ?val filter (?x != ?y)}
```

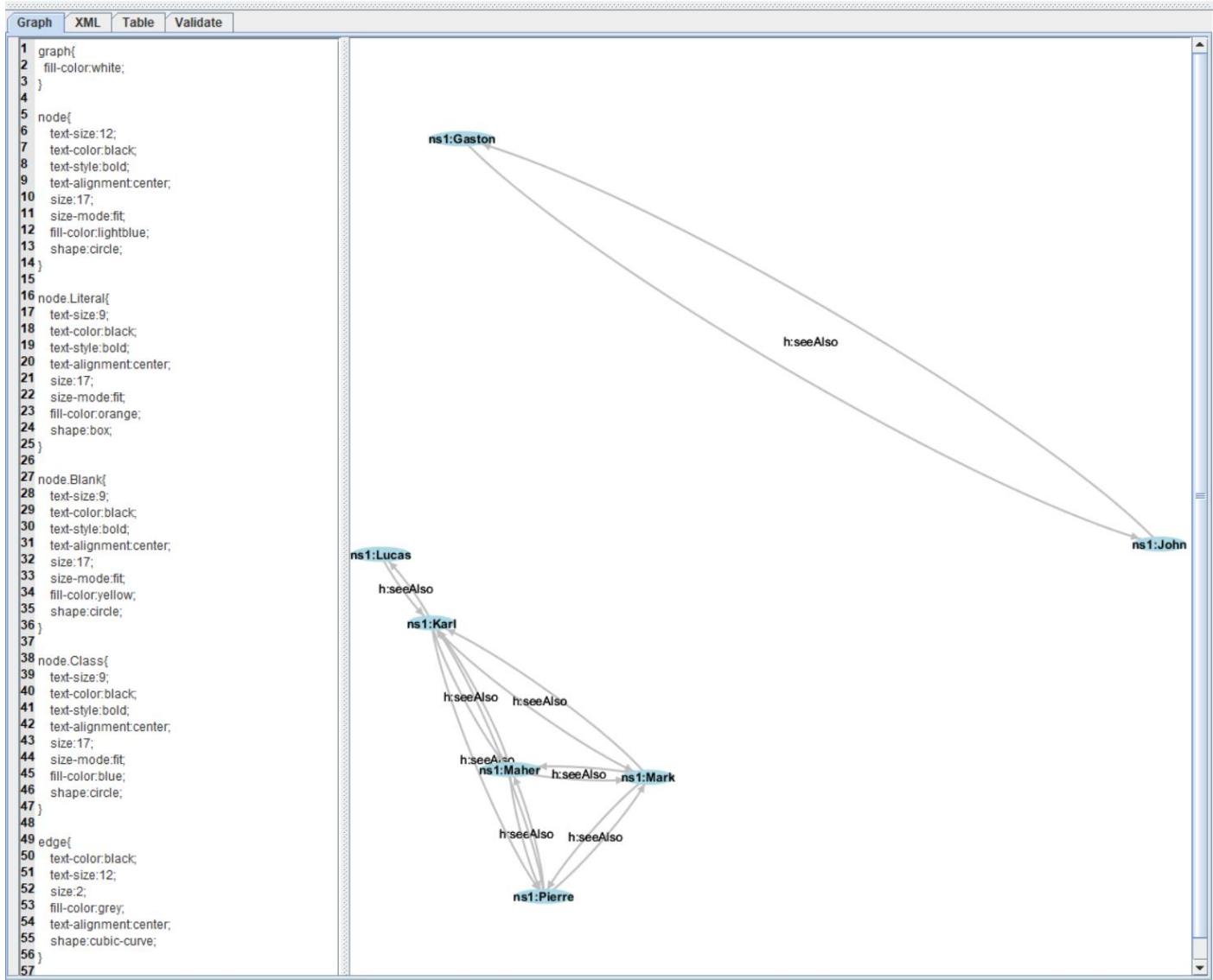
```
insert { ?x h:seeAlso ?y } where {?x h:shoesize ?val. ?y h:shoesize ?val filter (?x != ?y)}
```

<ANSWER HERE/>

6. Visualize the resources connected by seeAlso (use the CONSTRUCT clause)

screenshot:

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 construct {?x h:seeAlso ?y} where {?x h:seeAlso ?y}
```



<ANSWER HERE/>

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

Query:

```
select ?x ?y where { ?x h:seeAlso+ ?y }
```

<ANSWER HERE/>

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

```

1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 construct {?x h:seeAlso ?y} where {?x h:seeAlso ?y }

```

Graph XML Table Validate

```

1 graph{
2   fill-color:white;
3 }
4
5 node{
6   text-size:12;
7   text-color:black;
8   text-style:bold;
9   text-alignment:center;
10  size:17;
11  size-mode:fit;
12  fill-color:lightblue;
13  shape:circle;
14 }
15
16 node.Literal{
17  text-size:9;
18  text-color:black;
19  text-style:bold;
20  text-alignment:center;
21  size:17;
22  size-mode:fit;
23  fill-color:orange;
24  shape:box;
25 }
26
27 node.Blank{
28  text-size:9;
29  text-color:black;
30  text-style:bold;
31  text-alignment:center;
32  size:17;
33  size-mode:fit;
34  fill-color:yellow;
35  shape:circle;
36 }
37

```

Question 12

- Find the largest shoe size

Query:

```
select (max(?val) as ?max ) where { ?x h:shoesize ?val }
```

<ANSWER HERE/>

```

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

```

Graph XML Table Validate

?max

14

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

Query:

```
select ?senior ?max where { {select (max(?val) as ?max) where { ?person h:shoesize ?val } } ?senior h:shoesize ?max. }
```

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2
3 select ?senior ?max where { {select (max(?val) as ?max) where { ?person h:shoesize ?val } } ?senior h:shoesize ?max. }
4
```

Graph	XML	Table	Validate
?senior http://www.inria.fr/2007/09/11/humans.rdfs-instances#John		?max 14	

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

Query:

```
select (avg(?val) as ?average )where { ?x h:shoesize ?val}
```

<ANSWER HERE/>

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

Graph	XML	Table	Validate
		?average 9.25	

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

Query:

<ANSWER HERE/>

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

Graph	XML	Table	Validate
		?average 9.25	

Question 13

Find couples without children

Query:

```
select ?x ?y where {?x h:hasSpouse ?y minus {{?x h:hasChild ?z} union {?y h:hasChild ?u}} }
```

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2 select ?x ?y where {?x h:hasSpouse ?y minus {?x h:hasChild ?z} union {?y h:hasChild ?u}} }
```

Graph	XML	Table	Validate
?x			?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve			http://www.inria.fr/2007/09/11/humans.rdfs-instances#David
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer			http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene

Question 14

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

Insert:

<ANSWER HERE/>

```
insert DATA {<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Adam> a h:Person; h:shoesize 32}
```

Screenshot result:

<ANSWER HERE/>

```
1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2 select ?p ?o where <http://www.inria.fr/2007/09/11/humans.rdfs-instances#Adam> ?p ?o
3
```

?p	?o
http://www.inria.fr/2007/09/11/humans.rdfs#shoesize	32
http://www.w3.org/1999/02/22-rdf-syntax-ns#type	http://www.inria.fr/2007/09/11/humans.rdfs#Person

Question 15

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

query:

```
select ?x ?y where { ?x (h:hasSpouse | h:hasChild | h:hasFather | h:hasMother | h:hasParent)+ ?y }
```

```
construct {?x h:seeAlso ?y } where { ?x (h:hasSpouse | h:hasChild | h:hasFather | h:hasMother | h:hasParent)+ ?y }
```

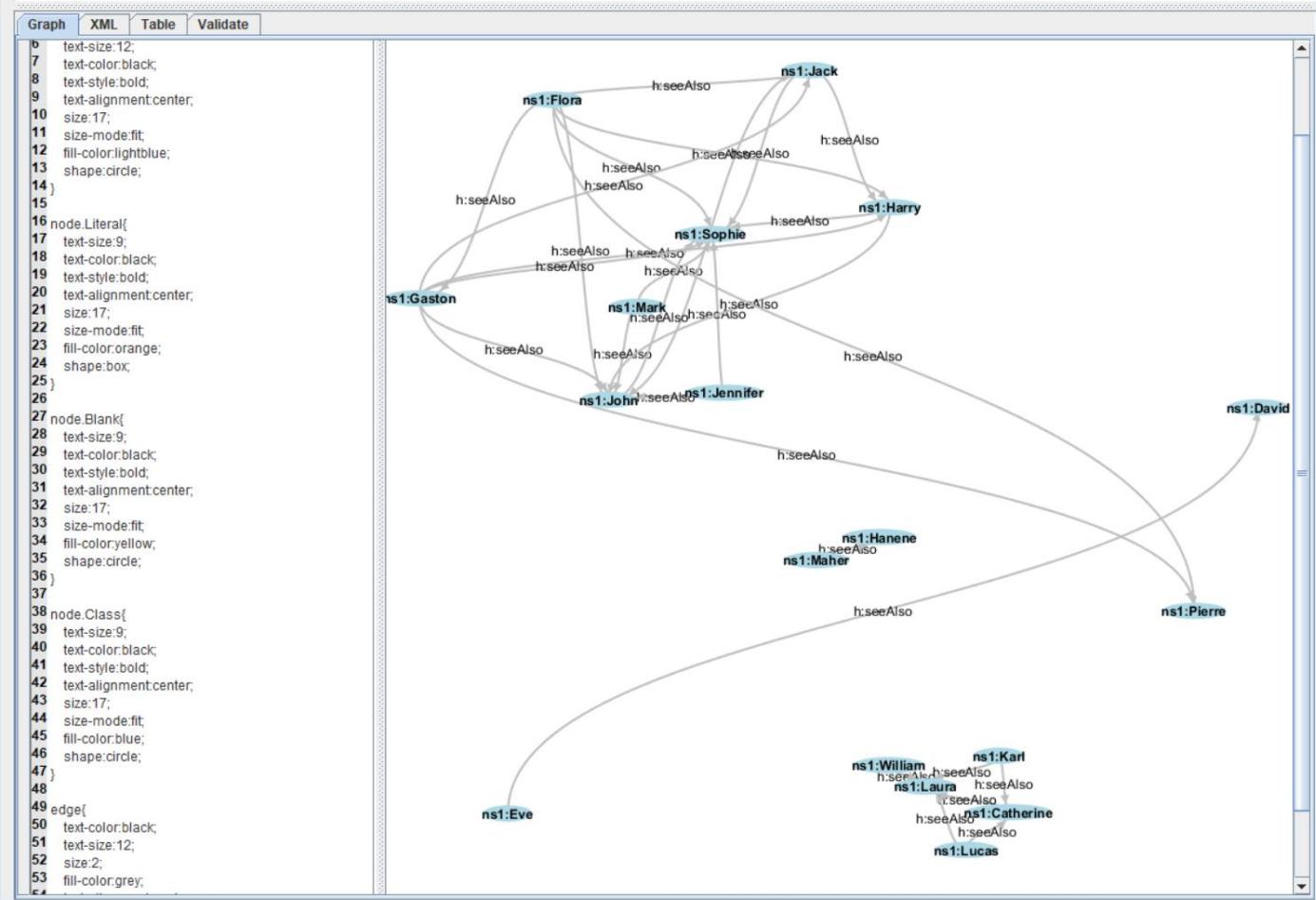
screenshot:

<ANSWER HERE/>

```

1 prefix h: <http://www.inria.fr/2007/09/11/humans.rdfs#>
2 construct { ?x h:seeAlso ?y } where { ?x (h:hasSpouse | h:hasChild | h:hasFather | h:hasMother | h:hasParent)+ ?y }
3

```



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

The query connects to the French dbpedia sparql endpoint, retrieves 20 first triples that form spouse couples along with their respective names. Then the query constructs a local RDF graph

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

query:

```
insert { ?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
  }
}
```

<ANSWER HERE/>

Rq: the insertion is performed on the local RDF, not the dbpedia distant one, otherwise the insertion would assumed to trigger a failure

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

<ANSWER HERE/>

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

<ANSWER HERE/>

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

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
@prefix : <http://inria.fr/devices#>
:Phone rdfs:subClassOf :Device .
:Computer rdfs:subClassOf :Device .
:Smartphone rdfs:subClassOf :Computer .
:Smartphone rdfs:subClassOf :Phone .
```

<ANSWER HERE/>

By transitivity: smartphone rdfs:subClassOf :Device

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

```
@prefix rdfs: < http://www.w3.org/2000/01/rdf-schema# >
@prefix : <http://inria.fr/member#>
:employeeOf rdfs:subPropertyOf :proRelationWith .
:hasControlOver rdfs:subPropertyOf :proRelationWith .
:isShareholderOf rdfs:subPropertyOf :hasControlOver .
:isCEOof rdfs:subPropertyOf :employeeOf, :hasControlOver .
```

<ANSWER HERE/>

```
:isCEOof rdfs:subPropertyOf :proRelationWith  
:isShareholderOf rdfs:subPropertyOf :proRelationWith
```

Q4.5. What can be said about the types of the resources that will be linked by the properties defined below?

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
@prefix : <http://inria.fr/humans#>  
:driverOf rdfs:subPropertyOf :isControling .  
:piloteOf rdfs:subPropertyOf :isControling .  
:isControling rdfs:domain :Human ; rdfs:range :Object .  
:driverOf rdfs:range :Car .  
:piloteOf rdfs:domain :Adult ; rdfs:range :Plane .
```

<ANSWER HERE/>

All of them will apply

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

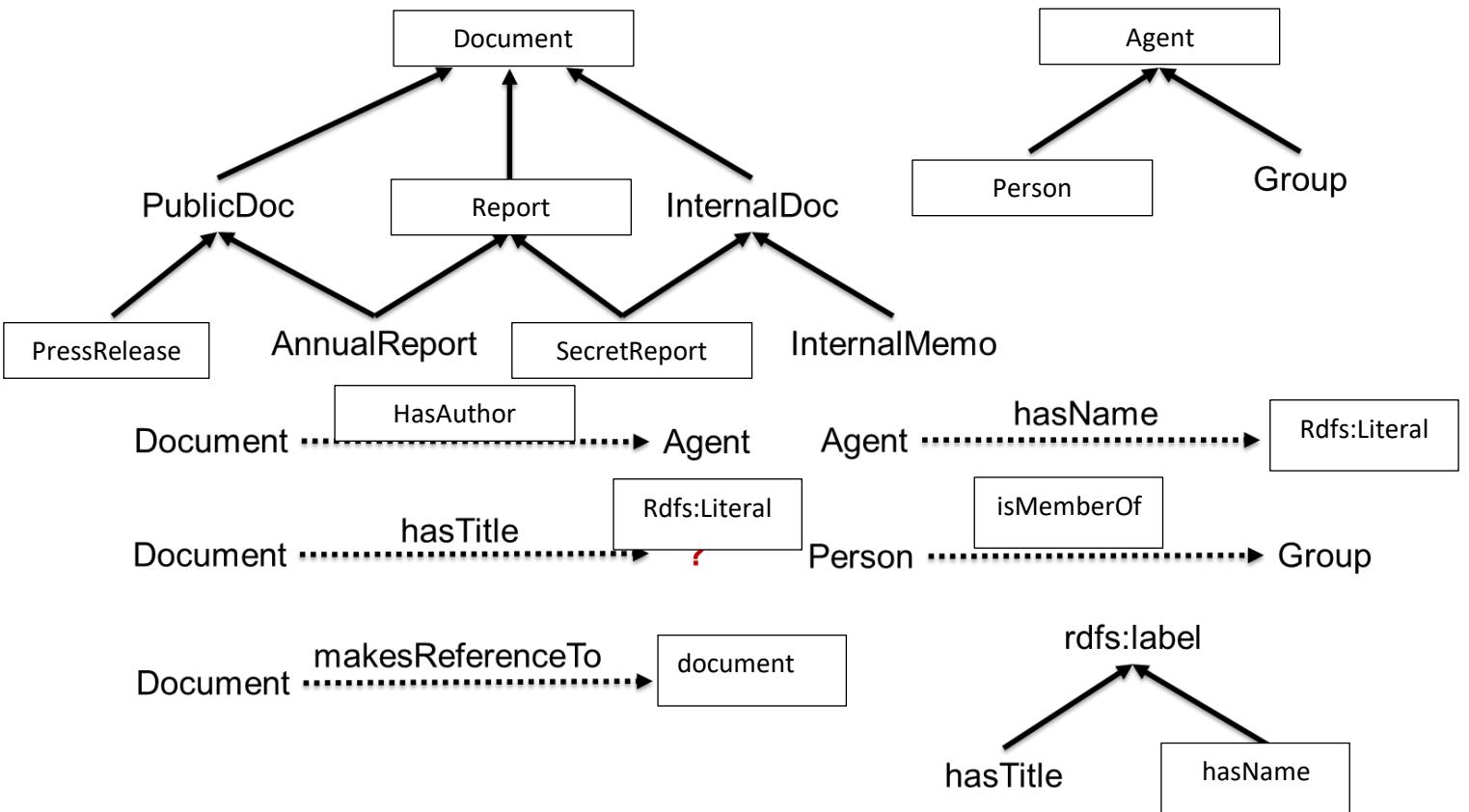
```
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
@base <http://inria.fr/2005/humans.rdfs>  
<p1> a rdf:Property ; rdfs:label "age"@fr .  
<c1> a rdfs:Class; rdfs:comment "un être humain"@fr .
```

- <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 .

<ANSWER HERE/>

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.

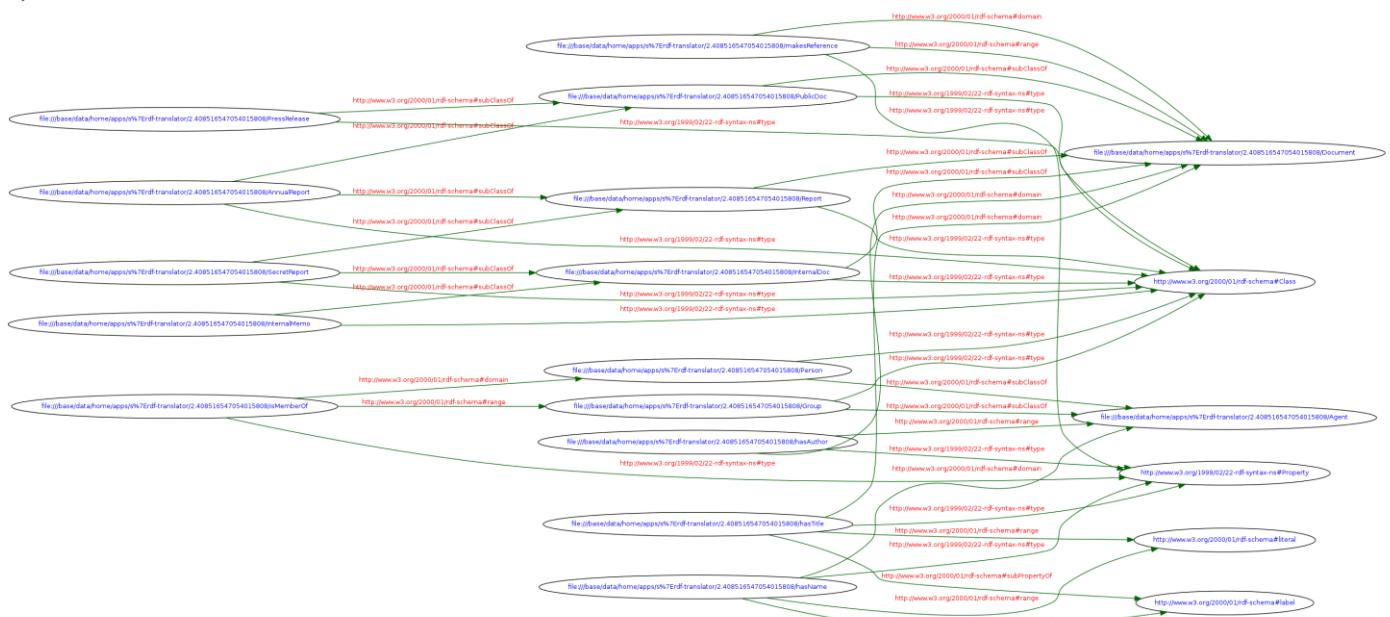


```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
2 @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
3 @base <http://base/data/home/apps/v%7Erdf-translator/2.409516547054015808/>
4 <PressRelease> a rdfs:Class ; rdfs:subClassOf <PublicDoc> .
5 <AnnualReport> a rdfs:Class ; rdfs:subClassOf <PublicDoc>, <Report> .
6 <SecretReport> a rdfs:Class ; rdfs:subClassOf <InternalDoc>, <Report> .
7 <InternalMemo> a rdfs:Class ; rdfs:subClassOf <InternalDoc> .
8 <PublicDoc> a rdfs:Class ; rdfs:subClassOf <Document> .
9 <Report> a rdfs:Class ; rdfs:subClassOf <Document> .
10 <InternalDoc> a rdfs:Class ; rdfs:subClassOf <Document> .
11 <Person> a rdfs:Class ; rdfs:subClassOf <Agent> .
12 <Group> a rdfs:Class ; rdfs:subClassOf <Agent> .
13
14 <hasAuthor> a rdf:Property ; rdfs:domain <Document> ; rdfs:range <Agent> .
15 <hasTitle> a rdf:Property ; rdfs:subPropertyOf rdfs:label ; rdfs:domain <Document> ; rdfs:range rdfs:literal .
16 <hasName> a rdf:Property ; rdfs:subPropertyOf rdfs:label ; rdfs:domain <Agent> ; rdfs:range rdfs:literal .
17 <makesReferenceTo> a rdf:Property ; rdfs:domain <Document> ; rdfs:range <Document> .
18 <isMemberOf> a rdf:Property ; rdfs:domain <Person> ; rdfs:range <Group> .

```

Graph of the data model



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 Jen 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 may have to update the files Jen.rdf and Jen.ttl to use your ontology.

Your schema:

```
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .  
@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .  
@prefix voc: <http://www.unice.fr/voc#> .  
@base <http://www.unice.fr/voc> .  
  
voc:Human a rdfs:Class .  
voc:FamilyMember a rdfs:Class .  
voc:Man rdfs:subClassOf voc:Human .  
voc:Woman rdfs:subClassOf voc:Human .  
  
voc:name a rdf:Property ; rdfs:subPropertyOf rdfs:Literal .  
voc:familyLink a rdf:Property ; rdfs:domain voc:FamilyMember; rdfs:range  
voc:FamilyMember.  
voc:hasSpouse a rdf:Property; rdfs:subPropertyOf voc:familyLink.  
voc:hasFather a rdf:Property .  
voc:shoeSize a rdf:Property ; rdfs:subPropertyOf rdfs:Literal .  
voc:hasColleague a rdf:Property .  
voc:trouserSize a rdf:Property ; rdfs:subPropertyOf rdfs:Literal .  
voc:age a rdf:Property ; rdfs:subPropertyOf rdfs:Literal .  
voc:hasChild a rdf:Property; rdfs:subPropertyOf voc:familyLink .
```

<ANSWER HERE/>

- 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 Jen.rdf
- The interface contains a default SPARQL query:
Select ?x ?t where {?x rdf:type ?t}
Launch the query and look at the results.

Screenshot:

<ANSWER HERE/>

1 Select ?x ?t where {?x rdf:type ?t}			
Graph	XML	Table	Validate
?x		?t	
http://www.w3.org/1999/02/22-rdf-syntax-ns#type		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/data#Thomas		http://www.unice.fr/voc#Man	
http://www.unice.fr/voc#name		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/data#Fabien		http://www.unice.fr/voc#Man	
http://www.unice.fr/data#Jen		http://www.unice.fr/voc#Woman	
http://www.unice.fr/voc#hasCollegue		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/voc#trouserSize		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/data#Catherine		http://www.unice.fr/voc#Woman	
http://www.unice.fr/voc#shoeSize		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/voc#hasSpouse		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/data#Seb		http://www.unice.fr/voc#Man	
http://www.unice.fr/voc#age		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/voc#hasChild		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/data#Stefan		http://www.unice.fr/voc#Man	
http://www.w3.org/1999/02/22-rdf-syntax-ns#Type		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	
http://www.unice.fr/voc#hasFather		http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	

- Modify your ontology to declare the classes of Man and Woman as sub classes of Human (don't change the data), reload the schemas and data and search for the humans to see the results

Screenshot:

2	prefix : < http://www.unice.fr/voc# >
3	
4	select * where {
5	?x a :Human
6	}
?	
http://www.unice.fr/data#Thomas	
http://www.unice.fr/data#Fabien	
http://www.unice.fr/data#Jen	
http://www.unice.fr/data#Catherine	
http://www.unice.fr/data#Seb	
http://www.unice.fr/data#Stefan	

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

Screenshot:

```

1 prefix voc: <http://www.unice.fr/voc#>
2
3 select * where { ?x voc:familyLink ?y}
4

```

Graph	XML	Table	Validate
?x			?y
http://www.unice.fr/data#Jen			http://www.unice.fr/data#Seb
http://www.unice.fr/data#Jen			http://www.unice.fr/data#Anny
http://www.unice.fr/data#Jen			http://www.unice.fr/data#Stefan
http://www.unice.fr/data#Seb			http://www.unice.fr/data#Anny
http://www.unice.fr/data#Seb			http://www.unice.fr/data#Stefan

- 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:

<ANSWER HERE/>

```

1 prefix voc: <http://www.unice.fr/voc#>
2
3 select * where { ?x a voc:FamilyMember}
4

```

Graph	XML	Table	Validate
?x			
http://www.unice.fr/data#Jen			
http://www.unice.fr/data#Seb			
http://www.unice.fr/data#Anny			
http://www.unice.fr/data#Stefan			

About the human.rdfs schema

1. If you don't have the human schema file yet, 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?

<ANSWER HERE/>

the namespace associated with the ontology is associated with the base. Association done via

xml:base="<http://www.inria.fr/2007/09/11/humans.rdfs>"

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?

<ANSWER HERE/>

the namespace of the RDFS language is associated to namespace

`xmlns:rdfs="http://www.w3.org/2000/01/rdf-schema#"`

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

<ANSWER HERE/> the age property does not have a signature (no domain, nor range), so any resource can use this property

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:

<ANSWER HERE/>

Graph of the data model



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: select ?x where {?x a rdfs:Class}`

<ANSWER HERE/>

1	<code>select ?x where {?x a rdfs:Class}</code>
2	

Graph XML Table Validate

?x

```

http://www.inria.fr/2007/09/11/humans.rdfs#Animal
http://www.inria.fr/2007/09/11/humans.rdfs#Male
http://www.inria.fr/2007/09/11/humans.rdfs#Female
http://www.inria.fr/2007/09/11/humans.rdfs#Man
http://www.inria.fr/2007/09/11/humans.rdfs#Person
http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer
http://www.inria.fr/2007/09/11/humans.rdfs#Researcher
http://www.inria.fr/2007/09/11/humans.rdfs#Woman

```

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

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

<ANSWER HERE/>

1 select ?x ?y where {?x rdfs:subClassOf ?y}

2

Graph	XML	Table	Validate
?			?
http://www.inria.fr/2007/09/11/humans.rdfs#Male		http://www.inria.fr/2007/09/11/humans.rdfs#Animal	
http://www.inria.fr/2007/09/11/humans.rdfs#Female		http://www.inria.fr/2007/09/11/humans.rdfs#Animal	
http://www.inria.fr/2007/09/11/humans.rdfs#Man		http://www.inria.fr/2007/09/11/humans.rdfs#Male	
http://www.inria.fr/2007/09/11/humans.rdfs#Man		http://www.inria.fr/2007/09/11/humans.rdfs#Person	
http://www.inria.fr/2007/09/11/humans.rdfs#Person		http://www.inria.fr/2007/09/11/humans.rdfs#Animal	
http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer		http://www.inria.fr/2007/09/11/humans.rdfs#Person	
http://www.inria.fr/2007/09/11/humans.rdfs#Researcher		http://www.inria.fr/2007/09/11/humans.rdfs#Person	
http://www.inria.fr/2007/09/11/humans.rdfs#Woman		http://www.inria.fr/2007/09/11/humans.rdfs#Female	
http://www.inria.fr/2007/09/11/humans.rdfs#Woman		http://www.inria.fr/2007/09/11/humans.rdfs#Person	

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: describe <http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>

answers:

<ANSWER HERE/>

1 describe <http://www.inria.fr/2007/09/11/humans.rdfs#shoesize>

2

?_ast_v_0	?_ast_p_0
http://www.w3.org/1999/02/22-rdf-syntax-ns#Property	http://www.w3.org/1999/02/22-rdf-syntax-ns#type
"express in some way the approximate length of the shoes for a person."@en	http://www.w3.org/2000/01/rdf-schema#comment
"taille, exprimée en points, des chaussures d'une personne."@fr	http://www.w3.org/2000/01/rdf-schema#comment
http://www.inria.fr/2007/09/11/humans.rdfs#Person	http://www.w3.org/2000/01/rdf-schema#domain
"shoe size"@en	http://www.w3.org/2000/01/rdf-schema#label
"size"@en	http://www.w3.org/2000/01/rdf-schema#label
"pointure"@fr	http://www.w3.org/2000/01/rdf-schema#label

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 rdfs: <http://www.w3.org/2000/01/rdf-schema#>

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

answers:

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 select * where {?x rdfs:label "personne"@fr. ?x rdfs:label ?y filter (lang(?y) = "fr" )}
3

```

Graph	XML	Table	Validate
?			?
http://www.inria.fr/2007/09/11/humans.rdfs#Person		"homme"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#Person		"personne"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#Person		"être humain"@fr	
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 rdfs: <http://www.w3.org/2000/01/rdf-schema#>

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

```

answers:

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 select * where {?x rdfs:label "size"@en . ?x rdfs:label ?y}

```

Graph	XML	Table	Validate
?			?
http://www.inria.fr/2007/09/11/humans.rdfs#shoysize		"shoe size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#shoysize		"size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#shoysize		"pointure"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize		"size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize		"shirt size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize		"taille de chemise"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize		"taille"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize		"size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize		"taille"@fr	
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize		"trousers size"@en	
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize		"taille de pantalon"@fr	

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

query:

```

@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>

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

select * where { ?x rdfs:domain :Person } union { ?y rdfs:range :Person } 

```

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where { ?x rdfs:domain :Person} union {?y rdfs:range :Person} 
```

Graph	XML	Table	Validate
?x			?y
http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend			
http://www.inria.fr/2007/09/11/humans.rdfs#shoeshize			
http://www.inria.fr/2007/09/11/humans.rdfs#shirtszie			
http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize			
http://www.inria.fr/2007/09/11/humans.rdfs#hasSpouse			
	http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend		
	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:

<ANSWER HERE/>

```

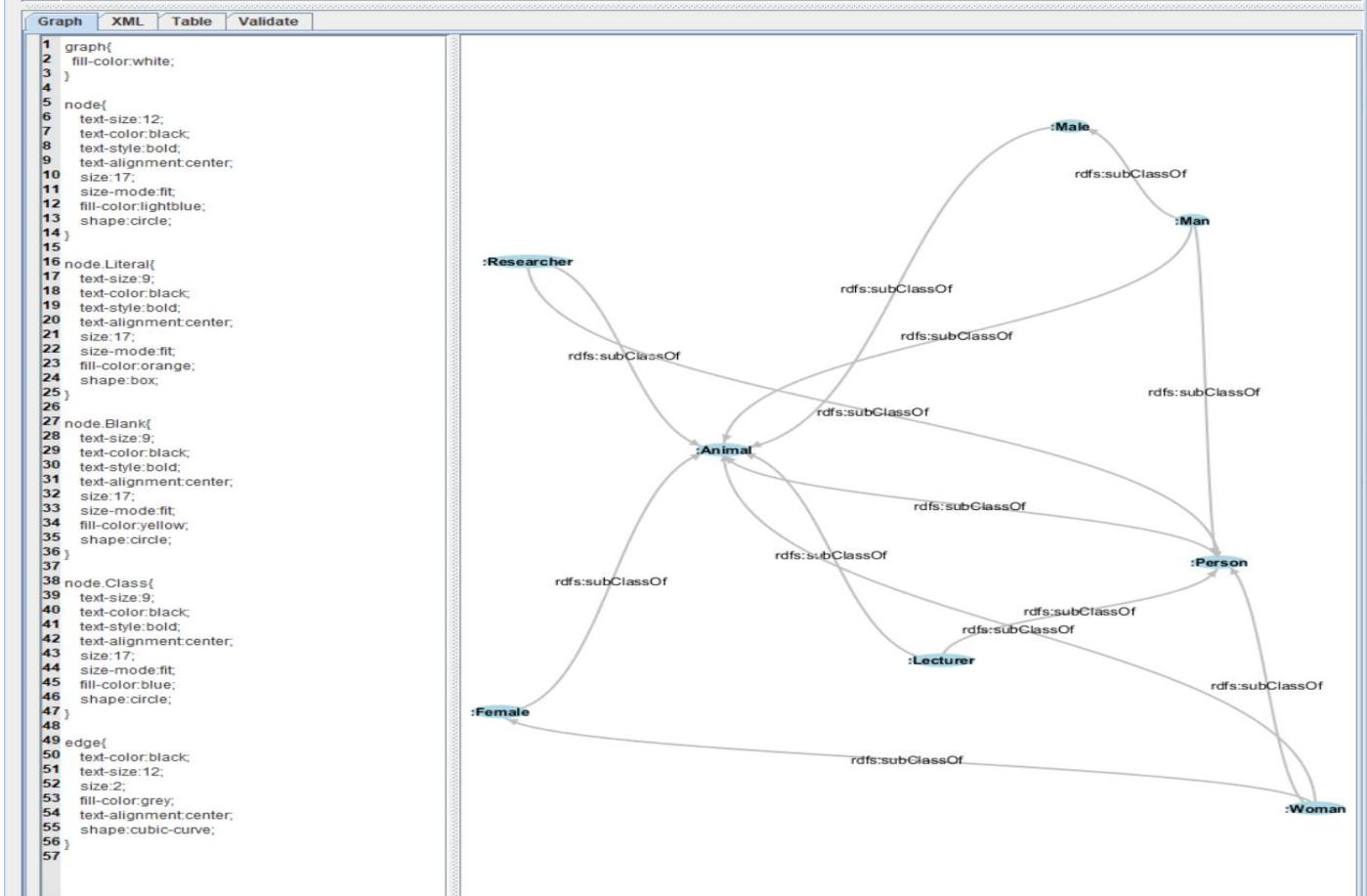
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
@prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
construct {?x rdfs:subClassOf ?y} where {?x (rdfs:subClassOf)+ ?y } 
```

screenshot:

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 construct {?x rdfs:subClassOf ?y} where {?x (rdfs:subClassOf)+ ?y } 
```



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

query:

<ANSWER HERE/>

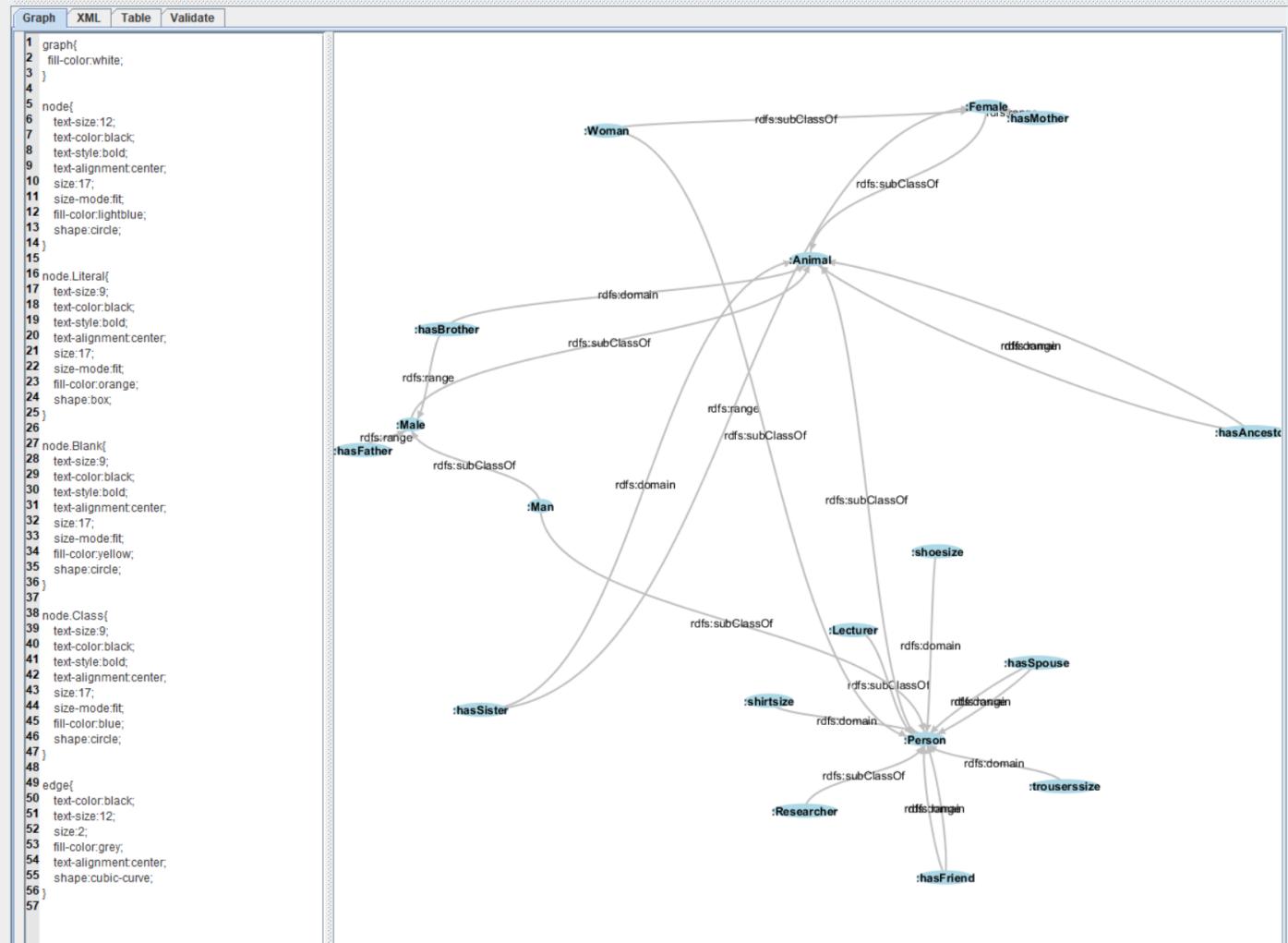
```
construct {?x rdfs:subClassOf ?y. ?z rdfs:domain ?dom. ?z rdfs:range ?range}
```

```
where { {?x rdfs:subClassOf ?y } union {?z rdfs:domain ?dom.} union { ?z rdfs:range ?range} }
```

screenshot:

<ANSWER HERE/>

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdf#>
3 construct {?x rdfs:subClassOf ?y. ?z rdfs:domain ?dom. ?z rdfs:range ?range}
4 where { {?x rdfs:subClassOf ?y } union {?z rdfs:domain ?dom.} union { ?z rdfs:range ?range} }
```



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:

```
select * where {?x a :Person}
```

Number of results before:

<ANSWER HERE/>

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Person}
```

Graph XML Table Validate

?x

http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve
http://www.inria.fr/2007/09/11/humans.rdfs-instances#David
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher

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

New number of results after and your explanation:

<ANSWER HERE/>

The loading of the schema enabled the inferencing and allowed to propagate the type :Person to new entities

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Person}
```

Graph XML Table Validate

?x

http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve
http://www.inria.fr/2007/09/11/humans.rdfs-instances#David
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mourad

Question 2

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

```
Query: select * where {?x a :Male. ?x :hasSpouse ?y}
```

<ANSWER HERE/>

Number of results: **2 results**

<ANSWER HERE/>

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Male. ?x :hasSpouse ?y}
```

Graph	XML	Table	Validate

?

?x	?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene

2. In the data declare that Lucas has to father Karl. Reset Coarse, 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:

<ANSWER HERE/>

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

Number of results and your explanation:

```
now we have 3. Adding Karl as father will infer that karl is Male and then Karl and his spouse Catherine is added to the result set
```

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Male. ?x :hasSpouse ?y}
```

Graph	XML	Table	Validate

?

?x	?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene

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:

```
select * where {?x a :Lecturer. ?x a ?y}
```

<ANSWER HERE/>

Number of results and your explanation:

7 results. Lecturer is a subclass of Person. Eve and Laura are the only Lecturer, but Eve and Laura have other types due to inferencing of the ontology so we will output all the types associated with Eve and Laura

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Lecturer. ?x a ?y}
```

Graph	XML	Table	Validate
?x		?y	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve		http://www.inria.fr/2007/09/11/humans.rdfs#Person	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve		http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura		http://www.inria.fr/2007/09/11/humans.rdfs#Person	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura		http://www.inria.fr/2007/09/11/humans.rdfs#Lecturer	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura		http://www.inria.fr/2007/09/11/humans.rdfs#Researcher	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura		http://www.inria.fr/2007/09/11/humans.rdfs#Animal	
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura		http://www.inria.fr/2007/09/11/humans.rdfs#Female	

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:

```
select * where {?x a :Person. ?x a :Male}
```

<ANSWER HERE/>

Your explanation of the result:

jack is originally of Type Man, but Man is a sub Class of Male and also Person, so it is inferred that Jack is Male AND Person so it is included in the result set

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x a :Person. ?x a :Male}
```

Graph	XML	Table	Validate
?x			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl			
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			

Question 4

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

Query:

```
select * where {?x :hasAncestor ?y}
```

<ANSWER HERE/>

Your explanation of the result:

The property hasParent is a sub Property of hasAncestor, and HasFather and hasMother are sub Properties of hasParent. Thus the shown inferred results

<ANSWER HERE/>

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {?x :hasAncestor ?y}
```

Graph	XML	Table	Validate														
		<table><thead><tr><th>?x</th><th>?y</th></tr></thead><tbody><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#John</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#John</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher</td></tr></tbody></table>	?x	?y	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	
?x	?y																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher																

Question 5

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

Query:

```
select * where {{?x (:hasSpouse | :hasChild )+ ?y} }
```

<ANSWER HERE/>

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

Query:

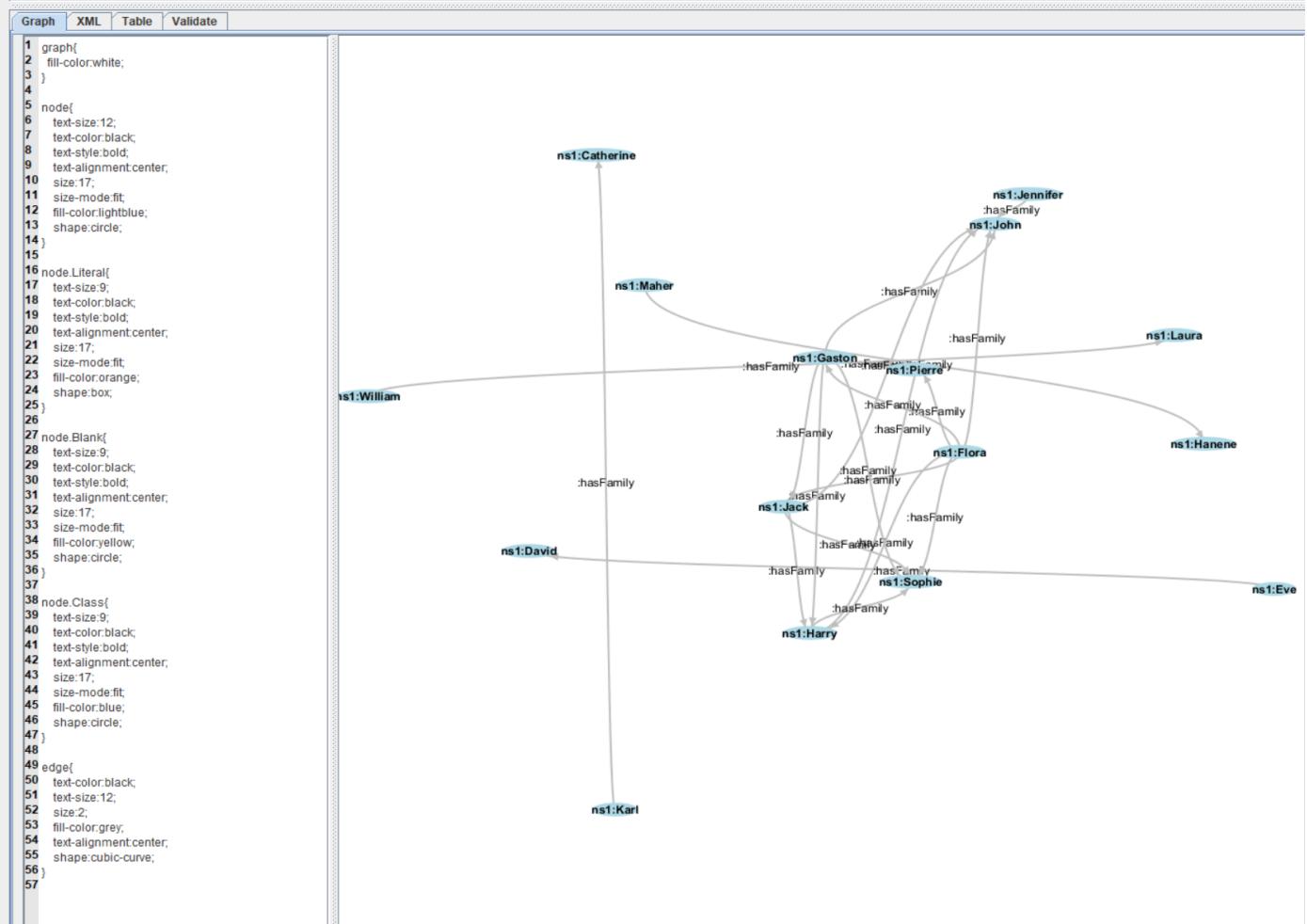
```
construct {?x :hasFamily ?y } where {{?x (:hasSpouse | :hasChild )+ ?y} }
```

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 construct {?x :hasFamily ?y} where {(?x (:hasSpouse | :hasChild )+ ?y) }

```



Question 6

- 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:

<ANSWER HERE/>

```

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

```

Query:

```
construct {?x :olderThan ?y} where {?x :age ?val1 ?y :age ?val2. filter (?val1 > ?val2) }
```

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 construct {?x :olderThan ?y} where {?x :age ?val1 ?y :age ?val2. filter (?val1> ?val2) }

```

Graph	XML	Table	Validate
?x	?val1	?y	?val2
http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc.	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc.	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc.	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42
http://www.inria.fr/2007/09/11/humans.rdfsInstances#G...	102	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc...	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Flora	95	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc...	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Pie...	71	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc...	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc...	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Will...	42	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38	http://www.inria.fr/2007/09/11/humans.rdfsInstances#John	37
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Mark	14
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Luc...	12
http://www.inria.fr/2007/09/11/humans.rdfsInstances#Ma...	38	http://www.inria.fr/2007/09/11/humans.rdfsInstances#Karl	36

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

Query:

```
select * where {?x :olderThan{0,1} ?y }
```

Question 7

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

Query:

```
select * where {<http://www.inria.fr/2007/09/11/humans.rdfsInstances#John> ?p ?o. ?p rdfs:label ?val. filter (contains(?val, "size")) }
```

<ANSWER HERE/>

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3 select * where {<http://www.inria.fr/2007/09/11/humans.rdfsInstances#John> ?p ?o. ?p rdfs:label ?val. filter (contains(?val, "size")) }

```

Graph	XML	Table	Validate
?o	?p	?val	
12	http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize	"size"@en	
12	http://www.inria.fr/2007/09/11/humans.rdfs#shirtsize	"shirt size"@en	
14	http://www.inria.fr/2007/09/11/humans.rdfs#shoesize	"shoe size"@en	
14	http://www.inria.fr/2007/09/11/humans.rdfs#shoesize	"size"@en	
44	http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize	"size"@en	
44	http://www.inria.fr/2007/09/11/humans.rdfs#trouserssize	"trousers size"@en	

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:

```
select ?prop ?obj ?p ?val where {<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura> ?prop ?obj. ?prop ?p ?val . filter (lang(?val)="fr") }
```

<ANSWER HERE/>

1	@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>																								
2	@prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>																								
3	select ?prop ?obj ?p ?val where {<http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura> ?prop ?obj. ?prop ?p ?val . filter (lang(?val)="fr") }																								
<hr/>																									
<table border="1"><thead><tr><th>Graph</th><th>XML</th><th>Table</th><th>Validate</th></tr></thead><tbody><tr><th>?prop</th><th>?obj</th><th>?p</th><th>?val</th></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice</td><td>http://www.w3.org/2000/01/rdf-schema#comment</td><td>"relation entre une personne et une autre personne qui est l'objet d...@fr</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice</td><td>http://www.w3.org/2000/01/rdf-schema#label</td><td>"a pour ami"@fr</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#name</td><td>Laura</td><td>http://www.w3.org/2000/01/rdf-schema#comment</td><td>"désignation de quelque chose."@fr</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs#name</td><td>Laura</td><td>http://www.w3.org/2000/01/rdf-schema#label</td><td>"nom"@fr</td></tr></tbody></table>		Graph	XML	Table	Validate	?prop	?obj	?p	?val	http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice	http://www.w3.org/2000/01/rdf-schema#comment	"relation entre une personne et une autre personne qui est l'objet d...@fr	http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice	http://www.w3.org/2000/01/rdf-schema#label	"a pour ami"@fr	http://www.inria.fr/2007/09/11/humans.rdfs#name	Laura	http://www.w3.org/2000/01/rdf-schema#comment	"désignation de quelque chose."@fr	http://www.inria.fr/2007/09/11/humans.rdfs#name	Laura	http://www.w3.org/2000/01/rdf-schema#label	"nom"@fr
Graph	XML	Table	Validate																						
?prop	?obj	?p	?val																						
http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice	http://www.w3.org/2000/01/rdf-schema#comment	"relation entre une personne et une autre personne qui est l'objet d...@fr																						
http://www.inria.fr/2007/09/11/humans.rdfs#hasFriend	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Alice	http://www.w3.org/2000/01/rdf-schema#label	"a pour ami"@fr																						
http://www.inria.fr/2007/09/11/humans.rdfs#name	Laura	http://www.w3.org/2000/01/rdf-schema#comment	"désignation de quelque chose."@fr																						
http://www.inria.fr/2007/09/11/humans.rdfs#name	Laura	http://www.w3.org/2000/01/rdf-schema#label	"nom"@fr																						

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 .
```

<ANSWER HERE/>

Jane a ex:Woman

John a ex: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 .
```

<ANSWER HERE/>

Jim a ex:Grand Father

We can't say anything about Jack

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 .
```

<ANSWER HERE/>

Jane hasParent Jim

John hasSpouse John

Jane has Ancestor James

Jim has Ancestor James

James hasChild Jim

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 .
```

<ANSWER HERE/>

JimmyHendrix owl:differentFrom JamesPatrickPage

jimmyPage a foaf:Person

JamesPatrickPage a foaf:Person

JamesPatrickPage a ex:Human

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 ]  
    )  
]
```

<ANSWER HERE/>

If you are a Person and in the complement of Lucky then you are an unluckyPerson

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 .
```

<ANSWER HERE/>

Because of the restriction, James and Janes become 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) .
```

<ANSWER HERE/>

```
Ex:A ex:b ex:c ex:PersonList
Ex:a a ex:Person
Ex:b ex:c a ex:PersonList
Ex:b a ex:Person
Ex:c a:PersonList
Ex:c a: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 .
```

<ANSWER HERE/>

The system will conclude ex:James and ex:Jimmy are the same person

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 .
```

<ANSWER HERE/>

Ex:Jim becomes 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 is a symmetrical property and do the same for and hasFriend .

Code added to the schema:

<ANSWER HERE/>

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

Before change

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>  
3  
4 select * where { ?x :hasSpouse ?y }  
5
```

Graph	XML	Table	Validate																
		<table border="1"><thead><tr><th>?x</th><th>?y</th></tr></thead><tbody><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#David</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#John</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#William</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene</td></tr></tbody></table>	?x	?y	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene	
?x	?y																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura																		
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene																		

After change

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>  
3  
4 select * where { ?x :hasSpouse ?y }
```

Graph	XML	Table	Validate																														
		<table border="1"><thead><tr><th>?x</th><th>?y</th></tr></thead><tbody><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#John</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#David</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#David</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#William</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#John</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#William</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene</td></tr><tr><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene</td><td>http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher</td></tr></tbody></table>	?x	?y	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	
?x	?y																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve	http://www.inria.fr/2007/09/11/humans.rdfs-instances#David																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#David	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Eve																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura	http://www.inria.fr/2007/09/11/humans.rdfs-instances#William																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jennifer	http://www.inria.fr/2007/09/11/humans.rdfs-instances#John																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#William	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene																																
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Hanene	http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher																																

2. Declare that `hasChild` is the inverse property of the `hasParent` property.

Code added to the schema:

<ANSWER HERE/>

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

Before change

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3
4 select * where { ?x :hasParent ?y }
5
```

Graph	XML	Table	Validate
		?x	?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark			http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher

After change

```
1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3
4 select * where { ?x :hasParent ?y }
```

Graph	XML	Table	Validate
		?x	?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark			http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher

3. Declare `hasAncestor` as transitive property.

Code added to the schema:

<ANSWER HERE/>

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

Before change

```

2 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
3 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
4
5 select * where { ?x :hasAncestor ?y }

```

Graph	XML	Table	Validate
?x			?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark			http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher

After change

```

1 @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#>
2 @prefix : <http://www.inria.fr/2007/09/11/humans.rdfs#>
3
4 select * where { ?x :hasAncestor ?y }

```

Graph	XML	Table	Validate
?x			?y
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Harry
http://www.inria.fr/2007/09/11/humans.rdfs-instances#John			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Sophie
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Mark			http://www.inria.fr/2007/09/11/humans.rdfs-instances#John
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Jack			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Gaston
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Pierre			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Flora
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Lucas			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Karl
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Catherine			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Laura
http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher			http://www.inria.fr/2007/09/11/humans.rdfs-instances#Maher

4. Declare the disjunction 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:

<ANSWER HERE/>

```

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

```

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

Code added to the schema:

<ANSWER HERE/>

```

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

```

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

Code added to the schema:

<ANSWER HERE/>

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

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):

<ANSWER HERE/>

```
<Class rdf:ID="Organization">
</Class>

<Class rdf:ID="University">
    <subClassOf rdf:resource="#Organization"/>
</Class>

<rdf:Property rdf:ID="mainEmployer">
    <domain rdf:resource="#Person"/>
    <range rdf:resource="#Organization"/>
</rdf:Property>

<owl:Class rdf:ID="Professor">
    <owl:intersectionOf rdf:parseType="Collection">
        <owl:Class rdf:about="#Researcher"/>
        <owl:Class rdf:about="#Lecturer"/>
    </owl:intersectionOf>
    <subClassOf>
        <owl:Restriction>
            <owl:onProperty rdf:resource="#mainEmployer" />
            <owl:allValuesFrom rdf:resource="#Organization" />
        </owl:Restriction>
    </subClassOf>
</owl:Class>
```

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

<ANSWER HERE/>

```
<Professor rdf:ID="Laura">  
    <mainEmployer rdf:resource="#Organization"/>  
</Professor>
```

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:

<ANSWER HERE/>

```
<owl:Class rdf:id="Person">

    <subClassOf>

        <owl:Restriction>

            <owl:onProperty rdf:resource="#hasParent" />

            <owl:someValuesFrom rdf:resource="#Woman" />

        </owl:Restriction>

    </subClassOf>

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

<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>

<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>

<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<YOUR EXAMPLE HERE/>
<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 .
```

<ANSWER HERE/>

wrong because there is only one possible preferred label per language

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

<ANSWER HERE/>

1. <http://www.w3.org/2004/02/skos/core#>
2. Definition: Section 10. Mapping Properties
3. Super-properties: skos:mappingRelation skos:narrower
4. Inverse of: 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?

<ANSWER HERE/>

1. <rdfs:subClassOf rdf:resource="http://purl.org/dc/terms/MediaType"/>
2. @prefix cc: <http://creativecommons.org/ns#> .
3. @prefix dc: <http://purl.org/dc/elements/1.1/> .
4. @prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
5. @prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .

```

6. @prefix xml: <http://www.w3.org/XML/1998/namespace> .
7. @prefix xsd: <http://www.w3.org/2001/XMLSchema#> .
8.
9. <https://www.amazon.com/Vinci-Code-Robert-Langdon/dp/0307474275> cc:permits
   cc:DerivativeWorks,
10.    cc:Distribution,
11.    cc:Reproduction ;
12.    cc:requires cc:Attribution,
13.    cc:Notice,
14.    cc:ShareAlike ;
15.    dc:creator <http://ns.inria.fr/Dan-Brown#me> ;
16.    dc:date "2014-12-01" ;
17.    dc:format "text/html" ;
18.    dc:language "en" ;
19.    dc:publisher <http://inria.fr> ;
20.    dc:subject "Cryptography, Novel, Religion" ;
21.    dc:title "Da Vinci Code" .

```

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?

<ANSWER HERE/>

3/ Types are: `owl#ObjectProperty, owl#InverseFunctionalProperty`

4/ `rdf:resource="http://xmlns.com/foaf/0.1/page"/>`

5/ domain: `Agent` ---- range: `Document`

```

<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//"/>

```

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/>

<ANSWER HERE/>

```
@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 rel: <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 "SEBAI" ;  
    foaf:givenname "Maher" ;  
    foaf:homepage <http://www.example.org/msebai> ;  
    foaf:knows [ a foaf:Person ;  
        foaf:mbox <mailto:hanene@yahoo.com> ;  
        foaf:name "Hanene" ] ;  
    foaf:mbox <mailto:magnus.zell@plutocow.com> ;  
    foaf:name "Maher SEBAI" ;  
    foaf:nick "WhosThatBud" ;  
    foaf:title "Mr" ;  
    foaf:workplaceHomepage <www.dsti.edu.institute> ;  
    rel:friendOf <http://www.unice.fr/data#Mickael>;  
    rel:spouseOf <http://www.unice.fr/data#Hanene>;  
    rel:hasMet <http://www.unice.fr/data#Slash>;  
    rel:enemyOf <http://www.unice.fr/data#NoOne>;  
    rel:siblingOf <http://www.unice.fr/data#Issam>;
```

Q6.6 What does this mean?

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

<ANSWER HERE/>

BioRDF2DBLP is linking BioRDF and DBLP and list triples in the two dataset using property exactMath

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.

<ANSWER HERE/>

2/ This query asks for all the sparql endpoint

3/ SELECT ?ds ?p ?val WHERE { ?ds rdfs:label "DBpedia-fr". ?ds ?p ?val }

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" .
```

<ANSWER HERE/>

Plot used stats1998 and generated Bar-chart. Stats1998 are distributed in csv format

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 .
```

<ANSWER HERE/>

RDF annotation saying that the dump is a transformation of sql result of employee and used foaf to represent the employees

Q6.10 practice:

1. Connect to the LOV directory: <https://lov.linkeddata.es/>
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)

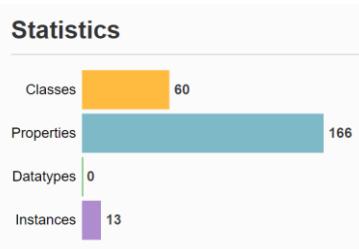
<ANSWER HERE/>

3/ the music ontology is used <http://purl.org/ontology/mo/>

4/Revision : Revision: 2.1.5

5/ yes other ontologies playlist ontology, theater ontology, open vocal ontology

6/



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

<ANSWER HERE/>

2 triples are constructed: first triple as an example

1/ resource: "https://www.ietf.org/rfc/rfc{2616}.txt";

Predicate : dc :title

Value : Hypertext Transfer Protocol -- HTTP/1.1

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>
```

<ANSWER HERE/>

Cathy foaf:type Person

Cathy foaf:name Catherine faron

Cathy foaf:mabox faron@i3s.unice

Cathy foaf:knows <Fabien gandon>

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>

<ANSWER HERE/>

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

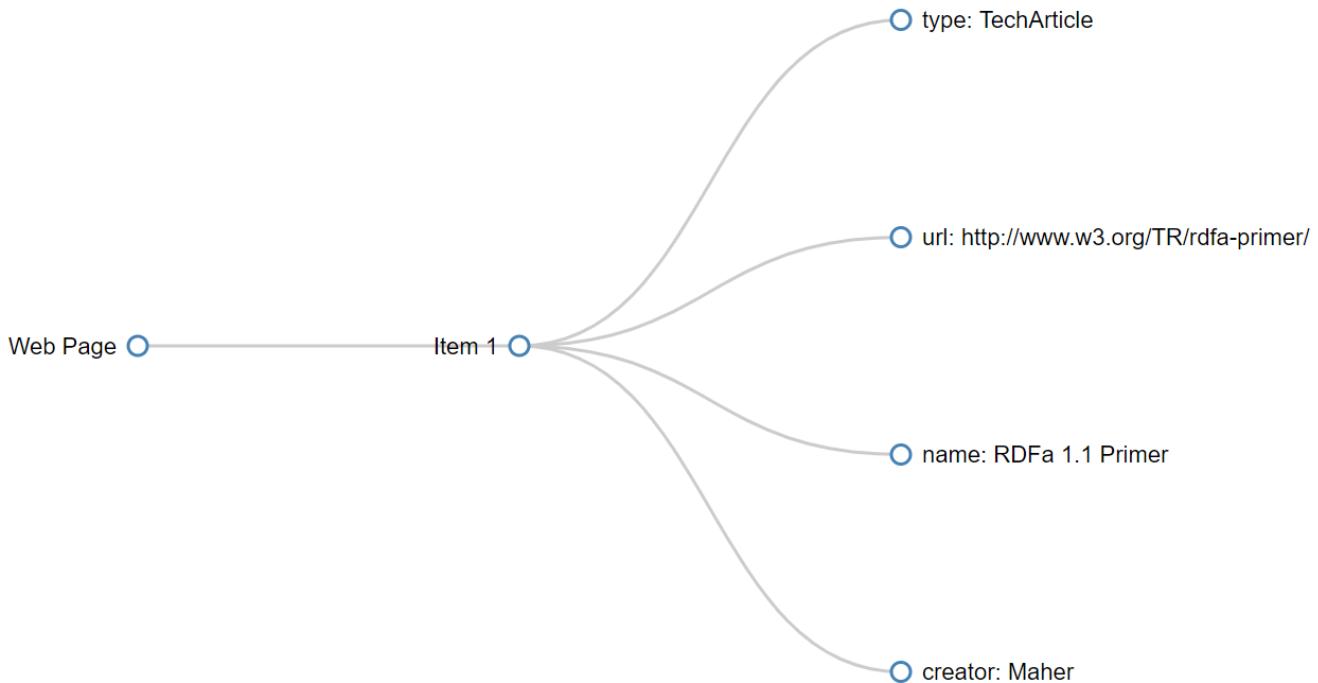
Second one is schema rdf in both html and rdf format

Q7.4 Use the online tool to play with RDFa adding for instance a “creator” property

<https://rdfa.info/play/>

<ANSWER HERE/>

```
<span property="creator">Maher</span></a>.
```



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/>

<ANSWER HERE/>

```
@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 xlink: <http://www.w3.org/1999/xlink> .
@prefix xml: <http://www.w3.org/XML/1998/namespace> .
@prefix xsd: <http://www.w3.org/2001/XMLSchema#> .

<https://www.imdb.com/title/tt1375666/> og:description "Directed by Christopher Nolan. With Leonardo DiCaprio, Joseph Gordon-Levitt, Ellen Page, Ken Watanabe. A thief who steals corporate secrets through the use of dream-sharing technology is given the inverse task of planting an idea into the mind of a CEO." ;
    og:image "https://m.media-
amazon.com/images/M/MV5BMjAxMzY3NjcxNF5BMl5BanBnXkFtZTcwNTI5OTM0Mw@@._V1_UY1200_CR90,0,630,1200_
AL_.jpg" ;
    og:site_name "IMDb" ;
    og:title "Inception (2010)" ;
    og:type "video.movie" ;
    og:url "http://www.imdb.com/title/tt1375666/" ;
    ns1:fbmlapp_id "115109575169727" .

```

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/>

<ANSWER HERE/>

```
{
  "@context": {
    "admin": "http://webns.net/mvcb/",
    "foaf": "http://xmlns.com/foaf/0.1/",
    "rdf": "http://www.w3.org/1999/02/22-rdf-syntax-ns#",
    "rdfs": "http://www.w3.org/2000/01/rdf-schema#",
    "rel": "http://purl.org/vocab/relationship/",
```

```
"xsd": "http://www.w3.org/2001/XMLSchema#"

},
"@graph": [
{
"@id": "_:ub66bL18C16",
"@type": "foaf:Person",
"foaf:mbox": {
"@id": "mailto:hanene@yah.oo"
},
"foaf:name": "Hanene"
},
{
"@id": "file:///base/data/home/apps/s%7Erdf-translator/2.408516547054015808/",
"@type": "foaf:PersonalProfileDocument",
"admin:errorReportsTo": {
"@id": "mailto:leigh@ldodds.com"
},
"admin:generatorAgent": {
"@id": "http://www.ldodds.com/foaf/foaf-a-matic"
},
"foaf:maker": {
"@id": "file:///base/data/home/apps/s%7Erdf-translator/2.408516547054015808/#me"
},
"foaf:primaryTopic": {
"@id": "file:///base/data/home/apps/s%7Erdf-translator/2.408516547054015808/#me"
}
},
{
"@id": "file:///base/data/home/apps/s%7Erdf-translator/2.408516547054015808/#me",
"@type": "foaf:Person",
"foaf:family_name": "SEBAI",
"foaf:givenname": "Maher",
"foaf:homepage": {
"@id": "http://www.example.org/msebai"
},
"foaf:knows": {
```

```
"@id": "_:ub66bL18C16"
},
"foaf:mbox": {
    "@id": "mailto:magnus.zell@plutocow.com"
},
"foaf:name": "Maher SEBAI",
"foaf:nick": "WhosThatBud",
"foaf:title": "Mr",
"foaf:workplaceHomepage": {
    "@id": "file:///base/data/home/apps/s%7Erdf-
translator/2.408516547054015808/www.dsti.edu.institute"
},
"rel:enemyOf": {
    "@id": "http://www.unice.fr/data#NoOne"
},
"rel:friendOf": {
    "@id": "http://www.unice.fr/data#Mickael"
},
"rel:hasMet": {
    "@id": "http://www.unice.fr/data#Slash"
},
"rel:siblingOf": {
    "@id": "http://www.unice.fr/data#Issam"
},
"rel:spouseOf": {
    "@id": "http://www.unice.fr/data#Hanene"
}
}
]
}
```

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.

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