RDFS practical session

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

Family.rdfs

- Write the the RDF schema that you used in the description of Margot in a RDF/XML (or in turtle and then translate it) and save the RDF/XML in a file called “Family.rdfs”.
- Of course, this assumes that the URIs for the classes and properties declared/used must match in both files. You mays have to update the files Margot.rdf and Margot.ttl to use your ontology.
- Check that your RDF schema and RDF files are valid using the W3C's RDF validation service.
- Launch the standalone interface of Corese and load your files Family.rdfs and Margot.rdf
- The interface contains a default SPARQL query:
  `Select? x? t where {? x RDF: type? t}`
  Launch the query and examine the results.
- Modify your ontology to declare the class of Humans as a super class of Man and Woman (don’t change the data), reload the schemas and data and search for the humans to see the results
- Modify your ontology to declare the property of familyLink as a super property of hasChild and hasSpouse (don’t change the data), reload the schemas and data and search for the family links to see the results.
- Modify your ontology to declare the property a 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.

About the human.rdfs schema

1. Download the RDF schema available at this address and save it as “human.rdfs”: http://wimmics.inria.fr/doc/tutorial/human_2013.rdfs
2. What is the namespace associated with this ontology? How was it associated?
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?
5. What classes can have the age property?
6. Look at the beginning of the file and draw the subgraph of the hierarchy containing class Animal, Man and Woman.

Query the schema

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.
2. Write a query to find all the links subClassOf in the ontology.
3. Write a query to find the definitions and translations of "shoe size".
4. Write a query to find the synonyms of the word 'person' in French. What are the answers?
5. Write a query to find the different meaning of the terms "size" and "man" (disambiguation). What are the answers?
6. What are properties that use the class Person in their signatures?
7. Rebuild the hierarchy of Classes (CONSTRUCT) considering only the classes in the humans.rdfs schema
8. To the previous CONSTRUCT add the signatures of the relations.

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

Question 2

1. Write a query to find <u>Males</u> and their wives. How many answers do you get? Explain this result.
2. In the data declare that Lucas has to father Karl. Reset Corese, reload the ontology and the data, and then rerun the query to find <u>Males</u> and their wives. Explain the new result.

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.
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.

Question 4

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

Question 5

1. Write a query to find the family cores (couples and their children) using a SELECT
2. Modify it to display the result with a CONSTRUCT query

Question 6

1. Declare the olderThan relationship in the schema to indicate between two people which is eldest and construct the arcs between peoples with a SPARQL query
2. Find a query that generates only the minimum number of link without redundancy with olderThan transitivity.

Question 7

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

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.