# RDF exercises

## Exercise 1

Take the following natural language sentences:

- "TELECOM SudParis is located at 9 rue Charles Fourier 91000, Evry, France"
- "The web site http://www.w3.org/ hosts the homepage for the W3C Semantic Web activity"

(a) Write down the sentences in RDF triple notation, using at least 3 triples for each sentence

(b) Write down the sentences in RDF triple notation with at least one blank node, using at least 3 triples for each sentence

URI : URN or URL or URC

| | | |
|---|---|---|
| urn:TELECOMSudParis | Located_at | urn:locationOfTelecomSudParis |
| urn:locationOfTelecomSudParis | hasNumber | 9 |
| urn:locationOfTelecomSudParis | HasStreet | Charles Fourier |
| urn:locationOfTelecomSudParis | HasZipCode | 91000 |
| urn:locationOfTelecomSudParis | hasCity | Evry |
| urn:locationOfTelecomSudParis | HasCountry | France |
| http://www.w3.org/ | Is-a | Web site |
| urn: W3CSemanticWeb | hasActivity | urn:ActivityOfW3CSemanticWeb |
| http://www.w3.org/ | Hosts_website_of | urn:ActivityOfW3CSemanticWeb |
| | | |
| | | |
| | | |

## Exercise 2

Take your solution of exercise 1b and write them down as a directed labelled graph where resources are ellipses, literals are boxes and properties are arrows.

## Exercise 3

Write the following sentences in RDF triple notation, directed label graph model, and RDF/XML based syntax.

- John is the husband of Mary.
- John was born the 01st of January 1975.
- Mary is claiming that John was born at Galway.
- John has as children Brian, Sinead, and David.
- John got as children Brian, Sinead, and then David (in that order).
- Galway can be reached either by train, by bus, or by plane.
- Galway can be reached only by train, by bus or by plane.

# RDFS exercises

## Exercise 1

Consider the following RDFS ontology:

prefix ex:, namespace URI: http://www.exemple.org
prefix rdf:, namespace URI: http://www.w3.org/1999/02/22-rdf-syntax-ns#
prefix rdfs:, namespace URI: http://www.w3.org/2000/01/rdf-schema#
prefix xsd:, namespace URI: http://www.w3.org/2001/XMLSchema#

```
ex:A rdf:type rdfs:Class .
ex:B rdf:type rdfs:Class.
ex:C rdf:type rdfs:Class.
ex:B rdfs:subClassOf ex:A .
ex:C rdfs:subClassOf ex:B .

ex:P1 rdf:type rdf:Property .
ex:P1 rdfs:domain ex:A .
ex:P1 rdfs:range ex:B .
ex:P2 rdf:type rdf:Property.
ex:P2 rdfs:domain ex:B .
ex:P2 rdfs:range ex:C .
ex:P2 rdfs:subPropertyOf ex:P1 .
ex:P3 rdf:type rdf:Property .
ex:P3 rdfs:domain ex:C .
ex:P3 rdf:range xsd:integer .
ex:P3 rdfs:subPropertyOf ex:P2 .

ex:a rdf:type ex:A .
ex:b rdf:type ex:B .
ex:c1 rdf:type ex:C .
ex:c2 rdf:type ex:C .
ex:a ex:P1 ex:b .
ex:b ex:P2 ex:c1 .
ex:c1 ex:P3 3^^xsd:integer.
```

Draw the corresponding RDFS and RDF graphs.

## Exercise 2

Consider the following RDFS ontology:

prefix pe:, namespace URI: http://www.personne.org
prefix rdf:, namespace URI: http://www.w3.org/1999/02/22-rdf-syntax-ns#
prefix rdfs:, namespace URI: http://www.w3.org/2000/01/rdf-schema#
prefix xsd:, namespace URI: http://www.w3.org/2001/XMLSchema#

```
pe:Animal rdf:type rdfs:Class .
pe:Personne rdf:type rdfs:Class.
pe:Femme rdf:type rdfs:Class.
pe:Homme rdf:type rdfs:Class.
pe:Ville rdf:type rdfs:Class.
pe:Femme  rdfs:subClassOf   ex:Personne .
pe:Homme  rdfs:subClassOf   ex:Personne .

pe:Habite rdf:type   rdf:Property .
pe:Habite   rdfs:domain   pe:Personne .
pe:Habite   rdfs:range   pe:Ville .
pe:APourNom   rdf:type   rdf:Property .
pe:APourNom   rdfs:domain  pe:Personne  .
pe:APourNom   rdfs:range   xsd:string  .
pe:Possède   rdf:type   rdf:Property.
pe:Possède  rdfs:domain   pe:Personne  .
pe:Possède   rdfs:range  pe:Animal  .

pe:Snoopy rdf:type ex:Animal .
pe:Anne rdf:type ex:Femme .
pe:Paris   rdf:type  ex:Ville  .
pe:Anne   ex:Habite  ex:Paris .
pe:Anne   ex:APourNom  Dutra^^xsd:string  .
pe:Anne  ex:Possède  ex:Snoopy  .
```

Draw the corresponding RDFS and RDF graphs.

## Exercise 3

Create an RDFS ontology (in triple or graph notation) about Persons and Universities. Create at least 3 classes and at least 2 properties; use at least 2 *subClassOf* statements.

## Exercise 4

Extend your ontology of exercise 3 to include a set of instances, i.e., a number of persons and a number of universities. You can use either triple, graph or RDF/XML notation.
Make sure you include at least one container or collection and include at least one literal.

## Exercise 5

Model the very simple RDFS ontology described below:

- A pizza has a base and a topping.
- We distinguish two kinds of pizza base: Deep ban base, and Thin and Crispy base.
- We distinguish four kinds of pizza topping: Cheese, Meat, Seafood and Vegetable toppings.
- We consider three kinds of pizza: Cheesy, Named and Vegetarian.
- We distinguish four kinds of named pizza: American, AmericanHot, Margherita, and Soho.