IN3060/IN4060 – MANDATORY EXERCISE 7

Published date: 25.04.2024

Due date: 09.05.2024 23:59

Delivery files: 1: library.stottr

Delivery attempts: 1.

Read through the whole exercise text first.

These are the prefixes used in this document:

```
@prefix dc:   <http://purl.org/dc/elements/1.1/> .
@prefix dbpo: <http://dbpedia.org/ontology/> .
@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 geo:  <http://www.w3.org/2003/01/geo/wgs84_pos#> .
@prefix xsd:  <http://www.w3.org/2001/XMLSchema#> .
@prefix ottr: <http://ns.ottr.xyz/0.4/> .
@prefix o-rdf: <http://tpl.ottr.xyz/rdf/0.1/> .
@prefix ex: <http://example.com/ns#> .
```

The goal of the exercise is to convert data about public sculptures in Stavanger into RDF data using OTTR templates.

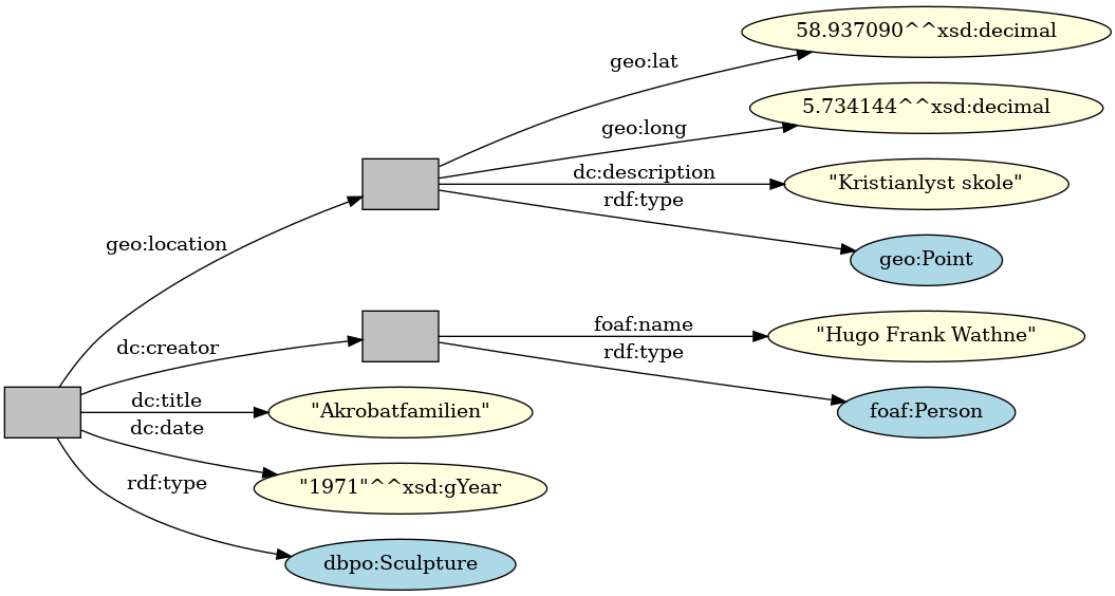
The dataset[1] is in the form of a small Excel file with one table. A sample, including all columns but only some rows, is shown below. **Note that the column "Year" is missing a value for some the rows.**

Latitude	Longitude	Location	Sculpture name	Artist	Year
58.937090	5.734144	Kristianlyst skole	Akrobatfamilien	Hugo Frank Wathne	1971
58.969712	5.731962	Torget	Alexander L. Kjelland	Magnus Vigrestad	1928
58.941782	5.669231	Ved Møllebukta	Fiskerens kone	Reier E. Eide	1994
58.959815	5.737274	Roaldsøy skole.	Fole	Svein Magnus Håvardstein	
58.969151	5.748208	Johannesparken	Folungene	Hugo Frank Wathne	1960

The RDF data you will produce must have the following form, here exemplified using the first row of the above sample. The RDF graph of the example is also depicted below.

```
[ a             dbpo:Sculpture ;
  dc:creator    [ a          foaf:Person ;
                  foaf:name  "Hugo Frank Wathne"
                ] ;
  dc:date       "1971"^^xsd:gYear ;
  dc:title      "Akrobatfamilien" ;
  geo:location  [ a             geo:Point ;
                  geo:lat       58.937090 ;
                  dc:description  "Kristianlyst skole" ;
                  geo:long      5.734144
                ]
] .
```

[1] The dataset is taken from a real dataset published on Stavanger's website for open data https://open.stavanger.kommune.no/dataset/skulpturer-i-stavanger and which is available via Norway's open data portal data.norge.no.)



To produce the specified RDF data you must define three templates: `ex:Sculpture`, `ex:Point` and `ex:Person`. The complete signatures of `ex:Point` and `ex:Person`, and an outline of the signature of `ex:Sculpture`, listing just the parameter variables, are given below.

```
ex:Person[
  ottr:IRI ?personIRI,
  ! xsd:string ?name   # non-blank
]  .

ex:Point[
  ottr:IRI ?locationIRI,
  ! xsd:decimal ?lat,       # non-blank
  ! xsd:decimal ?long,      # non-blank
  ?! xsd:string ?location   # optional, non-blank
]  .

ex:Sculpture [ ?lat, ?long, ?location, ?title, ?artistName, ?year ]  .
```

Exercises

Your task is to extend the signatures given above to complete templates so that they can be used to expand the dataset to produce the specified RDF format.

When defining these templates, the only other template you will need to use are the `ottr:Triple` and `http://tpl.ottr.xyz/rdf/0.1/Type` templates.

`ex:Person`

The template `ex:Person` should be defined so that the template instance

```
ex:Person([ ], "Hugo Frank Wathne" ) .
```

expands to

```
[ ]  a  foaf:Person ;
        foaf:name  "Hugo Frank Wathne" .
```

The template should make use of the `http://tpl.ottr.xyz/rdf/0.1/Type` template.

`ex:Point`

The template `ex:Point` should be defined so that the template instance

```
ex:Point ([ ], 58.937090, 5.734144, "Kristianlyst skole" ) .
```

expands to

```
[ ]  a  geo:Point ;
        geo:lat       58.937090 ;
        dc:description  "Kristianlyst skole" ;
        geo:long      5.734144 .
```

The template should make use of the `http://tpl.ottr.xyz/rdf/0.1/Type` template.

`ex:Sculpture`

The template `ex:Sculpture` must be defined so that each row in the dataset is translated to the RDF shape specified in the introduction.

The template must depend on the `ex:Person` and `ex:Point` templates. To correctly define the `ex:Sculpture` template you will also need to adapt the signature outline given above to the requirements set by the signatures on which the `ex:Sculpture` depends, i.e., make sure that parameter types, non-blank and optional flags are correctly set.

Expanding

The input Excel spreadsheet is found alongside this exercise set, at https://www.uio.no/studier/emner/matnat/ifi/IN3060/v23/obliger/skulpturer-i-stavanger.xls. It is annotated with tabOTTR instructions which allows Lutra or WebLutra to process the dataset as template instances. To expand the spreadsheet run the following command:

```
java -jar lutra.jar \
 --mode expand \
 -L stottr \
 -l lib/library.stottr \
 -f \
 -o output.ttl \
 -I tabottr \
 data/skulpturer-i-stavanger.xls
```

Run

```
java -jar lutra.jar --help
```

to get an explanation of what the different options mean.

Download the latest Lutra from https://ottr.xyz/downloads/lutra/.

You can also use WebLutra: https://weblutra.ottr.xyz. WebLutra runs a slighly old version of Lutra, so you may experience slightly different behaviour between the two.

You should use Mr. Oblig to verify that your template library is correct by checking that the expanded RDF graph is correct.

Information about OTTR

The website https://ottr.xyz contains all information about OTTR. Read the page at `ottr.xyz` to get an overview of OTTR. Watch the screencasts which can be found on the same page. Read and understand the primer about the fundamentals about OTTR found at https://primer.ottr.xyz.

The project is an ongoing research and innovation project at IfI. If you encounter bugs or irregularities, then please let us know by sending a good bug report[2] to https://gitlab.com/ottr/lutra/lutra/-/issues.

[2] See, e.g., https://testfort.com/blog/8-tips-for-writing-a-good-bug-report for some tips.