Assignment 7
Linked Data and Ontologies

Applied Data Science

Instructions

This assignment requires you to answer questions about linked data and create an ontology. Follow the instructions given in the questions for submission. Submit your assignment as a PDF file with your name and email in the header, single-spaced and font no larger than 12pt. Assignment will be graded based on the understanding of the underlying concepts.

Part 1 : Linked Open Data

Please answer the following questions according to what you have learnt from the 5 stars open data tutorial and your own knowledge.

1. What is the difference in the example given of the 1st and 2nd star?
2. To publish at the 5 star criteria level requires effort. Please describe the effort (cost) to go from 4 star to 5 star. i.e. use URIs to denote things, so that people can point at your stuff.
3. According to that site, what is the cost to a consumer that is associated with publishing at the level of 5 stars?
4. What changes in the RDF Metadata for Galway from level 4-star to 5-star?

Part 2: SPARQL query

For this part, you will go to the DBPedia SPARQL endpoint (https://dbpedia.org/sparql) and use SPARQL queries to answer the following questions. Take screenshots of the query results and attach them together with your SPARQL queries in your assignment.

1. Use the following SPARQL query as your starting point:

```
SELECT DISTINCT ?Concept where {[] a ?Concept} LIMIT 100
```

Run the query and you will see the first 100 concepts. Each concept is a link to somewhere else. Click the links and browse, get a sense of what these concepts are.
Then change the query so only first 10 concepts are returned. Take a screenshot of your result.

2. Now let's do something more interesting. The following query can show us what we can do with John Lennon:

```
SELECT *
WHERE { <http://dbpedia.org/resource/John_Lennon> ?p ?o}
```

We can use prefix to avoid typing the URI every time. The query above is equal to:

```
PREFIX db: <http://dbpedia.org/resource/>
SELECT *
WHERE { db:John_Lennon ?p ?o}
```

Now modify the following query to find Lennon's birth date and take a screenshot of it.

```
PREFIX db: <http://dbpedia.org/resource/>
PREFIX onto: <http://dbpedia.org/ontology/>

SELECT *
WHERE { db:John_Lennon onto:<Your Input> <Your Input> }
```

3. Say now we want to know the people who were born in Brooklyn, then the query below is what we need:

```
PREFIX db: <http://dbpedia.org/resource/>
PREFIX onto: <http://dbpedia.org/ontology/>

SELECT *
WHERE { ?s onto:birthPlace db:Brooklyn }
Limit 10
```

Now, modify the following SPARQL query and find out the people born in Brooklyn and died in Seattle and take a screenshot:

```
PREFIX db: <http://dbpedia.org/resource/>
PREFIX onto: <http://dbpedia.org/ontology/>

SELECT *
WHERE { ?s <Your input> <Your input> .
        ?s <Your input> <Your input> }
```

4. Now we want a list of people born after 1999. We need to apply a filter to the query result to achieve this.

```
PREFIX db: <http://dbpedia.org/resource/>
PREFIX onto: <http://dbpedia.org/ontology/>

SELECT *
WHERE {
?p onto:birthYear ?year .
FILTER (?year > "1999"^^<http://www.w3.org/2001/XMLSchema#gYear>)
}
```

Notice that

```
"1999"^^<http://www.w3.org/2001/XMLSchema#gYear>
```

is the representation of year 1999.
What you need to do is to find the people born in Brooklyn and born between the year 1985 and 1990 (not include 1985 and 1990). Take a screenshot of your result. Hint: && is used to represent the logic and in the filter.
Use the w3c specification (https://www.w3.org/TR/rdf-sparql-query/) as reference if you get stuck.

Part 3: Ontology

For this part you will create an ontology for pizzas.
To do this, you will use a tool called an ontology editor. The one you will use is called Protege, it is freely available from Stanford University. You can download and install Protege (https://protege.stanford.edu/) on your computer.
Go through the first 66 pages of the Protege OWL Pizza Tutorial up to exercise 38. Submit your pizza ontology in .owl(in RDF/XML format) together with the PDF file.

Notes: The Pizza Tutorial is based on Protege 4. The default version offered at their web site is Protege 5. The user interface of Protege 5 is slightly different from Protege 4. If you want the newest Protege, just download directly. As a result, you may need some deduction so you can follow the tutorial. If you want to avoid the extra trouble, go to https://protegewiki.stanford.edu/wiki/Protege_Desktop_Old_Versions for an older version of Protege.