SPARQL Tutorial

Triplestores store data in graph format. There are other kinds of graph-based databases, but RDF is the preferred format for linked data triplestores. A graph, in RDF terms, is simply a list of triples. The benefit of this is that two lists of triples can be combined to make a larger list of triples.

The other benefit of such a sparse data structure is that it can evolve as more data is added. With a relational database, the table structure must be defined before any data can be added. This is not the case with triples. Instead, the structure of a query response is determined at query time.

In order to query a triplestore, the language known as SPARQL is used. SPARQL is to a triplestore as SQL is to a relational database. It’s similar enough to look familiar to SQL users, but different enough to confuse you if you think they are alike!

SPARQL Basics

SELECT

Gets the Label (name) of all buildings
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT ?uri ?label WHERE {
   ?uri a <http://vocab.deri.ie/rooms#Building> .
}
ORDER BY ?label

Exercise

Our dataset contains information about points of service (eg places that sell things) on campus. The URI for a point of service is http://purl.org/goodrelations/v1#LocationOfSalesOrServiceProvisioning

Modify the above query to return a list of the names of all the points of service, rather than all the buildings.
OPTIONAL

One of the benefits of RDF over traditional databases is that all properties are optional. A building may contain triples describing its geographical location, but it would make no sense to assign these properties to a person, or a cup of coffee. Often we need to write a query that structures some information, but do not wish to ignore entities that lack those properties. For this we use the OPTIONAL keyword.

Gets the **Label (name)**, **Latitude** and **Longitude** of all buildings on Highfield Campus, or just the name if the lat/lon do not exist in the data.

```
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX spacerel: <http://data.ordnancesurvey.co.uk/ontology/spatialrelations/>

SELECT DISTINCT ?label ?lat ?long WHERE {
  ?building spacerel:within <http://id.southampton.ac.uk/site/1> .
  OPTIONAL {
  }
}
ORDER BY ?lat
```

**Exercise**

Going back to the query for points of service, all places that serve food to the public must have a food hygiene certificate. We include this information in our data using the predicate [uri] <http://purl.org/openorg/ukfhrsRatingValue> [value]. Modify your previous query so that, as well as returning the name of all points of service, it also returns the food hygiene certificate rating if it exists.
**FILTER**

So far we have effectively just been doing pattern matching, which is fine if we know exactly what we are looking for. However, there are times we need to add conditions to our queries, and this is where the FILTER keyword comes in.

Gets information about all buildings built after 1995.

```sparql
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX spacerel: <http://data.ordnancesurvey.co.uk/ontology/spatialrelations/>

SELECT DISTINCT ?label ?lat ?long ?build_date WHERE {
    ?building spacerel:within <http://id.southampton.ac.uk/site/1> .
    ?building <http://id.southampton.ac.uk/ns/buildingDate> ?build_date .
    FILTER ( ?build_date > "1995" )
    OPTIONAL {
    }
}
ORDER BY ?lat
```

**Exercise**

Modify your previous query to find points of service with a food hygiene rating of less than the maximum 5.
Other Examples

Finds events which are happening in a room which has an induction loop.

PREFIX event: <http://purl.org/NET/c4dm/event.owl#>
PREFIX tl: <http://purl.org/NET/c4dm/timeline.owl#>
PREFIX org: <http://www.w3.org/ns/org#>
PREFIX rooms: <http://vocab.deri.ie/rooms#>
PREFIX oo: <http://purl.org/openorg/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

SELECT DISTINCT ?start ?room_name ?event_name ?au_name WHERE {
  ?room oo:hasFeature ?feature .
  ?room a rooms:Room .
  OPTIONAL {
    ?au a <http://www.w3.org/ns/org#OrganizationalUnit> .
  }
  OPTIONAL { ?ev rdfs:label ?event_name . }

  OPTIONAL {
    GRAPH <http://id.southampton.ac.uk/dataset/room-features/latest> {
    }
  }

  FILTER ( ?start > "2013-02-19"^^xsd:dateTime )
}
ORDER BY ?start
GETS INFORMATION ABOUT ALL THE POINTS OF SERVICES AND SITES OR BUILDINGS THEY ARE ON, NEAR OR WITHIN. IT RETRIEVES ITS LABEL, PHONE NUMBER, CONTACT EMAIL, LATITUDE, LONGITUDE, ICON TO SHOW ON A MAP AND INFORMATION ABOUT DISABILITY ACCESS.

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX sr: <http://data.ordnancesurvey.co.uk/ontology/spatialrelations/>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dcterms: <http://purl.org/dc/terms/>
PREFIX geo: <http://www.w3.org/2003/01/geo/wgs84_pos#>
PREFIX oo: <http://purl.org/openorg/>

SELECT DISTINCT *
{
    ?pos a <http://purl.org/goodrelations/v1#LocationOfSalesOrServiceProvisioning> ;
    rdfs:label ?label .
    OPTIONAL {
        ?site a <http://www.w3.org/ns/org#Site> ; rdfs:label ?site_label .
        OPTIONAL { ?site skos:notation ?site_id }
    }
    OPTIONAL {
        ?near_site a <http://www.w3.org/ns/org#Site> ; rdfs:label ?near_site_label .
        OPTIONAL { ?near_site skos:notation ?near_site_id }
    }
    OPTIONAL {
        OPTIONAL { ?building sr:within ?site . ?site a <http://www.w3.org/ns/org#Site> ; rdfs:label ?site_label }
        OPTIONAL { ?building skos:notation ?building_id }
        OPTIONAL { ?site skos:notation ?site_id }
        OPTIONAL { ?building geo:lat ?building_lat ; geo:long ?building_long }
    }
    OPTIONAL { ?pos foaf:homepage ?homepage }
    OPTIONAL { ?pos foaf:phone ?phone }
    OPTIONAL { ?pos foaf:mbox ?email }
    OPTIONAL { ?pos dcterms:description ?description }
    OPTIONAL { ?pos geo:lat ?lat ; geo:long ?long }
    OPTIONAL { ?pos oo:mapIcon ?icon }
    OPTIONAL { ?pos <http://id.southampton.ac.uk/ns/disabledGoPage> ?disabledGoPage }
}
ORDER BY ?label