# Semantic Web Lab

In this tutorial we will use the [DBpedia SPARQL endpoint](http://dbpedia.org/sparql/) from [DBpedia.org](http://dbpedia.org), which as we discussed - is a freely available, community-backed, database filled with RDF data extracted from Wikipedia.

Type the [SPARQL](http://www.w3.org/TR/2013/REC-sparql11-query-20130321/#WritingSimpleQueries) Queries directly into the [DBpedia's SPARQL UI](http://dbpedia.org/sparql/) for testing.

**Grading**:

* + Enter your queries into the document in the areas indicated
  + Save the output in RDF/XML format – using our Class\_C*onvention\_*Lab10Qn.rdf where n is the question number
  + Q 1-5 : 5 points per query – no query, no points - no output, no points.
  + Q 6: 25 points – Create your own rdf [foaf](http://xmlns.com/foaf/spec/) file – include it in your zip file as
    - *your\_initials*foaf.rdf
  + Work on these individually – you may want to review some [SPARQL documentation](http://www.w3.org/TR/2013/REC-sparql11-query-20130321/#WritingSimpleQueries) here or here: <http://jena.apache.org/tutorials/sparql.html>
  + Don’t forget your readme files
  + Zip everything up as per convention

1. The following query will retrieve a list of all Books <<http://dbpedia.org/ontology/Book>> in DBpedia

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX ontology: <http://dbpedia.org/ontology/>

select distinct ?bookUri

where { ?bookUri rdf:type ontology:Book . }

Modify this query to retrieve a list of all movies (<http://dbpedia.org/ontology/Film>).

Place your Query here:

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX ontology: <http://dbpedia.org/ontology/>

select distinct ?filmUri

where { ?filmUri rdf:type ontology:Film . }

1. The following Query retrieves all the URIs that identify cities that are of type “Cities in Texas”

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
SELECT \* WHERE {  
    ?city rdf:type <http://dbpedia.org/class/yago/CitiesInTexas>  
}

Write a query to retrieve the areaTotal ([dbpedia-owl:areaTotal](http://dbpedia.org/ontology/areaTotal)) of cities of type “Cities in Texas”

PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT \*

WHERE

{

?city rdf:type <http://dbpedia.org/class/yago/CitiesInTexas> .

?city dbpedia-owl:areaTotal ?areaTotal .

}

1. Write a Query to return the cities that are of type "Cities in Texas" with their areaTotal and postal code

PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT \*

WHERE

{

?city rdf:type <http://dbpedia.org/class/yago/CitiesInTexas> .

?city dbpedia-owl:areaTotal ?areaTotal .

?city dbpedia-owl:postalCode ?postalCode .

}

1. Write a Query which returns the cities that are of type "Cities in Texas" with their areaTotal and postal code. The results should be returned in the order of their areaTotal. Return atmost 20 results, starting with the 10th result.

PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT \*

WHERE

{

?city rdf:type <http://dbpedia.org/class/yago/CitiesInTexas> .

?city dbpedia-owl:areaTotal ?areaTotal .

?city dbpedia-owl:postalCode ?postalCode .

}

LIMIT 20

OFFSET 10

1. Write a Query which returns the cities that are of type "Cities in Texas" with their areaTotal and postal code. The results should be returned in the order of their areaTotal. Return only results with total area greater than 300000000 (3.0e+08).
2. Create your foaf file

PREFIX dbpedia-owl: <http://dbpedia.org/ontology/>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT DISTINCT \*

WHERE

{

?city rdf:type <http://dbpedia.org/class/yago/CitiesInTexas> .

?city dbpedia-owl:areaTotal ?areaTotal .

?city dbpedia-owl:postalCode ?postalCode .

FILTER(?areaTotal > 300000000)

}

ORDER BY asc(?areaTotal)