

Εθνικό Μετσόβιο Πολυτεχνείο

Σχολή Ηλεκτρολόγων Μηχανικών
και Μηχανικών Υπολογιστών

Συστήματα και Τεχνολογίες Γνώσης

8^ο Εξάμηνο - Ροή Α

Πρώτη Σειρά Ασκήσεων

Semantic Web - RDF - SPARQL

Δημήτρης Δήμος - 031 17 165
dimitris.dimos647@gmail.com



Αθήνα
Άνοιξη, 2021

Ερώτημα 1

1. Η Κρήτη είναι ένα νησί της Ελλάδας, το οποίο διαιρείται στις διοικητικές περιφέρειες Χανίων, Ρεθύμνης, Ηρακλείου και Λασιθίου.

Η γνώση (1) μοντελοποιείται σε RDF (χωρίς χρήση των δυνατοτήτων του RDF Schema) ως εξής:

RDF Model

ex:Crete	rdf:type	ex:Island .
ex:Crete	exp:belongsTo	ex:Greece .
ex:Crete	exp:isDividedInto	_:Districts .
_:Districts	rdf:type	rdf:Bag .
_:Districts	rdf:_1	ex:Chania .
_:Districts	rdf:_2	ex:Rethymno .
_:Districts	rdf:_3	ex:Hraklio .
_:Districts	rdf:_4	ex:Lasithi .

Τροποποιούμε κι εμπλουτίζουμε τη γνώση σύμφωνα με το RDF Schema:

Ορισμός Λεξιλογίου

ex:Country	rdf:type	rdfs:Class .
ex:Island	rdf:type	rdfs:Class .
ex:AdminDistrict	rdf:type	rdfs:Class .
exp:belongsTo	rdf:type	rdf:Property .
exp:belongsTo	rdfs:domain	ex:Island .
exp:belongsTo	rdfs:range	ex:Country .
exp:isDividedInto	rdf:type	rdf:Property .
exp:isDividedInto	rdfs:domain	ex:Country .
exp:isDividedInto	rdfs:range	rdf:Bag .

Δεδομένα

ex:Crete	rdf:type	ex:Island .			
ex:Crete	rdfs:label	"Κρήτη" .			
ex:Greece	rdf:type	ex:Country .	ex:Crete	exp:belongsTo	ex:Greece .
ex:Greece	rdfs:label	"Ελλάδα" .	ex:Crete	exp:isDividedInto	_:Districts .
ex:Chania	rdf:type	ex:AdminDistrict .	_:Districts	rdf:type	rdf:Bag .
ex:Chania	rdfs:label	"Χανιά" .	_:Districts	rdf:_1	ex:Chania .
ex:Rethymno	rdf:type	ex:AdminDistrict .	_:Districts	rdf:_2	ex:Rethymno .
ex:Rethymno	rdfs:label	"Ρέθυμνο" .	_:Districts	rdf:_3	ex:Hraklio .
ex:Hraklio	rdf:type	ex:AdminDistrict .	_:Districts	rdf:_4	ex:Lasithi .
ex:Hraklio	rdfs:label	"Ηράκλειο" .			
ex:Lasithi	rdf:type	ex:AdminDistrict .			
ex:Lasithi	rdfs:label	"Λασιθί" .			

2. Ένα συγκεκριμένο κείμενο που έχει έναν τίτλο και αποτελείται κατά σειρά από μια ενότητα για τον πρόλογο, μια ενότητα για το κυρίως κείμενο και μια ενότητα για τα συμπεράσματα. Θεωρήστε ότι για το συγκεκριμένο κείμενο σας δίνονται ο τίτλος του και το κειμενικό περιεχόμενο κάθε ενότητάς του.

Η γνώση (2) μοντελοποιείται σε RDF (χωρίς χρήση των δυνατοτήτων του RDF Schema) ως εξής:

RDF Model

1ος Τρόπος

ex:Keimeno	rdf:type	ex:Text .
ex:Keimeno	exp:hasTitle	"Text Title"@en .
ex:Keimeno	exp:consistsOf	_:Sections .
_:Sections	rdf:type	rdf:List .
_:Sections	rdf:first	ex:Prologue .
_:Sections	rdf:rest	_:c1 .
_:c1	rdf:first	ex:MainBody .
_:c1	rdf:rest	_:c2 .
_:c2	rdf:first	ex:Conclusions .
_:c2	rdf:rest	rdf:nil .
ex:Prologue	exp:hasContent	ex:PrologueContent .
ex:MainBody	exp:hasContent	ex:MainBodyContent .
ex:Conclusions	exp:hasContent	ex:ConclusionsContent .
ex:PrologueContent	rdf:type	rdf:langString .
ex:MainBodyContent	rdf:type	rdf:langString .
ex:ConclusionsContent	rdf:type	rdf:langString .

2ος Τρόπος

ex:Keimeno	rdf:type	ex:Text .
ex:Keimeno	exp:hasTitle	"Text Title"@en .
ex:Keimeno	exp:consistsOf	_:Sections .
_:Sections	exp:include	ex:Prologue .
_:Sections	exp:include	ex:MainBody .
_:Sections	exp:include	ex:Conclusions .
ex:Prologue	exp:followedBy	ex:MainBody .
ex:MainBody	exp:precededBy	ex:Prologue .
ex:MainBody	exp:followedBy	ex:Conclusions .
ex:Conclusions	exp:precededBy	ex:MainBody .
ex:Prologue	exp:hasContent	ex:PrologueContent .
ex:MainBody	exp:hasContent	ex:MainBodyContent .
ex:Conclusions	exp:hasContent	ex:ConclusionsContent .
ex:PrologueContent	rdf:type	rdf:langString .
ex:MainBodyContent	rdf:type	rdf:langString .
ex:ConclusionsContent	rdf:type	rdf:langString .

Τροποποιούμε κι εμπλουτίζουμε τη γνώση σύμφωνα με το RDF Schema:

Ορισμός Λεξιλογίου

ex:Text	rdf:type	rdfs:Class .
ex:TextPart	rdf:type	rdfs:Class .
ex:TextPart	rdfs:subClassOf	ex:Text .
ex:consistsOf	rdf:type	rdfs:Property .
ex:consistsOf	rdfs:domain	ex:Text .
ex:consistsOf	rdfs:range	rdf:Seq .
ex:hasContent	rdf:type	rdfs:Property .
ex:hasContent	rdfs:domain	ex:TextPart .
ex:hasContent	rdfs:range	rdf:langString .

Δεδομένα

			ex:Keimeno	rdfs:label	"Text Title"@en .
			ex:Keimeno	exp:consistsOf	_:Sections .
			_:Sections	rdf:first	ex:Prologue .
			_:Sections	rdf:rest	_:c1 .
ex:Keimeno	rdf:type	rdf:Text .	_:c1	rdf:first	ex:MainBody .
ex:Prologue	rdf:type	ex:TextPart .	_:c1	rdf:rest	_:c2 .
ex:MainBody	rdf:type	ex:TextPart .	_:c2	rdf:first	ex:Conclusions .
ex:Conclusions	rdf:type	ex:TextPart .	_:c2	rdf:rest	rdf:nil .
			ex:Prologue	exp:hasContent	ex:PrologueContent .
			ex:MainBody	exp:hasContent	ex:MainBodyContent .
			ex:Conclusions	exp:hasContent	ex:ConclusionsContent .

3. Ο Κωνσταντίνος γνωρίζει ότι η πρωτεύουσα της Ρωσίας είναι η Μόσχα.

Η γνώση (3) μοντελοποιείται σε RDF (χωρίς χρήση των δυνατοτήτων του RDF Schema) ως εξής:

RDF Model

1ος Τρόπος

ex:Russia	exp:hasCapital	ex:RussianCapital .
ex:RussianCapital	exp:is	ex:Moscow .
ex:Constantinos	exp:knows	ex:RussianCapital .

2ος Τρόπος

ex:s1	rdf:type	rdf:Statement .
ex:s1	rdf:subject	ex:Russia .
ex:s1	rdf:predicate	exp:hasCapital .
ex:s1	rdf:object	ex:Moscow .
ex:Constantinos	exp:knows	ex:s1 .

3ος Τρόπος

ex:Constantinos exp:knows "Η πρωτεύουσα της Ρωσίας είναι η Μόσχα"

Τροποποιούμε κι εμπλουτίζουμε τη γνώση σύμφωνα με το RDF Schema:

Ορισμός Λεξιλογίου

ex:Country	rdf:type	rdfs:Class .
ex:City	rdf:type	rdfs:Class .
ex:Capital	rdf:type	rdfs:Class .
ex:Person	rdf:type	rdfs:Class .
ex:Man	rdf:type	rdfs:Class .
ex:Capital	rdfs:subClassOf	rdfs:City .
ex:Man	rdfs:subClassOf	rdfs:Person .
ex:hasCapital	rdf:type	rdfs:Property .
ex:hasCapital	rdfs:domain	ex:Country .
ex:hasCapital	rdfs:range	ex:Capital .
ex:knows	rdf:type	rdfs:Property .
ex:knows	rdfs:domain	ex:Person .
ex:knows	rdfs:range	rdf:Statement .

Δεδομένα

ex:Constantinos	rdf:type	ex:Man .
ex:Constantinos	exp:knows	ex:s1 .
ex:s1	rdf:subject	ex:Russia .
ex:s1	rdf:predicate	ex:hasCapital .
ex:s1	rdf:object	ex:Moscow .

Ερώτημα 2

Δοθείσες Τριάδες RDF

- (1) `ex:Student` `rdfs:subClassOf` `ex:Person` .
- (2) `ex:John` `rdf:type` `ex:Student` .
- (3) `ex:isFriendOf` `rdfs:subPropertyOf` `ex:knows` .
- (4) `ex:John` `ex:isFriendOf` `ex:Alice` .

Χρήσιμα Αξιώματα

- (A) `rdf:type` `rdfs:range` `rdfs:Class` .

- (4) $\xrightarrow{\text{RDF D2}}$ `ex:isFriendOf` `rdf:type` `rdfs:Property` (5)
- (5) $\xrightarrow{\text{RDFS 6}}$ `ex:isFriendOf` `rdfs:subPropertyOf` `ex:isFriendOf`
- (3) + (4) $\xrightarrow{\text{RDFS 7}}$ `ex:John` `ex:knows` `ex:Alice` (6)
- (1) + (2) $\xrightarrow{\text{RDFS 9}}$ `ex:John` `rdf:type` `ex:Person` (7)
- (6) $\xrightarrow{\text{RDF D2}}$ `ex:knows` `rdf:type` `rdfs:Property` (8)
- (8) $\xrightarrow{\text{RDFS 6}}$ `ex:knows` `rdfs:subPropertyOf` `ex:knows`
- (2) $\xrightarrow{(A)}$ `ex:Student` `rdf:type` `rdfs:Class` (9)
- (7) $\xrightarrow{(A)}$ `ex:Person` `rdf:type` `rdfs:Class` (10)
- (9) $\xrightarrow{\text{RDFS 10}}$ `ex:Student` `rdfs:subClassOf` `ex:Student`
- (10) $\xrightarrow{\text{RDFS 10}}$ `ex:Person` `rdfs:subClassOf` `ex:Person`

Σημειώνεται ότι από όλες τις τριάδες (δεδομένες και παραγόμενες) μπορούν μέσω του κανόνα συμπερασμού RDFS 4 (a και b) να παραχθούν τριάδες της μορφής:

`xxx` `rdf:type` `rdfs:Resource` .

τις οποίες δεν διατυπώνουμε εξατομικευμένα για συντομία.

Ερώτημα 3

1.(a) DBpedia Knowledge Base

```

PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbp: <http://dbpedia.org/property/>

SELECT DISTINCT ?person ?name
WHERE {
    ?person dbo:award dbr:Nobel_Prize_in_Physics .
    ?person dbp:name ?name .
    ?person dbo:almaMater ?university .
    ?university dbo:city ?city .
    ?city dbo:country ?eduCountry .
    ?person dbo:birthPlace ?birthPlace .
    ?birthPlace dbo:country ?birthCountry .

    FILTER (?eduCountry != ?birthCountry) .
    FILTER (lang(?name) = 'en' ) .
}

```

Return List

person	name
http://dbpedia.org/resource/Subrahmanyan_Chandrasekhar	"Subrahmanyan Chandrasekhar"@en
http://dbpedia.org/resource/Charles_K._Kao	"Charles K.Kao"@en
http://dbpedia.org/resource/Owen_Willans_Richardson	"Sir Owen Richardson"@en
http://dbpedia.org/resource/James_Chadwick	"Sir James Chadwick"@en
http://dbpedia.org/resource/David_J._Thouless	"David Thouless"@en
http://dbpedia.org/resource/Yang_Chen-Ning	"Yang Chen-Ning"@en
http://dbpedia.org/resource/Dennis_Gabor	"Dennis Gabor"@en
http://dbpedia.org/resource/Eugene_Wigner	"Eugene Wigner"@en
http://dbpedia.org/resource/Ernst_Ruska	"Ernst Ruska"@en
http://dbpedia.org/resource/Walter_Houser_Brattain	"Walter Houser Brattain"@en
http://dbpedia.org/resource/Lev_Landau	"Lev Landau"@en
http://dbpedia.org/resource/Max_Born	"Max Born"@en
http://dbpedia.org/resource/Maria_Goeppert_Mayer	"Maria Goeppert Mayer"@en
http://dbpedia.org/resource/Albert_A._Michelson	"Albert A. Michelson"@en
http://dbpedia.org/resource/Peter_Grünberg	"Peter Grünberg"@en
http://dbpedia.org/resource/Claude_Cohen-Tannoudji	"Claude Cohen-Tannoudji"@en
http://dbpedia.org/resource/Georges_Charpak	"Georges Charpak"@en
http://dbpedia.org/resource/Arno_Allan_Penzias	"Arno Allan Penzias"@en
http://dbpedia.org/resource/Horst_Ludwig_Störmer	"Horst Ludwig Störmer"@en

1.(b) Wikidata Knowledge Base

```

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>

SELECT DISTINCT ?person ?name
WHERE {
    wd:Q38104 wdt:P1346 ?person .
    ?person wdt:P69 ?education .
    ?education wdt:P17 ?education_country .
    ?person wdt:P19 ?place_of_birth .
    ?place_of_birth wdt:P17 ?country_of_birth .
    ?person rdfs:label ?name .

    FILTER(?education_country != ?country_of_birth) .
    FILTER (lang(?name) = 'en') .
}

```

Return List

person	name
http://www.wikidata.org/entity/Q937	Albert Einstein
http://www.wikidata.org/entity/Q7186	Marie Curie
http://www.wikidata.org/entity/Q35149	Wilhelm Röntgen
http://www.wikidata.org/entity/Q57067	Max von Laue
http://www.wikidata.org/entity/Q57068	Wilhelm Wien
http://www.wikidata.org/entity/Q57077	Ferdinand Braun
http://www.wikidata.org/entity/Q62891	Heike Kamerlingh Onnes
http://www.wikidata.org/entity/Q127234	Albert A. Michelson
http://www.wikidata.org/entity/Q57100	Maria Goeppert Mayer
http://www.wikidata.org/entity/Q131729	Lawrence Bragg
http://www.wikidata.org/entity/Q133232	Gabriel Lippmann
http://www.wikidata.org/entity/Q155786	Dennis Gabor
http://www.wikidata.org/entity/Q2128461	Rainer Weiss

2.(a) DBPedia Knowledge Base

```

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

SELECT ?actor1 ?actor2 (COUNT(?movie) AS ?count)
WHERE {
    ?movie dbo:starring ?actor1 .
    ?movie dbo:starring ?actor2 .
    ?actor1 dbo:spouse ?actor2 .
    FILTER(str(?actor1) < str(?actor2)) .
}
GROUP BY ?actor1 ?actor2
HAVING(COUNT(?movie) > 12)

```

2.(b) Wikidata Knowledge Base

```

PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>

SELECT ?actor1 ?actor2 (COUNT(?movie) AS ?movieCount)
WHERE {
    ?movie wdt:P31 wd:Q11424 .
    ?movie wdt:P161 ?actor1 .
    ?movie wdt:P161 ?actor2 .
    ?actor1 wdt:P26 ?actor2 .
    FILTER(str(?actor1) < str(?actor2))
}
GROUP BY ?actor1 ?actor2
HAVING(?movieCount > 12)

```

DBPedia Return List

actor1	actor2	count
http://dbpedia.org/resource/Ali_Seezan	http://dbpedia.org/resource/Niuma_Mohamed	14
http://dbpedia.org/resource/Gemini_Ganesan	http://dbpedia.org/resource/Savitri_(actress)	38
http://dbpedia.org/resource/Gene_Gauntier	http://dbpedia.org/resource/Jack_J._Clark	45
http://dbpedia.org/resource/Apu_Biswas	http://dbpedia.org/resource/Shakib_Khan	28
http://dbpedia.org/resource/C._R._Vijayakumari	http://dbpedia.org/resource/S._S._Rajendran	26
http://dbpedia.org/resource/Beverly_Bayne	http://dbpedia.org/resource/Francis_X._Bushman	20
http://dbpedia.org/resource/Sessue_Hayakawa	http://dbpedia.org/resource/Tsuru_Aoki	28
http://dbpedia.org/resource/Neetu_Singh	http://dbpedia.org/resource/Rishi_Kapoor	13
http://dbpedia.org/resource/Jayaram	http://dbpedia.org/resource/Parvathy_Jayaram	13
http://dbpedia.org/resource/Mohamed_Manik	http://dbpedia.org/resource/Sheela_Najeeb	15
http://dbpedia.org/resource/Dale_Evans	http://dbpedia.org/resource/Roy_Rogers	13

Wikidata Return List

actor1	actor2	movieCount
http://www.wikidata.org/entity/Q132952	http://www.wikidata.org/entity/Q443063	18
http://www.wikidata.org/entity/Q175571	http://www.wikidata.org/entity/Q36105	16
http://www.wikidata.org/entity/Q44426	http://www.wikidata.org/entity/Q70294	13
http://www.wikidata.org/entity/Q691515	http://www.wikidata.org/entity/Q752610	15
http://www.wikidata.org/entity/Q106482	http://www.wikidata.org/entity/Q274043	14
http://www.wikidata.org/entity/Q310692	http://www.wikidata.org/entity/Q459676	18
http://www.wikidata.org/entity/Q6006839	http://www.wikidata.org/entity/Q6990228	17
http://www.wikidata.org/entity/Q353758	http://www.wikidata.org/entity/Q465186	26
http://www.wikidata.org/entity/Q446526	http://www.wikidata.org/entity/Q708350	14
http://www.wikidata.org/entity/Q1398564	http://www.wikidata.org/entity/Q240527	21
http://www.wikidata.org/entity/Q288970	http://www.wikidata.org/entity/Q70251	14
http://www.wikidata.org/entity/Q1685858	http://www.wikidata.org/entity/Q3103749	14
http://www.wikidata.org/entity/Q1549201	http://www.wikidata.org/entity/Q2851509	28
http://www.wikidata.org/entity/Q33635	http://www.wikidata.org/entity/Q379157	32
http://www.wikidata.org/entity/Q4346250	http://www.wikidata.org/entity/Q540098	14
http://www.wikidata.org/entity/Q290996	http://www.wikidata.org/entity/Q53111	16
http://www.wikidata.org/entity/Q4907508	http://www.wikidata.org/entity/Q8776767	21
http://www.wikidata.org/entity/Q1533336	http://www.wikidata.org/entity/Q25055	16
http://www.wikidata.org/entity/Q1243816	http://www.wikidata.org/entity/Q459637	18
http://www.wikidata.org/entity/Q1958523	http://www.wikidata.org/entity/Q512435	13
http://www.wikidata.org/entity/Q3028225	http://www.wikidata.org/entity/Q451116	15
http://www.wikidata.org/entity/Q201617	http://www.wikidata.org/entity/Q697816	15
http://www.wikidata.org/entity/Q1704158	http://www.wikidata.org/entity/Q254289	17
http://www.wikidata.org/entity/Q12975497	http://www.wikidata.org/entity/Q7668111	16
http://www.wikidata.org/entity/Q1769734	http://www.wikidata.org/entity/Q1801276	18
http://www.wikidata.org/entity/Q16561332	http://www.wikidata.org/entity/Q2269100	13
http://www.wikidata.org/entity/Q4777709	http://www.wikidata.org/entity/Q4864991	14
http://www.wikidata.org/entity/Q4779123	http://www.wikidata.org/entity/Q5223859	75
http://www.wikidata.org/entity/Q3844045	http://www.wikidata.org/entity/Q6701100	13
http://www.wikidata.org/entity/Q3595617	http://www.wikidata.org/entity/Q5276736	18
http://www.wikidata.org/entity/Q15956720	http://www.wikidata.org/entity/Q93996645	13
http://www.wikidata.org/entity/Q2615937	http://www.wikidata.org/entity/Q3183784	21
http://www.wikidata.org/entity/Q14476290	http://www.wikidata.org/entity/Q4907192	30

Προβλήματα που εμφανίστηκαν

- Η βάση δεδομένων DBpedia δεν παρουσιάζει σε μεγάλο τμήμα της συμπαγή δομή, καθώς κάποιες οντότητες έχουν ορισμένες ιδιότητες, ενώ άλλες όχι, με αποτέλεσμα να είναι δύσκολη η διαμόρφωση query που να επιστρέφει όλες τις επιθυμητές οντότητες. Παρατηρείται μια μορφή "ανοργανωσίας".
- Ως εκ τούτου, τα αποτελέσματα από τις δύο βάσεις δεδομένων διαφέρουν σημαντικά (για παράδειγμα εμφανίζονται μόνο 2 κοινά στοιχεία στο υποερώτημα 1).
- Σημειώνεται ότι στο query του υποερωτήματος 1 (στο query της DBPedia) χρησιμοποιείται ο ονοματόχωρος "dbp" μονάχα για την ιδιότητα "dbp:name", ώστε να λάβουμε και το όνομα της κάθε οντότητας. Δεν χρησιμοποιήθηκε το "dbo:Name", διότι οι οντότητες είχαν το attribute "dbr:name" αντ' αυτού.
- Στο SPARQL Endpoint της DBPedia το τμήμα "HAVING(?count > 12)" γινόταν αντιληπτό σαν compile error και χρειάστηκε να αντικατασταθεί από το "HAVING(COUNT(?movie) > 12)".

Σημείωση - Σχόλιο

Στο SPARQL query του υποερωτήματος (1.b) χρησιμοποιείται ο όρος "wd:Q38104 wdt:P1346 ?person" που σημαίνει ότι αναζητούμε από την σελίδα των βραβείων νόμπελ (wd:Q38104) τους νικητές (wdt:P1346). Αυτό ενδέχεται (στην συγκεκριμένη περίπτωση έχει ελεγχθεί ότι συμβαίνει) να επιστρέφει διαφορετικά αποτελέσματα από αυτά που θα επιστρέφονταν αν αντ'αυτού του όρου αναζητούσαμε να δούμε αν έχει κερδίσει κάποιος βραβείο νόμπελ από την σελίδα του (δηλαδή με τον όρο "?person wdt:P166 wd:Q38104").

Ερώτημα 4

```
SELECT ?uri ?score
WHERE {
  {
    SELECT (COUNT(?temp) AS ?result)
    WHERE {
      ?temp a <http://example.org/Company> .
      ?temp <http://example.org/isLocatedIn> <http://dbpedia.org/resource/Greece> .
    }
  }
  {
    SELECT ?uri
    WHERE {
      ?uri a <http://example.org/Company> .
      ?uri <http://example.org/isLocatedIn> <http://dbpedia.org/resource/Greece> .
    }
  }
  BIND((1/?result) AS ?score) .
}
```