SPARQL
More Operators

Nicolò Pratelli

Istituto di Scienza e Tecnologie dell'Informazione -
Consiglio Nazionale delle Ricerche – Pisa

A.A. 2024-2025

More SPARQL Operators

Today we will use the following SPARQL operators:

1. the `COUNT` operator is used to count results, grouped by operator `GROUP BY`
2. the `SUM` aggregator is used to compute the sum over the values of a variable
3. The `MIN/MAX` aggregator is used to compute the min/max over the values of a variable
4. the `AVG` aggregator is used to compute the average over the values of a variable
5. the `HAVING` operator is used to filter on an aggregate

More SPARQL Operators

Today we will use the following SPARQL operators:

6. the `FILTER NOT EXISTS` operator is used to filter results based on the non-existence of a graph pattern
7. the `MINUS` operator is used to subtract results based on a specific graph pattern
8. the `BIND` operator is used to store the result of an expression in a variable
9. the `OFFSET` modifier causes the solutions generated to start after the specified number of solutions.

Counting and Grouping Results

```
# Extract the type of all resources, group the results by
class and count the number of resources for each class
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?class (COUNT(?class) AS ?count)
WHERE {
    ?iri rdf:type ?class .
} GROUP BY ?class
```

class	count
http://purl.org/dc/dcmitype/Text	8
http://xmlns.com/foaf/0.1/Person	6

SUM Aggregator

```
# Compute the sum of the prices of all the print
editions of the Divine Comedy
PREFIX : <http://example.org/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?label (SUM(?price) AS ?sum)
WHERE {
     wd:Q40185 rdfs:label ?label ;
            :hasPrintEdition ?printEdition .
     ?printEdition :price ?price .
} GROUP BY ?label
```

label	sum
Divine Comedy	235

AVG Aggregator

```
# Compute the average of the prices of all the print
editions of the Divine Comedy
PREFIX : <http://example.org/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?label (AVG(?price) AS ?avg)
WHERE {
     wd:Q40185 rdfs:label ?label ;
            :hasPrintEdition ?printEdition .
     ?printEdition :price ?price .
} GROUP BY ?label
```

label	avg
Divine Comedy	58.75

MIN and MAX Aggregator

```
# Compute the min/max price among all the print
editions of the Divine Comedy
PREFIX : <http://example.org/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?label (MIN/MAX(?price) AS ?min/?max)
WHERE {
    wd:Q40185 rdfs:label ?label ;
            :hasPrintEdition ?printEdition .
    ?printEdition :price ?price .
} GROUP BY ?label
```

label	min
Divine Comedy	40

label	max
Divine Comedy	80

HAVING Operator

```
# Find the label of works whose print editions have an
average price > 50
PREFIX : <http://example.org/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?label (AVG(?price) AS ?avg)
WHERE {
     ?work rdfs:label ?label ;
            :hasPrintEdition ?printEdition .
     ?printEdition :price ?price .
} GROUP BY ?label HAVING (?avg > 50)
```

label	avg
Tresor	65
Divine Comedy	58.75

Filtering on Non-Existence

```
# Extract IRI of all people without email address
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?iri
WHERE {
    ?iri rdf:type foaf:Person .
    FILTER NOT EXISTS
    {?iri foaf:mbox ?mbox . }
}
```

iri
http://www.wikidata.org/entity/Q4653
http://www.wikidata.org/entity/Q187349

FILTER NOT EXISTS is **inside** the WHERE clause

Subtracting Results

```
# Extract IRI of all people without email address
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?iri
WHERE {
    ?iri rdf:type foaf:Person .
    MINUS
    {?iri foaf:mbox ?mbox . }
}
```

iri
http://www.wikidata.org/entity/Q4653
http://www.wikidata.org/entity/Q187349

MINUS is **inside** the WHERE clause

BIND Operator

```
# Add 10 to the price of each print edition and check the new average
PREFIX : <http://example.org/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
SELECT ?label (AVG(?newPrice) AS ?avg)
WHERE {
    wd:Q40185 rdfs:label ?label ;
              :hasPrintEdition ?printEdition .
    ?printEdition :price ?price .
BIND ((?price + 10) AS ?newPrice)
} GROUP BY ?label
```

label	avg
Divine Comedy	68.75

OFFSET modifier

```
# Find only 3 among the works of Dante in alphabetical
order, starting after the first one
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dcterms:  <http://purl.org/dc/terms/>
SELECT ?label
WHERE {
     ?work rdfs:label ?label ;
           dcterms:creator wd:Q1067 .
} ORDER BY ?label LIMIT 3 OFFSET 1
```

label
De Vulgari Eloquentia
Divine Comedy
Monarchia

Instructions for Exercises

- Download the knowledge base from Moodle, open it in a text editor and look at its contents
- Launch the Blazegraph JAR file
- Go to the Blazegraph web interface and define a new namespace called “Seminar4” (no spaces!)
- Upload the knowledge base into **Blazegraph**
- Execute queries on the knowledge base
- Request a **review** when finished

Exercises

Write a query to display in Blazegraph:

1. the number of texts created by Dante Alighieri, grouped by language
2. labels of all texts created by Dante Alighieri, minus those written in Latin
3. IRIs and labels of all people who do not have children
4. distinct labels of people who did not write any texts in Italian nor Latin
5. Retrieve the IRI and the price of all the print editions of the Divine Comedy, ordered by price
6. Retrieve the IRI and the price of all the print editions that have been written in Italian, filtered by price < 50

Exercises

Write a query to display in Blazegraph:

7. Compute the total price of print editions written in Italian or English
8. Compute the average price of print editions written after 1860
9. Retrieve the labels of authors of works whose print editions have an average price > 60 (use GROUP BY and HAVING)
10. Retrieve the texts that are not print editions (alphabetically ordered), starting from the third result
11. Compute the average price of Divine Comedy print editions if the price of each print edition is reduced by 8 (use BIND without GROUP BY)
12. Retrieve the authors who have created more than 2 works, excluding the print editions