



342436 SEMANTIC WEB TECHNOLOGIES

Assoc. Prof. Ngamnij Archint

Chapter6-1
Introduction to SPARQL



Introduction to SPARQL

- **Topics**

- Introducing Inference Engine or Reasoner
- SPARQL-SPARQL Protocol and RDF Query Language

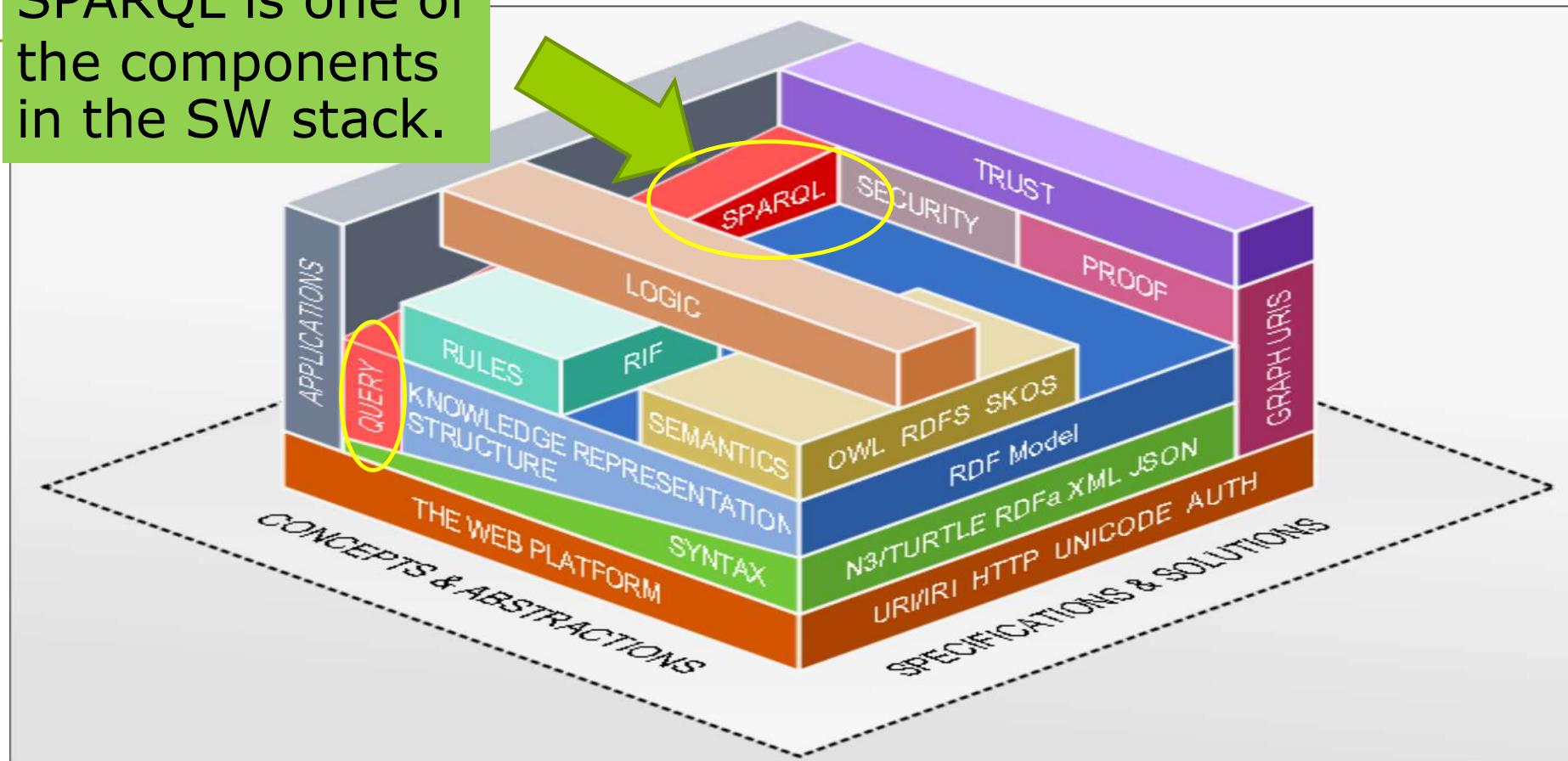
SPARQL



SPARQL Protocol and RDF Query Language

Semantic Web Technology Stack Visualization

SPARQL is one of the components in the SW stack.



SPARQL- A Query Language for RDF

▫ SPARQL-Protocol and RDF Query Language

- จะเป็นภาษาที่ถูกสร้างขึ้นมาเพื่อใช้สืบค้นข้อมูลที่อยู่ในรูปแบบโครงสร้าง triple ใน RDF graph
- จะสามารถทำงานอยู่บนโปรโตคอล HTTP.
- เป็นภาษามาตรฐานที่ถูกแนะนำโดย W3C (โดย version ล่าสุดคือ SPARQL 1.1 ซึ่งถูกนำเสนอเมื่อปี Mar 2013)
- จะมีโครงสร้างคล้ายกับ SQL (เพราะมีการใช้รูปแบบของคำสั่ง SELECT, FROM, WHERE เหมือนกัน)



SPARQL- A Query Language for RDF

- เราสามารถใช้ SPARQL ในเครื่องมือต่างๆ เช่น **RDF Triplestore** และ **Ontology Editor tools** เช่น Protégé.

The screenshot shows the Protégé ontology editor interface. A large green arrow points from the text above down to the 'SPARQL Query' tab in the top navigation bar. The 'SPARQL Query' tab is highlighted in blue. The main workspace shows an ontology with individuals: 49778888, Bachelor, and Master. The 'SPARQL Query' tab contains the following code:

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX dg: <http://www.university.edu/degree/>
PREFIX per: <http://www.university.edu/person/>

SELECT ?x ?y
WHERE {
    per:49778888 per:name ?x .
    per:49778888 :hasDegree ?y
}
```

Below the query is an 'Execute' button. The results table shows one row:

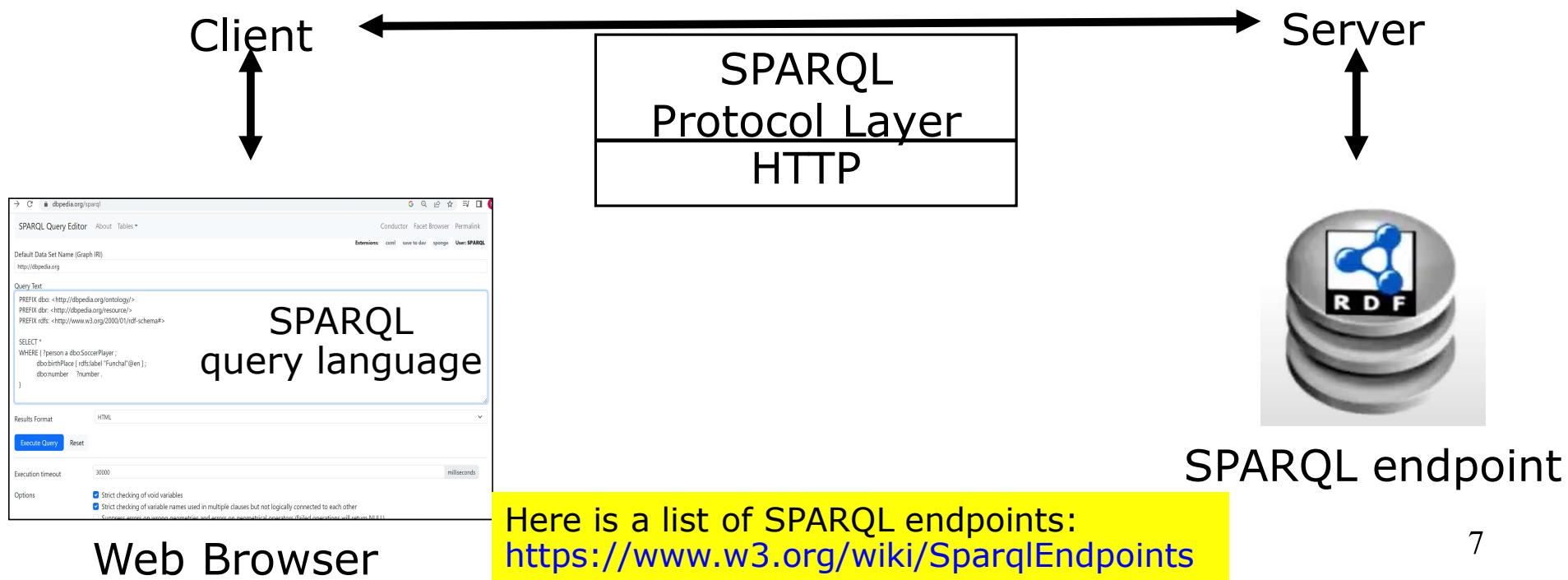
?x	?y
Willy John^^xsd:string	dg:Bachelor

At the bottom, it says '1 results'.



SPARQL- A Query Language for RDF

- อย่างไรก็ตาม จะพบว่ามีหลายองค์กรที่ให้บริการ API เพื่อให้คนอื่นใช้ Web browser ของตนเข้ามาสืบค้นข้อมูลจากฐานข้อมูล RDF ขององค์กรเหล่านั้นได้โดยผ่านทางสิ่งที่เรียกว่า **SPARQL endpoint** โดยผู้ใช้สามารถสร้าง SPARQL โดยเรียกใช้ interface ของ **SPARQL endpoint** ของผู้ให้บริการได้โดยตรงเลย (บน Web browser ของผู้ใช้เอง)





SPARQL- A Query Language for RDF

□ Here is a list of SPARQL endpoints:

<https://www.w3.org/wiki/SparqlEndpoints>

w3.org/wiki/SparqlEndpoints

Log in

W3C®

Page Discussion Read View source View history Search

SparqlEndpoints

see also: [SPARQL](#)

In addition to the list below, Mondeca provides a [SPARQL endpoint uptime service](#) which monitors the availability of all SPARQL endpoints that are cataloged in [CKAN](#). A similar service is provided by Vienna University.

Currently Alive SPARQL Endpoints

(alphabetical. let's avoid [PoorMansHypertext](#) and in-your-face URIs, please)

Project	status	SPARQL endpoint	Webform	comment
Wikidata	(2017-02-23) alive	endpoint	GUI	See also SPARQL federation input
BBC Programmes and Music	(2010-06-29) alive	endpoint	Ajax based Visual Query Builder	Powered by OpenLink Virtuoso ; also supports Faceted Browsing and Exploration
Bio2RDF	(2010-01-07) alive	List of 40 SPARQL endpoints	n/a	uses OpenLink Virtuoso
BioGateway	(2010-01-07)	endpoint	webforms	BioGateway provides many parameterizable SPARQL queries, both biological as ontological, on RDF graphs that were



SPARQL- A Query Language for RDF

- One of the famous SPARQL endpoints is **DBPedia**.

w3.org/wiki/SparqlEndpoints

		alive			
	data.gov	(2010-05-22) alive	endpoint	webform	uses OpenLink Virtuoso
	data.gov.uk	(2010-02-04) alive	endpoint	webform	The data.gov.uk endpoint
	DBLP Bibliography Database published through D2R Server (Freie Universität Berlin)	(2010-01-07) alive	endpoint	webform (Maybe only Firefox)	The DBLP database provides bibliographic information on major computer science journals and conference proceedings.
	DBpedia (University of Mannheim, Universität Leipzig, OpenLink Software)	(2010-01-07) alive	endpoint	SNORQL webform (Firefox/Safari/Opera); Ajax based Visual Query Builder	dbpedia.org is a community effort to extract structured information from periodic Wikipedia dumps and to make this information available on the Web. It is served to the public via a live instance of OpenLink Virtuoso, and also offers Faceted Browsing and Exploration
	DBpedia-live (Universität Leipzig, University of Mannheim, OpenLink Software)	(2010-01-07) alive	endpoint	webform	Based on, now parallel to, and soon to replace the existing dbpedia.org data sets, DBpedia-Live is constantly updated, based on Wikipedia change-feeds. It is served to the public via a live instance of OpenLink Virtuoso, and also offers Faceted Browsing and Exploration
	German DBpedia (AG Corporate Semantic Web, Freie Universität Berlin)	(2008-10-15) alive	endpoint	site	de.dbpedia.org is the German language chapter of DBpedia
	DBpedia Live German (AG Corporate Semantic Web, Freie Universität Berlin)	(2008-10-15) alive	endpoint	site	de.dbpedia.org is the German language chapter of DBpedia
	Spanish DBpedia (Universidad Autónoma de Madrid, Universidad	(2011-04-04)	endpoint	site	es.dbpedia.org is the Spanish chapter of DBpedia



Example of SPARQL Endpoint

https://dbpedia.org/sparql

dbpedia.org/sparql

SPARQL Query Editor About Tables ▾

Default Data Set Name (Graph IRI)
http://dbpedia.org

Query Text

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
SELECT *
WHERE { ?player a dbo:SoccerPlayer;
    dbpedia2:birthPlace dbr:Brazil
}
```

คำสั่งนี้คือให้สืบค้นว่ามีนักฟุตบอล (soccer player) คนไหนบ้างที่มีสถานที่เกิด (birthplace) อยู่ในประเทศ Brazil.

Execute Query Reset

Conductor Facet Browser Permalink

- ตัวอย่างนี้คือตัวอย่างของการใช้ interface ของ SPARQL endpoint ขององค์กร DBpedia
- ผู้ใช้สามารถเข้าถึง endpoint นี้ (<https://dbpedia.org/sparql>) และจะมี interface ให้ผู้ใช้สามารถป้อนคำสั่ง SPARQL เพื่อสอบถามชุดข้อมูล RDF (RDF dataset) ที่จัดเก็บไว้ในฐานข้อมูลของ DBpedia ได้โดยตรง
- เมื่อผู้ใช้กดปุ่ม **Execute Query** โปรแกรม API ของ DBpedia ก็จะถูกเรียกใช้เพื่อส่งคำสั่ง SPARQL ของผู้ใช้ไปสืบค้นข้อมูล RDF จาก DBpedia และส่งคืนผลลัพธ์ให้กับผู้ใช้ต่อไปได้

Result from SPARQL Endpoint

SPARQL | HTML5 table

player
http://dbpedia.org/resource/Cabeção_(footballer)
http://dbpedia.org/resource/Cabore
http://dbpedia.org/resource/Cadu_(footballer,_born_1974)
http://dbpedia.org/resource/Cadu_(footballer,_born_1982)
http://dbpedia.org/resource/Caetano_Calil
http://dbpedia.org/resource/Cafuringa
http://dbpedia.org/resource/Café_(futsal_player)
http://dbpedia.org/resource/Cahê
http://dbpedia.org/resource/Caio_(footballer,_born_1980)
http://dbpedia.org/resource/Caio_De_Cenco
http://dbpedia.org/resource/Caio_Japa
http://dbpedia.org/resource/Caio_Mendes
http://dbpedia.org/resource/Cairo_Lima

• อันนี้จะเป็นผลลัพธ์ที่ได้จากการสืบค้นของคำสั่งในหน้าที่แล้ว และถ้าเรา click ที่แต่ละ URI (ซึ่งเป็น resource ที่เกี่ยวกับรายชื่อนักฟุตบอล) ข้อมูลรายละเอียดของนักฟุตบอลคนนั้น ก็จะถูกนำมาแสดงอีกทีในหน้าตัดไป



Result from SPARQL Endpoint

This is the **URI** resource of
<http://dbpedia.org/resource/Cabore>

These are **Properties** and **values** of the **properties** for the “Cabore” resource.

Property	Value
dbo:Person/height	<ul style="list-style-type: none">• 187.0
dbo:abstract	<p>هو لاعب كرة قدم كان يلعب كمهاجم ،(مواليد 19 فبراير 1980) إفيرaldo دي جيسوس بيريرا . هو لاعب كرة قدم كان يلعب كمهاجم ،(مواليد 19 فبراير 1980) Cabore (بالبرتغالية) إفيرaldo دي جيسوس بيريرا .</p>





SPARQL – Using SPARQL endpoint of DBpedia

- ถ้าเราใช้ DBpedia SPARQL endpoint เราจึงสามารถเลือกรูปแบบการแสดงผลได้หลากหลายรูปแบบดังนี้

The screenshot shows the DBpedia SPARQL Query Editor interface. In the 'Results Format' section, a dropdown menu is open, listing various options: Auto, HTML, HTML (Faceted Browsing Links), Spreadsheet, XML, JSON, Javascript, Turtle, RDF/XML, N-Triples, CSV, and TSV. The 'HTML' option is highlighted with a blue selection bar. A large green arrow points from the text in the yellow callout box to this highlighted 'HTML' option.

Default Data Set Name (Graph IRI)
http://dbpedia.org

Conductor Facet Browser Permalink

Extensions: cxml save to daw sponge User: SPARQL

Query Text

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT *
WHERE {
    ?author rdf:type dbo:Person
    dbo:notableWork ?work .
}
LIMIT 10
```

Results Format

Execute Query Reset

Execution timeout 30000 milliseconds

เช่นถ้าเราคลิกที่ HTML ก็จะมี การนำข้อมูลมาแสดงในรูปแบบของเอกสาร HTML



SPARQL –Using SPARQL endpoint of DBpedia

author	work
http://dbpedia.org/resource/Cahit Zarifoğlu	http://dbpedia.org/resource/Yaşamak
http://dbpedia.org/resource/Caitlin Rother	http://dbpedia.org/resource/Lost Girls (book)
http://dbpedia.org/resource/Caitlín R. Kiernan	http://dbpedia.org/resource/The Drowning Girl
http://dbpedia.org/resource/Calder Willingham	http://dbpedia.org/resource/The Graduate
http://dbpedia.org/resource/Caleb Carr	http://dbpedia.org/resource/The Italian Secretary
http://dbpedia.org/resource/Caleb Carr	http://dbpedia.org/resource/The Alienist
http://dbpedia.org/resource/Caleb Carr	http://dbpedia.org/resource/The Angel of Darkness
http://dbpedia.org/resource/Caleb Carr	http://dbpedia.org/resource/Casing the Promised Land
http://dbpedia.org/resource/Caleb Carr	http://dbpedia.org/resource/Killing Time (Caleb Carr novel)
http://dbpedia.org/resource/Camilo José Cela	http://dbpedia.org/resource/The Hive (Cela novel)

ในการนี้เราเลือกให้แสดงผลลัพธ์ในรูปแบบ HTML ซึ่งก็จะมีการแปลงอักษรให้อยู่ในรูปแบบของตารางซึ่งจะเป็นรูปแบบปกติที่เราจะได้เจอตลอด



Another SPARQL Endpoint

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
```

```
SELECT * WHERE {
  ...
}
```

Results:

https://dbpedia.org/snorql

- ตัวอย่างนี้ก็จะเป็นอีกตัวอย่างของ SPARQL endpoint ที่ยอมให้ผู้ใช้ทั่วไปสามารถสืบค้นข้อมูลจากฐานข้อมูล DBpedia ได้
- Endpoint ตัวอย่างนี้จะมีการสร้าง **basic PREFIX** ให้กับผู้ใช้โดยอัตโนมัติ แต่เรา ก็ยังสามารถสร้างเองเพิ่มเติมได้อีกด้วย

Powered by [OpenLink Virtuoso](#) and [dbpedia](#)



Another SPARQL Endpoint

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
SELECT *
WHERE { ?player a dbo:SoccerPlayer;
          dbpedia2:birthPlace dbr:Brazil
}
```

Results:

SPARQL results:

player
:Cabeção_(footballer)
:Cabore
:Cadu_(footballer,_born_1974)
:Cadu_(footballer,_born_1982)
:Caetano_Calil
:Cafuringa
:Café_(futsal_player)
:Cahê
:Caio_(footballer,_born_1980)
:Caio_De_Cenço
:Caio_Japa
:Caio_Mendes
:Cairo_Lima
:Caju_(footballer,_born_1995)
:Caleb_(footballer)

เมื่อเรากดปุ่ม “Go” ก็จะมีการแสดงผลลัพธ์
ออกมากทางด้านล่างนี้

For Queries we need Variables

- SPARQL จะให้เราสามารถสร้าง variables ขึ้นมา ซึ่งจะต้องขึ้นต้นด้วยเครื่องหมาย ? เพื่อใช้เก็บผลลัพธ์จากการสืบค้นข้อมูล เช่น **?title**, **?author**, **?published**
- โครงสร้างของ SPARQL จะคล้ายกับ SQL
 - ในการสืบค้นข้อมูลเพื่อนำมาเก็บในตัวแปรข้างต้น จะต้องใช้คำสั่ง **SELECT**
 - เช่น **SELECT ?title ?author ?published**

- คำสั่ง **SELECT** นี้จะส่งผลลัพธ์ออกมาในรูปแบบของตารางเช่น

?title	?author	?published
Ontology Engineering	Pascal Hitzler	2016
Linked Data	David Wood	2013
A Semantic Web Primer	Grigoris Antoniou	2012

SPARQL Graph Pattern Matching

- การสืบค้นด้วย SPARQL จะเป็นการสืบค้นบนพื้นฐานของ **graph pattern matching**
- A Graph Pattern (Triple Pattern) ก็คือ RDF Triple ซึ่งจะประกอบด้วย (Subject, Property, Object)
- **Example:** ถ้าต้องการค้นหา รายชื่อประเทศ (country) และเมืองหลวง (capital) ของแต่ละประเทศ

```
PREFIX dbo:<http://dbpedia.org/ontology/>
SELECT *
WHERE { ?country dbo:capital ?capital }
```

* จะหมายถึงการนำข้อมูลในตัวแปร ?country และ ?capital ออกมาระดับทุกตัวแปร

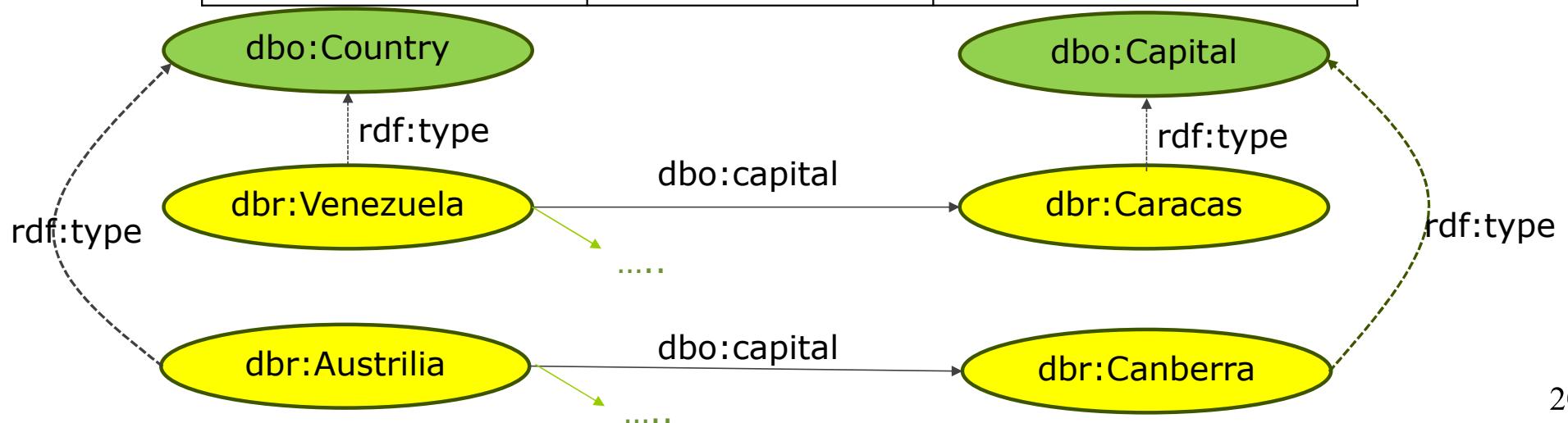
SPARQL Graph Pattern Matching

- สมมติว่าในฐานข้อมูล DBpedia มีการเก็บข้อมูลในรูปแบบของ RDF Graph dataset ที่อยู่ในรูปแบบของ triple ดังต่อไปนี้

Subject	Property	Object
dbr:Venezuela	rdf:type	dbo:Country
dbr:Venezuela	dbo:capital	dbr:Caracas
dbr:Venezuela	dbo:currency	dbr:Venezuelan_bolívar
dbr:Australia	rdf:type	dbo:Country
dbr:Australia	dbo:capital	dbr:Canberra
dbr:Australia	dbo:language	dbr:Australian_English
...

SPARQL Graph Pattern Matching

Subject	Property	Object
dbr:Venezuela	rdf:type	dbo:Country
dbr:Venezuela	dbo:capital	dbr:Caracas
dbr:Venezuela	dbo:currency	dbr:Venezuelan_bolívar
dbr:Australia	rdf:type	dbo:Country
dbr:Australia	dbo:capital	dbr:Canberra
dbr:Australia	dbo:language	dbr:Australian_English
...



SPARQL Graph Pattern Matching

?country และ ?capital เป็น variables ที่จะถูกใช้ในการเก็บผลลัพธ์ที่สืบค้นได้จาก triple ข้างล่างนี้ ซึ่งเรายังสามารถใช้ชื่อตัวแปรอื่นก็ได้เช่น ?a หรือ ?b เป็นต้น

Triple Pattern

?country dbo:capital ?capital

ในตัวอย่างการสืบค้นนี้ เราต้องการค่าข้อมูลที่เป็นประธาน (Subject) ของ property dbo:capital มาเก็บไว้ที่ตัวแปร ?country และนำค่าข้อมูลที่ทำหน้าที่เป็นกรรม (object) ของ property dbo:capital มาเก็บไว้ในตัวแปร ?capital

Subject	Property	Object
dbr:Venezuela	rdf:type	dbo:Country
dbr:Venezuela	dbo:capital	dbr:Caracas
dbr:Venezuela	dbo:currency	dbr:Venezuelan_bolívar
dbr:Australia	rdf:type	dbo:Country
dbr:Australia	dbo:capital	dbr:Canberra
dbr:Australia	dbo:language	dbr:Australian_English
...



SPARQL Graph Pattern Matching

← → ⌂ dbpedia.org/snorql/?query=PREFIX+dbo%3A<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0D%0A++++SELECT-

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX dbpedia: <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dbo: <http://dbpedia.org/ontology/>
```

```
SELECT *
WHERE { ?country dbo:capital ?capital }
```

Results: Browse Go! Reset

SPARQL results:

- :Caboloan ↗
- :Caer_Gwinntguic ↗
- :Cai_(state) ↗
- :Cai_(state) ↗
- :Cai_(state) ↗
- :Caliphate_of_Córdoba ↗
- :Cambodia ↗
- :Cambodia_(1953–1970) ↗
- :Cameroon ↗
- :Canada ↗
- :Canton_10 ↗
- :Canton_10 ↗
- :Canton_of_Baden ↗
- :Canton_of_Cartagena ↗
- :Canton_of_Fricktal ↗

country

- :San_Carlos,_Pangasinan ↗
- :Venta_Belgarum ↗
- :Fengtai_County ↗
- :Shangcai_County ↗
- :Xincai_County ↗
- :Córdoba,_Spain ↗
- :Phnom_Penh ↗
- :Phnom_Penh ↗
- :Yaoundé ↗
- :Ottawa ↗
- :Livno ↗
- :Tomislavgrad ↗
- :Baden,_Switzerland ↗
- :Cartagena,_Spain ↗
- :Rheinfelden_(Aargau) ↗

capital

ค่าข้อมูลที่เป็นรายชื่อของประเทศทั้งหมดจะถูกดึงขึ้นมาเก็บไว้ในตัวแปร **?country** และถูกนำมาแสดงยังหน้าจอข้างล่างด้านซ้าย และค่ารายชื่อเมืองหลวงของแต่ละประเทศก็จะถูกดึงขึ้นมาเก็บไว้ในตัวแปร **?capital** และถูกนำมาแสดงยังหน้าจอข้างล่างด้านขวา

ซึ่งชื่อตัวแปรคือ “**country**” ที่อยู่ข้างหลังเครื่องหมาย ? และชื่อตัวแปร “**capital**” ที่อยู่ข้างหลัง ? ก็จะถูกนำมาแสดงอยู่บนหน้าของตารางผลลัพธ์ในหน้าจอด้านล่างนี้ด้วย

SELECT * หมายถึงต้องการให้แสดงผลลัพธ์ซึ่งได้แก่ข้อมูลในตัวแปรทุกตัวอย่างมาให้เห็นในหน้าจอ

SPARQL Complex Query Pattern

- SPARQL Graph Patterns ยังสามารถเขียนในรูปแบบที่ซับซ้อนที่เป็นการเชื่อมโยงโครงสร้างกราฟแต่ละ triple เข้าด้วยกัน ดังตัวอย่างต่อไปนี้
- **Example:** ต้องการค้นหาชื่อประเทศ (country) และเมืองหลวง (capital) ของแต่ละประเทศ รวมทั้งจำนวนประชากร (population total) ของแต่ละประเทศด้วย

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

```
SELECT *
```

```
WHERE { ?country dbo:capital ?capital .  
       ?country dbo:populationTotalRanking ?populationrank  
 }
```

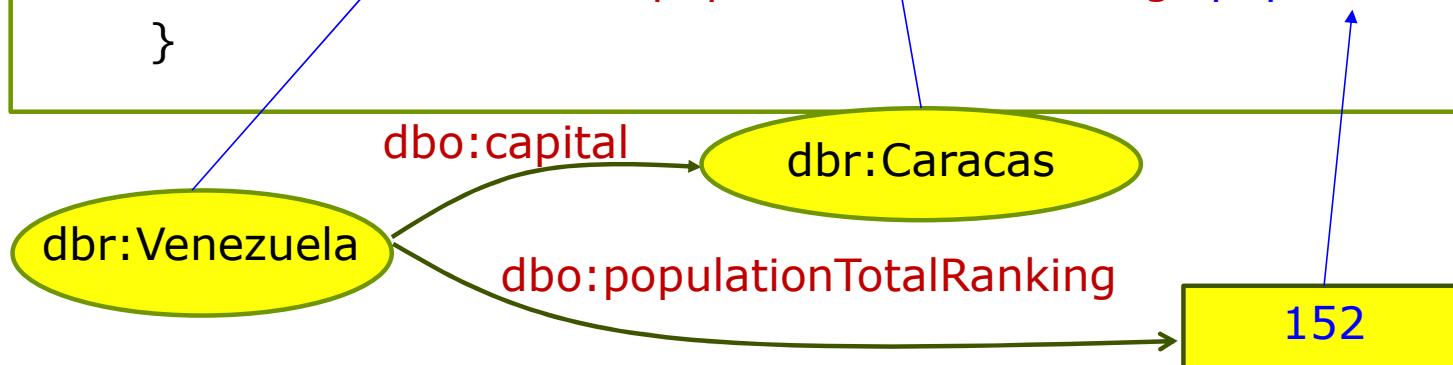
เราสามารถใช้เครื่องหมาย “.” แทนคำสั่ง “and” เพื่อเชื่อมโยงคำสั่งที่มีมากกว่า 1 triple ได้

SPARQL Complex Query Pattern

- เรายังสามารถเขียน SPARQL ให้อยู่ในรูปแบบของ turtle ได้ด้วย

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

```
SELECT *
WHERE { ?country dbo:capital ?capital ;
         dbo:populationTotalRanking ?populationrank }
```



SPARQL จะพยายาม
 match แต่ละ triple ที่
 ออยู่ใน WHERE ของ
 SPARQL กับ triples
 ต่างๆ ที่อยู่ใน graph
 ถ้าเจอ triple ที่ตรงกัน
 ก็จะนำผลลัพธ์จาก
 graph มาเก็บไว้ในตัว
 แปรและแสดงผล
 ออกมากทีหน้าจอ



SPARQL Complex Query Pattern

← → C

dbpedia.org/snorql/?query=PREFIX+dbo%3A<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0D%0A%

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
```

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

```
SELECT *
WHERE {?country dbo:capital ?capital ;
         dbo:populationTotalRanking ?populationrank
}
```

Results:

SPARQL results:

country	capital	populationrank
:Cambodia	:Phnom_Penh	72
:Cameroon	:Yaoundé	51
:Canada	:Ottawa	37
:Qatar	:Doha	139
:Romania	:Bucharest	63
:San_Marino	:City_of_San_Marino	191
:Saudi_Arabia	:Riyadh	40
:Omani_Empire	:Rustaq	52
:Omani_Empire	:Muscat	52
:Omani_Empire	:Stone_Town	52
:Belarus	:Minsk	96
:Belgium	:City_of_Brussels	82
:Belize	:Belmopan	175
:Benin	:Porto-Novo	74
:Bhutan	:Thimphu	165

SPARQL Complex Query Pattern

- ถ้าต้องการค้นหาว่า หนังสือ “Harry_Potter” มีสำนักพิมพ์ (publisher) คืออะไร และมีผู้แต่ง (author) คือใคร และวันเกิด (birthdate) ของผู้แต่งนั้นคือวันที่เท่าไร

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

```
SELECT *
WHERE { :Harry_Potter dbp:publisher ?publisher ;
         dbp:author ?author .
         ?author dbo:birthDate ?birthDate
}
```

ถ้าจะมีการเปลี่ยน
ประชานเป็นตัวใหม่ใน
triple ถัดไป อย่าลืมใช้
ก่อนขึ้น triple ตัวใหม่





SPARQL Complex Query Pattern

← → ⌂ 🔒 dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0APREFIX+dbp%0D%0APREFIX+owl%0D%0APREFIX+xsd%0D%0APREFIX+rdfs%0D%0APREFIX+rdf%0D%0APREFIX+foaf%0D%0APREFIX+dc%0D%0APREFIX+:%0D%0APREFIX+dbpedia2%0D%0APREFIX+dbpedia%0D%0APREFIX+skos%0D%0APREFIX+dbo%0D%0APREFIX+dbp%0D%0APREFIX+:%0D%0ASELECT+*%0D%0AWHERE+{:Harry_Potter+dbp:publisher+?publisher+.%0D%0A+ +dbp:author+?author+.%0D%0A+ +?author+dbo:birthDate+?birthDate+}%0D%0A

Results:

SPARQL results:

publisher	author	birthDate
:Bloomsbury_Publishing	:J._K._Rowling	"1965-07-31"^^xsd:date
:Scholastic_Press	:J._K._Rowling	"1965-07-31"^^xsd:date

SPARQL Complex Query Pattern

- ในกรณีที่สืบค้นข้อมูลต้องมีการเชื่อมต่อกันระหว่าง triple หลาย triple แต่บางครั้งก็อาจมีข้อมูลบางตัวในสืบค้นที่เราคิดว่าไม่สำคัญและไม่ต้องการนำมาแสดงผล เราสามารถใช้ **Blank node** เพื่อแทนข้อมูลที่ไม่สำคัญตัวนั้นได้ ดังตัวอย่างต่อไปนี้

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

```
SELECT ?person ?awardname ?awardfor
WHERE {
    ?person dbo:award [ dbp:name ?awardname ;
                        dbp:awardFor ?awardfor ]
}
```

เช่นตัวอย่างนี้เราต้องการค้นหาว่ามีใคร (**person**) บ้างที่ได้รับรางวัล (**award**) และรางวัลนั้นมีชื่อว่าอะไร (**awardname**) และรางวัลนั้นเป็นรางวัลเกี่ยวกับอะไร (**awardfor**)

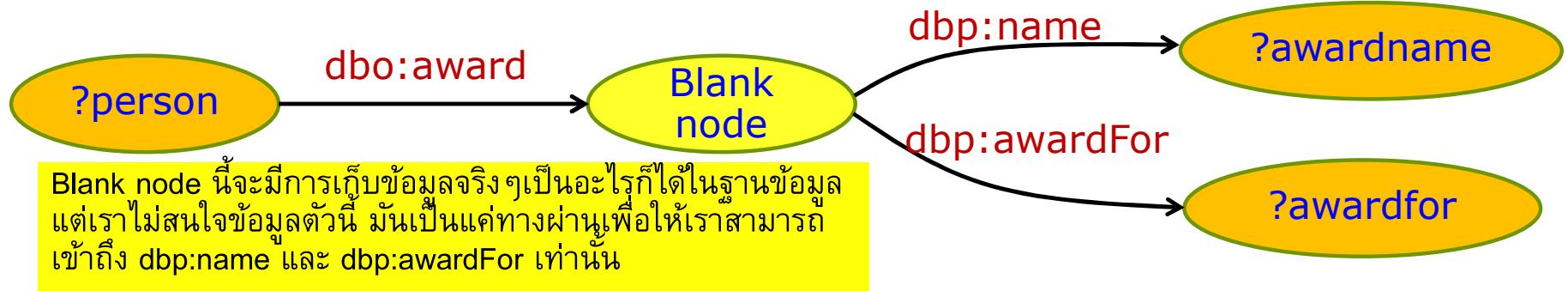
SPARQL Complex Query Pattern

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

```
SELECT ?person ?awardname ?awardfor
WHERE {
  ?person dbo:award [ dbp:name ?awardname ;
    dbp:awardFor ?awardfor ]
}
```

เราสามารถระบุเฉพาะบางตัวแปรที่เราต้องการให้แสดงในผลลัพธ์ และลำดับการแสดงผลของผลลัพธ์ในตัวแปรได้ด้วย

Blank node ถูกใช้เป็น object ของ property “dbo:award” ในขณะเดียวกัน blank node นี้ก็จะถูกใช้เพื่อทำหน้าที่เป็น subject ของ dbp:name และ dbp:awardFor



SPARQL Complex Query Pattern

หรืออาจเขียนอีกแบบก็ได้ โดยการใช้ตัวแปร (เช่น `?x`) แทน blank node และเราไม่ได้อ่านมูลในตัวแปร `?x` มาแสดงผลที่ `SELECT`

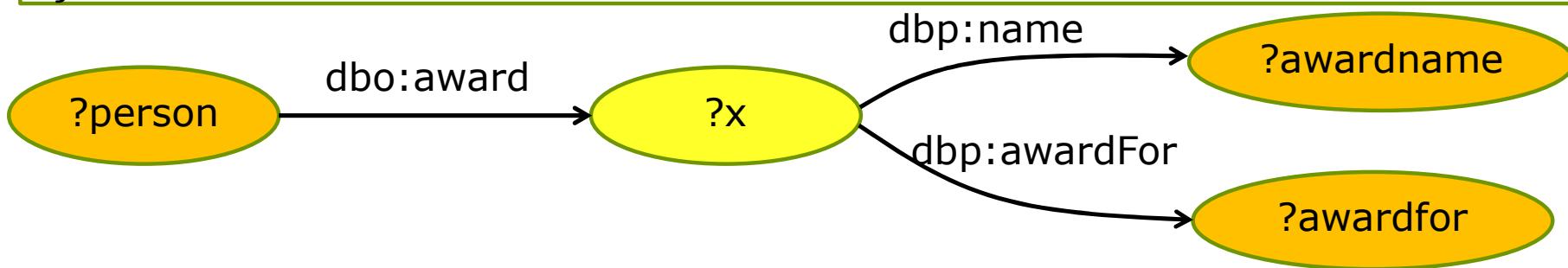
```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbp: <http://dbpedia.org/property/>
```

```
SELECT ?person ?awardname ?awardfor
```

```
WHERE {
```

```
    ?person dbo:award ?x.  
    ?x dbp:name ?awardname ;  
        dbp:awardFor ?awardfor
```

```
}
```





SPARQL Complex Query Pattern

SPARQL Explorer for https://dbpedia.org/sparql

```
SPARQL:  
PREFIX owl: <http://www.w3.org/2002/07/owl#>  
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>  
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>  
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>  
PREFIX foaf: <http://xmlns.com/foaf/0.1/>  
PREFIX dc: <http://purl.org/dc/elements/1.1/>  
PREFIX : <http://dbpedia.org/resource/>  
PREFIX dbpedia2: <http://dbpedia.org/property/>  
PREFIX dbpedia: <http://dbpedia.org/>  
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>  
  
PREFIX dbo: <http://dbpedia.org/ontology/>  
PREFIX dbp: <http://dbpedia.org	property/>  
  
SELECT ?person ?awardname ?awardfor  
WHERE  
{ ?person dbo:award [ dbp:name ?awardname ;  
    dbp:awardFor ?awardfor ] }  
Results: Browse Go! Reset
```

SPARQL results:

person	awardname	awardfor
:Richards_Miller	"Distinguished Eagle Scout Award"@en	"Distinguished service in his profession and to his community for a period of at least twenty-five years after attaining the level of Eagle Scout"@en
:Max_I._Silber	"Distinguished Eagle Scout Award"@en	"Distinguished service in his profession and to his community for a period of at least twenty-five years after attaining the level of Eagle Scout"@en
:William_Hillcourt	"Distinguished Eagle Scout Award"@en	"Distinguished service in his profession and to his community for a period of at least twenty-five years after attaining the level of Eagle Scout"@en
:W_Scott_Sorrels	"Distinguished Eagle Scout Award"@en	"Distinguished service in his profession and to his community for a period of at least twenty-five years after attaining the level of Eagle Scout"@en
:Hector_Perez	"Distinguished Eagle Scout Award"@en	"Distinguished service in his profession and to his community for a period of at least twenty-five years after attaining the level of Eagle Scout"@en
:Ross_Erin_Butler_Sr.	"Silver Antelope Award"@en	"outstanding service to youth within a BSA National Service Territory"@en
:William_Hillcourt	"Bronze Wolf"@en	"Outstanding service to Scouting"@en
:Hayat_Saif	"Bronze Wolf"@en	"Outstanding service to Scouting"@en
:Arthur_W._V._Reeve	"Bronze Wolf"@en	"Outstanding service to Scouting"@en
:Odd_Hopp	"Bronze Wolf"@en	"Outstanding service to Scouting"@en
:Victor_Steiner_Sr.	"Bronze Wolf"@en	"Outstanding service to Scouting"@en
:Savitri_(actress)	"Rashtrapati Award"@en	"achievements in the field of sports, art, military, literature, cinema, culture, science and technology, or Scouting"@en
:Vidyanand_Sarek	"Rashtrapati Award"@en	"achievements in the field of sports, art, military, literature, cinema, culture, science and technology, or Scouting"@en
:Kala_Nath_Shastri	"Rashtrapati Award"@en	"achievements in the field of sports, art, military, literature, cinema, culture, science and technology, or Scouting"@en
:S_V_Ranga_Rao	"Rashtrapati Award"@en	"achievements in the field of sports, art, military, literature, cinema, culture, science and technology, or Scouting"@en

SPARQL Complex Query Pattern

- ▢ ถ้าต้องการค้นหารายชื่อผู้แต่ง (authors) ทุกคน ชื่อหนังสือที่โดดเด่น (notable work) ของผู้แต่งแต่ละคน และจำนวนหน้า (number of page) ของหนังสือแต่ละเล่มด้วย

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>
```

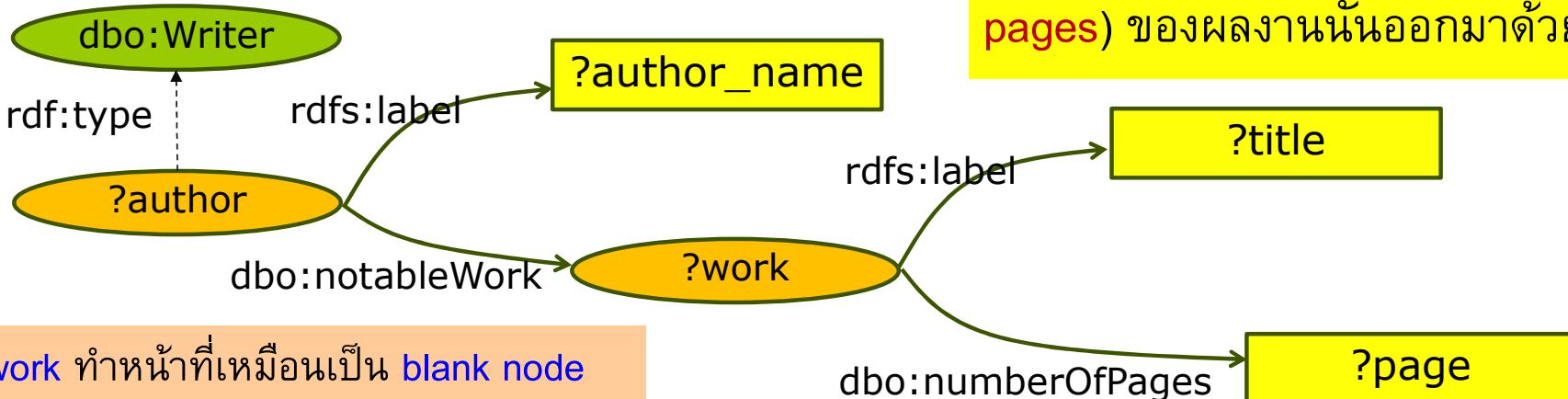
```
SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
}
```

ให้แสดงชื่อผู้แต่ง (author names), ชื่อหนังสือที่โดดเด่นของเขา(notable work) และจำนวนหน้า (number of pages) อีกมา

SPARQL Complex Query Pattern

```

SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
              rdfs:label ?author_name ;
              dbo:notableWork ?work .
    ?work rdfs:label ?title;
           dbo:numberOfPages ?page .
}
  
```



ค้นหาคนที่มี **rdf:type** เป็น **dbo:Writer**, และให้แสดงชื่อ (**author name**) ที่ถูกเก็บอยู่ใน **rdfs:label** ออกแบบ และให้แสดงชื่อผลงานของเขา (**notable works**) ที่ถูกเก็บอยู่ใน **rdfs:label** ของงานนั้นโดยนำมาเก็บในตัวแปร **?title** และให้แสดงจำนวนหน้า (**number of pages**) ของผลงานนั้นออกแบบด้วย



SPARQL Complex Query Pattern

← → ⌂ 🔒 dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dbo: <http://dbpedia.org/ontology/>

SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
}
```

Results:

SPARQL results:

author_name	title	page
"Caleb Carr"@en	"El caso del secretario italiano"@es	256
"Caleb Carr"@en	"The Italian Secretary"@en	256
"کالب کار"	"El caso del secretario italiano"@es	256
"کالب کار"	"The Italian Secretary"@en	256
"Caleb Carr"@de	"El caso del secretario italiano"@es	256
"Caleb Carr"@de	"The Italian Secretary"@en	256
"Caleb Carr"@es	"El caso del secretario italiano"@es	256
"Caleb Carr"@es	"The Italian Secretary"@en	256
"Caleb Carr"@fr	"El caso del secretario italiano"@es	256
"Caleb Carr"@fr	"The Italian Secretary"@en	256
"Caleb Carr"@it	"El caso del secretario italiano"@es	256
"Caleb Carr"@it	"The Italian Secretary"@en	256
"Caleb Carr"@pl	"El caso del secretario italiano"@es	256
"Caleb Carr"@pl	"The Italian Secretary"@en	256
"Caleb Carr"@en	"El alienista (novela)"@es	496

34

SPARQL: Additional Constraints

□ จากตัวอย่างที่แล้วให้ทำการเพิ่มเติมคำสั่งต่อไปนี้

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX dbo: <http://dbpedia.org/ontology/>
```

```
SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title;
        dbo:numberOfPages ?page .
}
```

```
ORDER BY ASC (?author_name)
OFFSET 10
LIMIT 100
```

ถ้าใช้ DESC (descending order) จะเป็นการเรียงจากมากไปน้อย

- ตัวอย่างนี้จะเป็นการสั่งให้แสดงผลลัพธ์โดยเรียงลำดับชื่อผู้แต่ง (**author name**) จากน้อยไปมา (**ASC-ascending order**) และให้ยกเว้น 10 คนแรกไม่ต้องเอามาแสดง (**OFFSET 10**) แต่ให้นำคนที่ 11 เป็นต้นไปมาแสดง โดยนำมาจำนวนทั้งหมด 100 คน (**LIMIT 100**)

SPARQL: FILTER Constraints

- ถ้าต้องการเพิ่มเติมคำสั่งต่อไปนี้อีก

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

```
SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
    FILTER (?page < 1000)
}
ORDER BY DESC (?author_name)
LIMIT 100
```

คำสั่ง **FILTER** นี้จะเป็นการกำหนดเงื่อนไขเพิ่มเติมคือให้เอาเฉพาะผลงานที่มีจำนวนหน้า < 1000 หน้าอุกมาแสดง

SPARQL: FILTER Constraints

← → ⌂ dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0D

SPARQL Explorer for https://dbpedia.org/sparql

```

SPARQL:
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

SELECT ?author_name ?title ?page
WHERE {
  ?author rdf:type dbo:Writer ;
    rdfs:label ?author_name ;
    dbo:notableWork ?work .
  ?work rdfs:label ?title ;
    dbo:numberOfPages ?page .
  FILTER (?page < 1000)
}
  
```

Results:

SPARQL results:

author_name	title	page
"Émile Zola"@in	"Germinal (román)"@cs	591
"Émile Zola"@in	"Germinal (Roman)"@de	591
"Émile Zola"@in	"Ζερμινάλ (μυθιστόρημα)"@el	591
"Émile Zola"@in	"Germinal"@eo	591
"Émile Zola"@in	"Germinal (novela)"@es	591
"Émile Zola"@in	"Germinal (roman)"@fr	591
"Émile Zola"@in	"Germinal (novel)"@en	591
"Émile Zola"@in	"Germinale (romanzo)"@it	591
"Émile Zola"@in	"제르미날"@ko	591
"Émile Zola"@in	"ジエルミナール (小説)"@ja	591
"Émile Zola"@in	"Germinal (roman)"@nl	591
"Émile Zola"@in	"Germinal (powieść)"@pl	591
"Émile Zola"@in	"Germinal"@pt	591
"Émile Zola"@in	"Жерминаль (роман)"@ru	591
"Émile Zola"@in	"Den stora qrvnstreiken"@sv	591

SPARQL: FILTER Constraints

□ ถ้ามีการเพิ่มเติมคำสั่งต่อไปนี้อีก

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

```
SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
              rdfs:label ?author_name ;
              dbo:notableWork ?work .
    ?work rdfs:label ?title ;
           dbo:numberOfPages ?page .
    FILTER (LANG (?author_name)="en") .
    FILTER (LANG (?title)="en") .
    FILTER (?page <500) .
}
LIMIT 100
```

คำสั่งแรกและคำสั่งที่สองจะเป็นการกรองเอาเฉพาะชื่อผู้แต่ง (**author name**) ที่เขียนเป็นภาษาอังกฤษ (“**en**”) และชื่อหนังสือ (**title**) ที่เขียนเป็นภาษาอังกฤษ (“**en**”)
คำสั่งสุดท้ายคือการกรองเอาเฉพาะผลงานที่มีจำนวนหน้า < 500 ออกมาระดับ



SPARQL: FILTER Constraints

← → ⌂ dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0I

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
    FILTER (LANG (?author_name)="en") .
    FILTER (LANG (?title)="en") .
    FILTER (?page <500) .
```

Results:

SPARQL results:

author_name	title	page
"Joseph Heller"@en	"Catch-22"@en	453
"Elizabeth Yates (writer)"@en	"Amos Fortune, Free Man"@en	181
"Fredrik Backman"@en	"Beartown (novel)"@en	418
"David Sheff"@en	"Beautiful Boy: A Father's Journey Through His Son's Addiction"@en	326
"Richard Fariña"@en	"Been Down So Long It Looks Like Up to Me"@en	269
"Reinaldo Arenas"@en	"Before Night Falls"@en	317
"Dean Brelis"@en	"Behind the Burma Road"@en	246
"Ann Patchett"@en	"Bel Canto (novel)"@en	318
"Maaza Mengiste"@en	"Beneath the Lion's Gaze"@en	308
"Joseph P. Lash"@en	"Eleanor: The Years Alone"@en	368
"Christopher Paul Curtis"@en	"Elijah of Buxton"@en	341
"V. Vale"@en	"Modern Primitives (book)"@en	212
"David Wellington (author)"@en	"Monster Island (Wellington novel)"@en	272
"Bibliography of André Gide"@en	"The Immoralist"@en	144
"André Gide"@en	"The Immoralist"@en	144
"Virginia Hamilton"@en	"M. C. Higgins, the Great"@en	279

SPARQL: FILTER Constraints

□ ถ้ามีการเพิ่มเติมคำสั่งต่อไปนี้อีก

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

```
SELECT ?author_name ?title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
    FILTER (LANG (?author_name)="en") .
    FILTER (LANG (?title)="en") .
    FILTER (?page <500) .
    FILTER REGEX (?title, "World", "i") .
}
```

```
LIMIT 100
```

string

Regular
expression

flags

คำสั่งนี้จะเป็นการกรองเอาเฉพาะชื่อผลงาน (**title**) ที่มีคำว่า “**World**” อู๊ด
ภายในชื่อนั้น และกำหนด flags เป็น “**i**”
หมายถึง ignoring คือไม่สนใจตัวใหญ่
หรือตัวเล็ก (case sensitive)



SPARQL: FILTER Constraints

← → ⌂ dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0D%

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
dbo:numberOfPages ?page .
FILTER (LANG (?author_name)="en") .
FILTER (LANG (?title)="en") .
FILTER (?page <500) .
FILTER REGEX (?title, "World", "i").
}
LIMIT 100
```

Results: Browse Go! Reset

SPARQL results:

author_name	title	page
"Susan Cain"@en	"Quiet: The Power of Introverts in a World That Can't Stop Talking"@en	333
"Donald Antrim"@en	"Elect Mr. Robinson for a Better World"@en	192
"Josh Viola"@en	"Cyber World"@en	250
"Christopher Priest (novelist)"@en	"Inverted World"@en	256
"Colum McCann"@en	"Let the Great World Spin"@en	368
"Elif Shafak"@en	"10 Minutes 38 Seconds in This Strange World"@en	311
"Alexander Weinstein (author)"@en	"Children of the New World"@en	240
"Daniel Kehlmann"@en	"Measuring the World"@en	272
"Daniel Kehlmann"@en	"Measuring the World"@en	304
"Jack Weatherford"@en	"Genghis Khan and the Making of the Modern World"@en	312
"Larry Niven"@en	"Ringworld"@en	342
"Stephen Briggs"@en	"The Discworld Companion"@en	288
"Andrei Gusev"@en	"The World According to Novikoff"@en	384
"Jade Chang"@en	"The Waags vs. the World"@en	354
"H. G. Wells"@en	"The War of the Worlds"@en	287

ชื่อผลงานทั้งหมดนี้จะมีคำว่า
“World” ออย่างน้อยในด้วย

SPARQL: FILTER Constraints

□ ถ้ามีการเพิ่มเติมคำสั่งต่อไปนี้อีก

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

```
SELECT ?author_name ?title ?de_title ?page
WHERE {
    ?author rdf:type dbo:Writer ;
        rdfs:label ?author_name ;
        dbo:notableWork ?work .
    ?work rdfs:label ?title ;
        dbo:numberOfPages ?page .
    FILTER (LANG (?author_name)="en") .
    FILTER (LANG (?title)="en") .
    FILTER REGEX (?title, "World", "i").
    OPTIONAL {?work rdfs:label ?de_title
              FILTER (LANG (?de_title) = "de") }
}
LIMIT 100
```



คำสั่ง OPTIONAL นี้จะหมายถึง การกำหนดที่เป็นทางเลือก เพิ่มเติมอีกว่าถ้าผลงานนั้น มีชื่อ ผลงานที่เป็นภาษาเยอรมัน ("de") ด้วย ก็ให้นำมาแสดงด้วย



SPARQL: FILTER Constraints

← → ⌂ dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0A%0D%0ASELECT+%3Fa

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>

dbo:numberOfPages ?page .
FILTER (LANG (?author_name) = "en") .
FILTER (LANG (?title) = "en") .
FILTER REGEX (?title, "World", "i").
OPTIONAL {?work rdfs:label ?de_title
          FILTER (LANG (?de_title) = "de") }
}
LIMIT 100
```

Results: [Browse](#) [Go!](#) [Reset](#)

บางผลงานมีแต่ชื่อผลงานเป็นภาษาอังกฤษ (@en)อย่างเดียว แต่บางผลงานก็จะมีทั้งชื่อที่เป็นภาษาอังกฤษ (@en) และภาษาเยอรมัน (@de)

บางผลงานมีแต่ชื่อผลงานเป็นภาษาอังกฤษ (@en)อย่างเดียว แต่บางผลงานก็จะมีทั้งชื่อที่เป็นภาษาอังกฤษ (@en) และภาษาเยอรมัน (@de)

SPARQL results:

author_name	title	de_title	page
"Susan Cain"@en	"Quiet: The Power of Introverts in a World That Can't Stop Talking"@en	-	333
"Donald Antrim"@en	"Elect Mr. Robinson for a Better World"@en	"Wählt Mr. Robinson für eine bessere Welt"@de	192
"Josh Viola"@en	"Cyber World"@en	-	250
"Christopher Priest (novelist)"@en	"Inverted World"@en	-	256
"Colum McCann"@en	"Let the Great World Spin"@en	-	368
"Elif Shafak"@en	"10 Minutes 38 Seconds in This Strange World"@en	-	311
"Charles C. Mann"@en	"1493: Uncovering the New World Columbus Created"@en	-	560
"Daniel Kehlmann"@en	"Measuring the World"@en	"Die Vermessung der Welt"@de	272
"Daniel Kehlmann"@en	"Measuring the World"@en	"Die Vermessung der Welt"@de	304
"Alexander Weinstein (author)"@en	"Children of the New World"@en	-	240
"Jack Weatherford"@en	"Genghis Khan and the Making of the Modern World"@en	-	312
"Larry Niven"@en	"Ringworld"@en	-	342
"Stephen Briggs"@en	"The Discworld Companion"@en	-	288
"H. G. Wells"@en	"The War of the Worlds"@en	"Der Krieg der Welten"@de	287
"Jade Chang"@en	"The Wangs vs. the World"@en	-	354

SPARQL –Example of function ‘count’

- ตัวอย่างที่จะมีการนับ (count) จำนวนประเทศ ทั้งหมดที่มีอยู่ใน DBpedia

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

```
SELECT count (?country) AS ?totalcountry
```

```
WHERE {
```

```
    ?c a dbo:Country ;
        rdfs:label ?country .
    FILTER (LANG (?country) = "en") .
```

```
}
```

Count จำนวนประเทศทั้งหมดที่มีอยู่
ในฐานข้อมูล DBpedia ผลลัพธ์ที่ได้
ให้เก็บไว้ในตัวแปร ?totalcountry

อันนี้มีความหมายเหมือนกับ
?c rdf:type dbo:Country



SPARQL – Example of function ‘count’

dbpedia.org/snorql/?query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.org%2Fontology%2F>%0D%0APREFIX+dbr%3A+<http%3A%2F%2Fdbpedia.org%2Fresource%2F>

SPARQL Explorer for https://dbpedia.org/sparql

SPARQL:

```
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX : <http://dbpedia.org/resource/>
PREFIX dbpedia2: <http://dbpedia.org/property/>
PREFIX dbpedia: <http://dbpedia.org/>
PREFIX skos: <http://www.w3.org/2004/02/skos/core#>
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX dbr: <http://dbpedia.org/resource/>

SELECT count (?country) AS ?totalcountry
WHERE {
  ?c a dbo:Country ;
    rdfs:label ?country .
  FILTER (LANG (?country)="en") .
```

Results:

SPARQL results:

totalcountry
14235

SPARQL – Using “FROM”

□ การใช้คำสั่ง “FROM” ใน SPARQL

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

```
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
```

```
SELECT ?nick ?name
```

```
FROM <http://www.dajobe.org/foaf.rdf>
```

```
WHERE {
```

```
    ?x a foaf:Person ;  
        foaf:nick ?nick ;  
        foaf:name ?name
```

```
}
```

Target graph URI ซึ่งเราสามารถ view ดูโครงสร้างข้อมูลได้บน Web browser โดยตรง

SPARQL ยังยอมให้เราสามารถใช้ FROM ได้เช่นกัน ในกรณีที่เราต้องการสืบค้นข้อมูลของกราฟ URI จากที่อื่น เราสามารถระบุชื่อกราฟ URI นั้นได้โดยตรงในคำสั่ง FROM ของ SPARQL ที่เราใช้บนหน้า interface ของ SPARQL endpoint นั้นได้เลย อย่างไรก็ตาม ไม่ใช่ว่าทุกๆ SPARQL endpoint จะอนุญาตให้เราใช้ FROM เช่นนี้ได้ ตัวอย่างเช่น Dbpedia endpoint จะไม่อนุญาตให้เราระบุการใช้กราฟ URI จากที่อื่นได้ แต่ก็มีบาง endpoint เท่านั้นที่อนุญาตให้ทำเช่นนี้ได้ ดังตัวอย่างต่อไปนี้



SPARQL – Using SPARQLer processor



Validators

- [SPARQL query validator](#)
- [SPARQL update validator](#)
- [RDF data validator](#)
- [IRI validator](#)

Query

- [General purpose SPARQL processor](#)

Click this link



SPARQL Services

The general SPARQL query service is accessed directly using the SPARQL protocol at URL /sparql

Links to SPARQL 1.1 Specifications

The full set of SPARQL specification is:

- [SPARQL Query language](#)
- [SPARQL Update](#)
- [SPARQL Protocol](#)
- [SPARQL Graph Store Protocol](#)
- SPARQL Result formats
 - [SPARQL Query Results JSON Format](#)
 - [SPARQL Query Results CSV and TSV Formats](#)
 - [SPARQL Query Results XML Format](#)
- [SPARQL Service Description](#)
- [SPARQL Federated Query](#)

SPARQL – Using SPARQLer processor



The screenshot shows a web browser window with the URL <https://sparql.org/sparql.html>. The page title is "SPARQLer - General purpose processor". The main content area contains a SPARQL query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?nick ?name
FROM <http://www.dajobe.org/foaf.rdf>
WHERE {
  ?x a foaf:Person ;
    foaf:nick ?nick ;
    foaf:name ?name
}
```

A red arrow points from the explanatory text in the yellow callout box to the "FROM" clause of the query. A green arrow points from the URL bar to the browser address.

**ถ้าเรามีการระบุ Graph URI
อยู่ในคำสั่ง FROM นี้แล้ว ก็
ไม่จำเป็นต้องไประบุ graph
URI ที่ช่องด้านล่างนี้อีก**

Target graph URI (or use FROM in the query)

If no dataset is provided, the query will execute against an empty one.

The query can contain use VALUES to set some variables.

Output:

XSLT style sheet (blank for none):

Force the accept header to text/plain regardless

SPARQL – Using SPARQLer processor


sparql.org/sparql.html

SPARQLer - General purpose processor

General SPARQL query : input query, set any options and press "Get Results"

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?nick ?name
WHERE {
  ?x a foaf:Person ;
      foaf:nick ?nick ;
      foaf:name ?name
}
```

แต่ถ้าเราไม่ได้มีการระบุ
Graph URI อยู่ในคำสั่ง

FROM เราจึงสามารถระบุ
graph URI ในช่องข้างล่างนี้
ได้เช่นกัน

Target graph URI (or use FROM in the query)

If no dataset is provided, the query will execute against an empty one.

The query can contain use VALUES to set some variables.

Output: ←

XSLT style sheet (blank for none):

Force the accept header to text/plain regardless

นอกจากนี้เรายังสามารถเลือกรูปแบบของผลลัพธ์
ที่ต้องการให้นำมาแสดงได้อีกด้วยหลังจากนั้นก็ให้
คลิก Get Results

SPARQL – Using SPARQLer processor

← → C 🔒 sparql.org/sparql?query=PREFIX+rdf%3A+<http%3A%2F%2Fwww.w3.org%2F1999%2F02%2F22-rdf-syntax-ns%23>%0D%0APREFIX...

SPARQLer Query Results

nick	name
"zool"	"Jo Walsh"
"sbp"	"Sean B. Palmer"
"DanC"	"Dan Connolly"
"libby"	"Libby Miller"
"shellac"	"Damian Steer"
"em"	"Eric Miller"
"emiller"	"Eric Miller"
"danbri"	"Dan Brickley"
"mortenf"	"Morten Frederiksen"
"jang"	"Jan Grant"
"ndw"	"Norm Walsh"



SPARQL – Using SPARQLer processor

https://librdf.org/query/

Redland Rasqal RDF Query Demonstration

by [Dave Beckett](#)

This is a demonstration of using [Rasqal](#) to perform queries in either of the SPARQL languages over RDF data. The data is loaded into a [Redland](#) model and then queried and results accessed via the [Redland Perl](#) language binding.

The query must be in SPARQL and needs a URI of some RDF content to query and run it. The example query is for FOAF and finds the names of all the people in the graph. There are other example queries further down the page.

Some RDF content you can use could be my FOAF file at: <http://www.dajobe.org/foaf.rdf> or the W3C's RSS 1.0 file at: <http://www.w3.org/2000/08/w3c-synd/home.rss>

RDF content URIs

Query

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT DISTINCT ?name
WHERE {
?x rdf:type foaf:Person .
?x foaf:name ?name
}
ORDER BY ?name
```

ถ้าต้องการดูว่ามี Graph URI ที่ไหน
ที่ให้บริการสืบค้นข้อมูลบ้าง เรา
สามารถไปที่ endpoint นี้เพื่อดูได้

in SPARQL with an XML result JSON result



SPARQL – Using SPARQLer processor

← → ⌂ librdf.org/query/

[Run this query](#)

3. Print the description of a project and maintainer(s) using DOAP

Data: <http://librdf.org/raptor/raptor.rdf>

Query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX doap: <http://usefulinc.com/ns/doap#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>

SELECT $description $maintainerName
WHERE {
    $project rdf:type doap:Project .
    $project doap:description $description .
    $project doap:maintainer $m .
    $m foaf:name $maintainerName
}
```

[Run this query](#)

4. Print the names and optional nicks of people in my FOAF file where

Data: <http://www.dajobe.org/foaf.rdf>

Query:

```
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX foaf: <http://xmlns.com/foaf/0.1/>
SELECT ?name ?nick
WHERE {
    ?x rdf:type foaf:Person .
    ?x foaf:name ?name .
    OPTIONAL { ?x foaf:nick ?nick }
}
```

เมื่อ scroll ลงมาด้านล่างของเว็บ ก็จะเห็นรายชื่อตัวอย่างของ graph URI ที่ให้บริการสืบค้นข้อมูล เราจึงสามารถ copy Graph URI นี้และนำไปใช้สืบค้นกับ SPARQLer ได้ด้วย

SPARQL –Selecting Display Result

□ Consider the following SPARQL

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT *
WHERE {
    ?author rdf:type dbo:Writer ;
              dbo:notableWork ?work
}
LIMIT 10
```

คำสั่ง SPARQL ที่เราพบมากที่สุดคือการใช้คำสั่ง **SELECT** ซึ่งอาจตามด้วย ***** หรือจะตามด้วยชื่อตัวแปรที่เรากำหนดก็ได้ เช่น (**?author ?work**)

SPARQL –Using “DESCRIBE”

- นอกจากคำสั่ง SELECT แล้วเรายังสามารถใช้คำสั่งต่อไปนี้ได้ด้วย

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
DESCRIBE *
WHERE {
    ?author rdf:type dbo:Writer;
              dbo:notableWork ?work
}
LIMIT 10
```



SPARQL – Using “DESCRIBE”

SPARQL Query Editor About Tables ▾ Conductor Facet Browser Permalink

Default Data Set Name (Graph) http://dbpedia.org

Query Text

```
PREFIX dbo: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
DESCRIBE *
WHERE {
    ?author rdf:type dbo:Person .
    ?author dbo:notableWork ?work .
}
LIMIT 10
```

Results Format HTML (table)

Execute Query Reset

Execution timeout 30000 milliseconds

HTML (table)

Turtle
Turtle (beautified)
RDF/JSON
RDF/XML
N-Triples
XHTML+RDFa
ATOM+XML
ODATA/JSON
JSON-LD (plain)
JSON-LD (with context)
HTML (list)
HTML (table)
HTML+Microdata (basic)
HTML+Microdata (table)
HTML+JSON-LD (basic)
HTML+Turtle (basic)
Turtle (beautified - browsing oriented)
Microdata/JSON
CSV
TSV

การใช้ DESCRIBE แทน SELECT จะทำให้เราสามารถเลือกรูปแบบการแสดงผลได้มากกว่าการใช้ SELECT



SPARQL – Using “DESCRIBE”

SPARQL HTML5 RDFa and Microdata document		
Subject	Predicate	Object
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Person100007846
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/PhysicalEntity100001930
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/ontology/Animal
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://www.w3.org/2002/07/owl#Thing
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/WikicatTurkishPoets
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/ontology/Species
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/WikicatTurkishWriters
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/ontology/Eukaryote
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/WikicatTurkishTranslator
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Whole100003553
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Communicator10961066
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://xmlns.com/foaf/0.1/Person
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Poet110444194
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/ontology/Person
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Linguist110264219
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Writer110794014
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Object100002684
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/YagoLegalActorGeo
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/LivingThing100004258
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/WikicatPeopleFromAnka
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Translator110725280
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://schema.org/Person
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/Organism100004475
http://dbpedia.org/resource/Cahit_Zarifoğlu ⓘ	http://www.w3.org/1999/02/22-rdf-syntax-ns#type ⓘ	http://dbpedia.org/class/yago/CausalAgent100007347

การใช้ DESCRIBE นอกจาก
การแสดงผลลัพธ์ที่เก็บอยู่ใน
ตัวแปรของ Subject หรือ
Object แล้ว ก็จะมีการแสดง
properties ทุกตัวของผลลัพธ์
(author) ออกมาด้วยทั้งหมด



SPARQL – Using “DESCRIBE”

SPARQL Query Editor About Tables ▾ Conductor Facet Browser Permalink

Default Data Set Name (Graph) http://dbpedia.org

Query Text

```
PREFIX dbo: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
DESCRIBE *
WHERE {
    ?author rdf:type dbo:Person .
    ?author dbo:notableWork ?work .
}
LIMIT 10
```

Results Format Turtle

Turtle

- Turtle (beautified)
- RDF/JSON
- RDF/XML
- N-Triples
- XHTML+RDFa
- ATOM+XML
- ODATA/JSON
- JSON-LD (plain)
- JSON-LD (with context)
- HTML (list)
- HTML (table)
- HTML+Microdata (basic)
- HTML+Microdata (table)
- HTML+JSON-LD (basic)
- HTML+Turtle (basic)
- Turtle (beautified - browsing oriented)
- Microdata/JSON
- CSV
- TSV

Execute Query Reset

ทดสอบเลือกรูปแบบการ
แสดงผลเป็นแบบ
“Turtle” ดู



SPARQL –Using “DESCRIBE”

```
← → ⌂ dbpedia.org/sparql?default-graph-uri=http%3A%2F%2Fdbpedia.org&query=PREFIX+dbo%3A+<http%3A%2F%2Fdbpedia.o

@prefix rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#> .
@prefix yago: <http://dbpedia.org/class/yago/> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type yago:Person100007846 ,
    yago:PhysicalEntity100001930 .
@prefix dbo: <http://dbpedia.org/ontology/> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type dbo:Animal .
@prefix owl: <http://www.w3.org/2002/07/owl#> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type owl:Thing ,
    yago:WikicatTurkishPoets ,
    dbo:Species ,
    yago:WikicatTurkishWriters ,
    dbo:Eukaryote ,
    yago:WikicatTurkishTranslators ,
    yago:Whole100003553 ,
    yago:Communicator109610660 .
@prefix foaf: <http://xmlns.com/foaf/0.1/> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type foaf:Person ,
    yago:Poet110444194 ,
    dbo:Person ,
    yago:Linguist110264219 ,
    yago:Writer110794014 ,
    yago:Object100002684 ,
    yago:YagoLegalActor ,
    yago:YagoLegalActorGeo ,
    yago:LivingThing100004258 ,
    yago:WikicatPeopleFromAnkara ,
    yago:Translator110725280 .
@prefix schema: <http://schema.org/> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type schema:Person ,
    yago:Organism100004475 ,
    yago:CausalAgent100007347 .
@prefix ns6: <http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type ns6:NaturalPerson ,
    dbo:Writer .
@prefix wikidata: <http://www.wikidata.org/entity/> .
<http://dbpedia.org/resource/Cahit_Zarifo\u011Flu> rdf:type wikidata:Q5 ,
    wikidata:Q729 ,
    wikidata:Q215627 ,
```

ผลลัพธ์ที่ได้จะถูกแสดงอยู่
ในรูปแบบของ “turtle”

SPARQL – Using “CONSTRUCT”

- นอกจากนี้ก็ยังมีอีกคำสั่งคือ “CONSTRUCT” ที่ใช้แทน “SELECT” ได้ด้วย

```
PREFIX dbo: <http://dbpedia.org/ontology/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
CONSTRUCT { ?author <http://example.org/hasWrite> ?work . }
WHERE {
    ?author rdf:type dbo:Writer;
              dbo:notableWork ?work
}
```

LIMIT 10

การใช้ **CONSTRUCT** จะทำให้เราสามารถสร้างรูปแบบของผลลัพธ์ที่ประกอบไปด้วย **Subject**, **Property**, **Object** แต่สำหรับ **Property** นั้นเรายังสามารถกำหนดว่า เราต้องการใช้คำว่าอะไรเป็น **property** ก็ได้ เช่นจากตัวอย่างจะกำหนดให้ใช้คำว่า “<http://example.org/hasWrite>” เป็น **property** ที่คุณระหว่าง **Subject** (?author) และ **Object** (?work)



SPARQL – Using “CONSTRUCT”

SPARQL Query Editor About Tables ▾ Conductor Facet Browser Permalink

Default Data Set Name (Graph): http://dbpedia.org

Query Text

```
PREFIX dbo: <http://dbpedia.org/resource/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
CONSTRUCT { ?author <http://dbpedia.org/resource/John_Henry_Wright> }
WHERE {
    ?author rdf:type dbo:Person
    dbo:notableWork ?work .
}
LIMIT 10
```

Results Format: HTML (table)

Execute Query Reset

HTML (table)

การใช้ CONSTRUCT จะคล้ายกับ DESCRIBE คือจะเปิดโอกาสให้เราสามารถเลือกรูปแบบการแสดงผลได้มากกว่าการใช้ SELECT.



SPARQL – Using “CONSTRUCT”

SPARQL | HTML5 RDFa and Microdata document

Subject

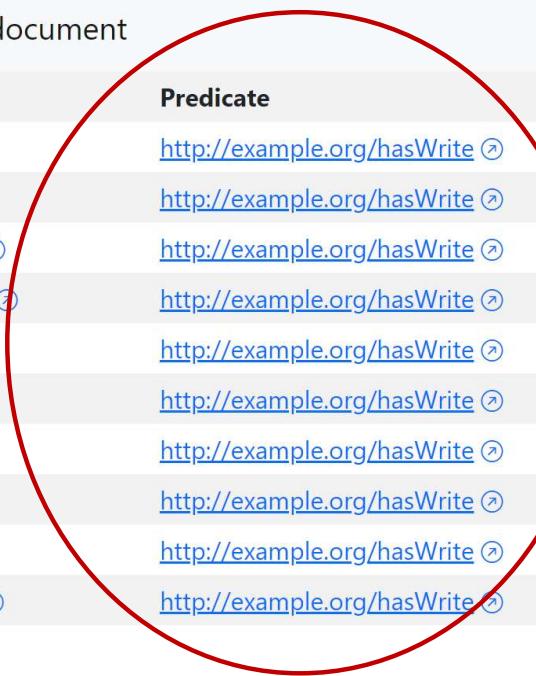
http://dbpedia.org/resource/Cahit_Zarifo\u011f\u011f
http://dbpedia.org/resource/Caitlin_Rother
http://dbpedia.org/resource/Caitl\u00edn_R._Kiernan
http://dbpedia.org/resource/Calder_Willingham
http://dbpedia.org/resource/Caleb_Carr
http://dbpedia.org/resource/Caleb_Carr
http://dbpedia.org/resource/Caleb_Carr
http://dbpedia.org/resource/Caleb_Carr
http://dbpedia.org/resource/Caleb_Carr
http://dbpedia.org/resource/Camilo_Jos\u00e9_Cela

Predicate

<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>
<http://example.org/hasWrite>

Object

<http://dbpedia.org/resource/Y\u011f\u011famak>
[http://dbpedia.org/resource/Lost_Girls_\(book\)](http://dbpedia.org/resource/Lost_Girls_(book))
http://dbpedia.org/resource/The_Drowning_Girl
http://dbpedia.org/resource/The_Graduate
http://dbpedia.org/resource/The_Angel_of_Darkness
http://dbpedia.org/resource/Casing_the_Promised_Land
[http://dbpedia.org/resource/Killing_Time_\(Caleb_Carr_novel\)](http://dbpedia.org/resource/Killing_Time_(Caleb_Carr_novel))
http://dbpedia.org/resource/The_Alienist
http://dbpedia.org/resource/The_Italian_Secretary
[http://dbpedia.org/resource/The_Hive_\(Cela_novel\)](http://dbpedia.org/resource/The_Hive_(Cela_novel))



จะเห็นว่าผลลัพธ์ที่ได้จะมีคำว่า “<http://example.org/hasWrite>” ถูกใช้เป็นค่าข้อมูลของ **Predicate (Property)** เพื่อใช้แบ่งแยกระหว่าง **Subject** และ **Object** ในผลลัพธ์ทุกๆตัวเลย



Reference

- **SPARQL 1.1 Query Language**
(<https://www.w3.org/TR/sparql11-query/>)
- **DBpedia endpoint**
(<https://dbpedia.org/sparql>) and (<https://dbpedia.org/snorql/>)
- **Semantic Web Development Tools**
(<https://www.w3.org/2001/sw/wiki/Tools>)
- **Apache Jena**
(<https://jena.apache.org/index.html>)
- **Redland RDF**
(<https://librdf.org/>)
- **Redland RDF Query Example**
(<https://librdf.org/query/>)
- **SPARQLer - General purpose processor**
(<http://sparql.org/sparql.html>)
- **Semantic Reasoner**
(https://en.wikipedia.org/wiki/Semantic_reasoner)