18B – INTRODUCTION TO SPARQL

CS 1656

Introduction to Data Science

Alexandros Labrinidis – http://labrinidis.cs.pitt.edu
University of Pittsburgh

SPARQL (CONTINUED)

Examples are taken from http://www.learningsparql.com

RDF DATA: ex012.ttl

```
# filename: ex012.ttl

@prefix ab: <http://learningsparql.com/ns/addressbook#> .
@prefix d:  <http://learningsparql.com/ns/data#> .

d:i0432 ab:firstName "Richard" .
d:i0432 ab:lastName  "Mutt" .
d:i0432 ab:homeTel   "(229) 276-5135" .
d:i0432 ab:email     "richard49@hotmail.com" .

d:i9771 ab:firstName "Cindy" .
d:i9771 ab:lastName  "Marshall" .
d:i9771 ab:homeTel   "(245) 646-5488" .
d:i9771 ab:email     "cindym@gmail.com" .

d:i8301 ab:firstName "Craig" .
d:i8301 ab:lastName  "Ellis" .
d:i8301 ab:email     "craigellis@yahoo.com" .
d:i8301 ab:email     "c.ellis@usairwaysgroup.com" .
```

Graph representation?

SPARQL by Example – ex008.rq

```
# filename: ex008.rq

PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?person
WHERE
{ ?person ab:homeTel "(229) 276-5135" . }
```

RESULTS:

```
-----------------------------------------
| person                                |
=========================================
| http://learningsparql.com/ns/data#d:i0432 |
-----------------------------------------
```

SPARQL by Example – ex017.rq

```
# filename: ex017.rq

PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last
WHERE
{
  ?person ab:homeTel "(229) 276-5135" .
  ?person ab:firstName ?first .
  ?person ab:lastName  ?last .
}
```

RESULTS:

```
----------------------
| first      | last    |
======================
| "Richard"  | "Mutt"  |
----------------------
```

SPARQL by Example – ex047.rq

```
# filename: ex047.rq

PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last
WHERE
{
  ?person ab:homeTel "(229) 276-5135" ;
          ab:firstName ?first ;
          ab:lastName  ?last .
}
```

RESULTS:

first	last
"Richard"	"Mutt"

SPARQL by Example – ex015.rq

```
# filename: ex015.rq

PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?craigEmail
WHERE
{
  ?person ab:firstName "Craig" .
  ?person ab:lastName  "Ellis" .
  ?person ab:email ?craigEmail .
}
```

craigEmail
“c.ellis@usairwaysgroup.com”
“craigellis@yahoo.com”

SPARQL by Example – ex023.rq

```
# filename: ex023.rq

PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?craigEmail ?homeTel
WHERE
{
  ?person ab:firstName "Craig" .
  ?person ab:lastName  "Ellis" .
  ?person ab:email ?craigEmail .
  ?person ab:homeTel ?homeTel .
}
```

craigEmail	homeTel

Why empty?

RDF Data: ex054.ttl

```
# filename: ex054.ttl
@prefix ab: <http://learningsparql.com/ns/addressbook#> .
@prefix d:  <http://learningsparql.com/ns/data#> .

d:i0432 ab:firstName "Richard" .
d:i0432 ab:lastName  "Mutt" .
d:i0432 ab:homeTel   "(229) 276-5135" .
d:i0432 ab:nick      "Dick" .
d:i0432 ab:email     "richard49@hotmail.com" .

d:i9771 ab:firstName "Cindy" .
d:i9771 ab:lastName  "Marshall" .
d:i9771 ab:homeTel   "(245) 646-5488" .
d:i9771 ab:email     "cindym@gmail.com" .

d:i8301 ab:firstName "Craig" .
d:i8301 ab:lastName  "Ellis" .
d:i8301 ab:workTel   "(245) 315-5486" .
d:i8301 ab:email     "craigellis@yahoo.com" .
d:i8301 ab:email     "c.ellis@usairwaysgroup.com" .
```

SPARQL by Example – ex055.rq

```
# filename: ex055.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last ?workTel
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName ?last ;
     ab:workTel ?workTel .
}
```

RESULTS:

```
-------------------------------------------
| first    | last     | workTel            |
===========================================
| "Craig" | "Ellis" | "(245) 315-5486" |
-------------------------------------------
```

SPARQL by Example – ex057.rq

```
# filename: ex057.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last ?workTel
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName ?last .
 OPTIONAL
  { ?s ab:workTel ?workTel . }
}
```

RESULTS

```
-------------------------------------------
| first     | last       | workTel          |
===========================================
| "Craig"   | "Ellis"    | "(245) 315-5486" |
| "Cindy"   | "Marshall" |                  |
| "Richard" | "Mutt"     |                  |
-------------------------------------------
```

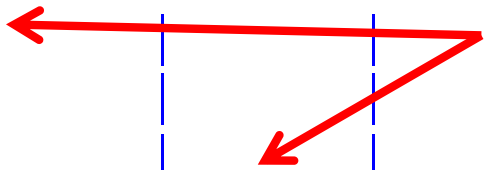
SPARQL by Example – ex059.rq

```
# filename: ex059.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last ?workTel ?nick
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName ?last .
 OPTIONAL {
  ?s ab:workTel ?workTel ;
     ab:nick ?nick .
 } }
```

RESULTS

first	last	workTel	nick
"Craig"	"Ellis"		
"Cindy"	"Marshall"		
"Richard"	"Mutt"		



SPARQL by Example – ex061.rq

```
# filename: ex061.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last ?workTel ?nick
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName ?last .
  OPTIONAL { ?s ab:workTel ?workTel . }
  OPTIONAL { ?s ab:nick ?nick .  }
}
```

RESULTS

first	last	workTel	nick
"Craig"	"Ellis"	"(245) 315-5486"	
"Cindy"	"Marshall"		
"Richard"	"Mutt"		"Dick"

SPARQL by Example – ex063.rq

```
# filename: ex063.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first ?last
WHERE
{
  ?s ab:lastName ?last .
  OPTIONAL { ?s ab:nick ?first . }
  OPTIONAL { ?s ab:firstName ?first . }
}
```

Order of OPTIONAL Statements matters!

RESULTS

```
-------------------------
| first    | last       |
=========================
| "Craig"  | "Ellis"    |
| "Cindy"  | "Marshall" |
| "Dick"   | "Mutt"     |
-------------------------
```

SPARQL QUERIES OVER MULTIPLE "TABLES"

All are still in a single RDF file, but if we were to store the info using the relational model, we would need multiple tables.

RDF Data: ex069.ttl – Page #1

```
# filename: ex069.ttl
@prefix ab: <http://learningsparql.com/ns/addressbook#> .
@prefix d:  <http://learningsparql.com/ns/data#> .

# People
d:i0432 ab:firstName "Richard" ;
        ab:lastName  "Mutt" ;
        ab:email     "richard49@hotmail.com" .

d:i9771 ab:firstName "Cindy" ;
        ab:lastName  "Marshall" ;
        ab:email     "cindym@gmail.com" .

d:i8301 ab:firstName "Craig" ;
        ab:lastName  "Ellis" ;
        ab:email     "c.ellis@usairwaysgroup.com" .
```

RDF Data: ex069.ttl – Page #2

```
# Courses
d:course34 ab:courseTitle "Modeling Data with OWL" .
d:course71 ab:courseTitle "Enhancing Websites with RDFa" .
d:course59 ab:courseTitle "Using SPARQL with non-RDF Data" .
d:course85 ab:courseTitle "Updating Data with SPARQL" .

# Who's taking which courses
d:i8301 ab:takingCourse d:course59 .
d:i9771 ab:takingCourse d:course34 .
d:i0432 ab:takingCourse d:course85 .
d:i0432 ab:takingCourse d:course59 .
d:i9771 ab:takingCourse d:course59 .
```

SPARQL by Example – ex070.rq

```
# filename: ex070.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?last ?first ?courseName
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName  ?last ;
     ab:takingCourse ?course .

  ?course ab:courseTitle ?courseName .
}
```

SPARQL by Example – ex070.rq

RESULTS

last	first	courseName
"Ellis"	"Craig"	"Using SPARQL with non-RDF Data"
"Marshall"	"Cindy"	"Using SPARQL with non-RDF Data"
"Marshall"	"Cindy"	"Modeling Data with OWL"
"Mutt"	"Richard"	"Using SPARQL with non-RDF Data"
"Mutt"	"Richard"	"Updating Data with SPARQL"

SPARQL by Example – pairs

Q: How should we change the following query to return pairs of people (first names) who are taking the same course?

```
# filename: ex070.rq
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?last ?first ?courseName
WHERE {
  ?s ab:firstName ?first ;
     ab:lastName  ?last ;
     ab:takingCourse ?course .

  ?course ab:courseTitle ?courseName .
}
```

SPARQL by Example – pairs

```
PREFIX ab: <http://learningsparql.com/ns/addressbook#>

SELECT ?first1 ?first2 ?courseName
WHERE {
  ?s1 ab:firstName    ?first1 ;
      ab:takingCourse ?course .
  ?s2 ab:firstName    ?first2 ;
      ab:takingCourse ?course .
  ?course ab:courseTitle ?courseName .
}
```

SPARQL PATH QUERIES

RDF Data: ex074.ttl

```
# filename: ex074.ttl
@prefix dc: <http://purl.org/dc/elements/1.1/> .
@prefix rdfs: <http://www.w3.org/2000/01/rdf-schema#> .
@prefix c: <http://learningsparql.com/ns/citations#> .
@prefix : <http://learningsparql.com/ns/papers#> .

:paperA   dc:title     "Paper A" .
:paperB   rdfs:label   "Paper B" ;
          c:cites      :paperA .
:paperC   c:cites      :paperA .
:paperD   c:cites      :paperA ,
                       :paperB .
:paperE   c:cites      :paperA .
:paperF   c:cites      :paperC , :paperE .
:paperG   c:cites      :paperC , :paperE .
:paperH   c:cites      :paperD .
:paperI   c:cites      :paperF , :paperG .
```

SPARQL by Example – ex075.eq

```
# filename: ex075.rq

PREFIX dc: <http://purl.org/dc/elements/1.1/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX : <http://learningsparql.com/ns/papers#>

SELECT ?s ?title
WHERE { ?s (dc:title | rdfs:label) ?title . }
```

RESULTS

```
-----------------------
| s        | title    |
=======================
| :paperB  | "Paper B" |
| :paperA  | "Paper A" |
-----------------------
```

ex077.eq – which papers cited paperA

```
# filename: ex077.rq

PREFIX : <http://learningsparql.com/ns/papers#>
PREFIX c: <http://learningsparql.com/ns/citations#>

SELECT ?s
WHERE { ?s c:cites :paperA . }
```

RESULTS

```
-----------
| s       |
===========
| :paperB |
| :paperC |
| :paperD |
| :paperE |
-----------
```

SPARQL by Example – ex078.rq

```
# filename: ex078.rq

PREFIX : <http://learningsparql.com/ns/papers#>
PREFIX c: <http://learningsparql.com/ns/citations#>

SELECT ?s
WHERE { ?s c:cites+ :paperA . }
```

Adding a + sign looks for papers that cite paperA,
and papers that cite those, and papers that cite those,
etc, until it runs out of papers.

RESULTS

s
:paperE
:paperG
:paperI
:paperF
:paperI
:paperD
:paperH
:paperC
:paperG
:paperI
:paperF
:paperI
:paperB
:paperD
:paperH

SPARQL by Example – ex080.rq

```
# filename: ex080.rq

PREFIX : <http://learningsparql.com/ns/papers#>
PREFIX c: <http://learningsparql.com/ns/citations#>

SELECT ?s
WHERE { ?s c:cites{3} :paperA . }
```

Exactly three links away.

RESULTS

s
:paperI
:paperI
:paperI
:paperI
:paperH

SPARQL by Example – ex082.rq

```
# filename: ex082.rq

PREFIX : <http://learningsparql.com/ns/papers#>
PREFIX c: <http://learningsparql.com/ns/citations#>

SELECT ?s
WHERE { ?s c:cites/c:cites/c:cites :paperA . }
```

Exactly three links away, using path expression

RESULTS

s
:paperI
:paperI
:paperI
:paperI
:paperH

SPARQL by Example – ex083.rq

```
# filename: ex083.rq

PREFIX : <http://learningsparql.com/ns/papers#>
PREFIX c: <http://learningsparql.com/ns/citations#>

SELECT ?s
WHERE { :paperA ^c:cites ?s }
```

Inverse property path:
shows which papers cite paperA
(same as ex077.rq)

FANCY SPARQL QUERIES

Distinct, Union, Filter, Limit

SPARQL by Example – ex105.rq

```
# filename: ex105.rq

PREFIX dm: <http://learningsparql.com/ns/demo#>

SELECT ?s ?cost
WHERE
{
  ?s dm:cost ?cost .
  FILTER (?cost < 10)
}
```

RESULTS

s	cost
<http://learningsparql.com/ns/data#item126>	5
<http://learningsparql.com/ns/data#item432>	8

SPARQL by Example – ex109.rq

```
# filename: ex109.rq
PREFIX dm:  <http://learningsparql.com/ns/demo#>
PREFIX db: <http://dbpedia.org/resource/>

SELECT ?s ?cost ?location
WHERE
{
  ?s dm:location ?location ;
     dm:cost ?cost .
  FILTER (?location IN (db:Montreal, db:Lisbon)) .
}
```

RESULTS

s	cost	location
<http://learningsparql.com/ns/data#item857>	12	db:Montreal
<http://learningsparql.com/ns/data#item126>	5	db:Lisbon

SPARQL by Example – ex111.rq

```
# filename: ex111.rq

PREFIX dm: <http://learningsparql.com/ns/demo#>
PREFIX db: <http://dbpedia.org/resource/>

SELECT ?s ?cost ?location
WHERE
{
  ?s dm:location ?location ;
     dm:cost ?cost .
  FILTER (?cost IN (8, 12, 10)) .
}
```

SPARQL by Example – ex112.rq

```
# filename: ex112.rq
PREFIX dm:  <http://learningsparql.com/ns/demo#>
PREFIX db: <http://dbpedia.org/resource/>

SELECT ?s ?cost ?location
WHERE
{
  ?s dm:location ?location ;
     dm:cost ?cost .
  FILTER (?location NOT IN (db:Montreal, db:Lisbon)) .
}
```

RESULTS

s	cost	location
<http://learningsparql.com/ns/data#item693>	10	"Heidelberg"
<http://learningsparql.com/ns/data#item432>	8	db:Boston