

# Semantic Data

## Practice 7 : SPARQL

Jean-Louis Binot

# Introduction

## □ Types of exercises

- *Jungle Book* exercises (handmade dataset).
- Exercises on DBPedia and other sources (external datasets) – left as reading / suggested exercises.

Note : results of executing the queries may have changed wrt what is shown on these slides.

## □ Useful links :

- *Jungle Book* dataset: <http://www.montefiore.ulg.ac.be/~binot/INFO8005/Junglebook-practice.rdf>
- Redland rascal SPARQL endpoint to test queries: <http://librdf.org/query/>
- DBPedia SNORQL endpoint to test DBPedia queries : <http://dbpedia.org/snorql/>  
(select XML output).

## □ Prefixes for *Jungle Book* exercises (need to be added to each solution) :

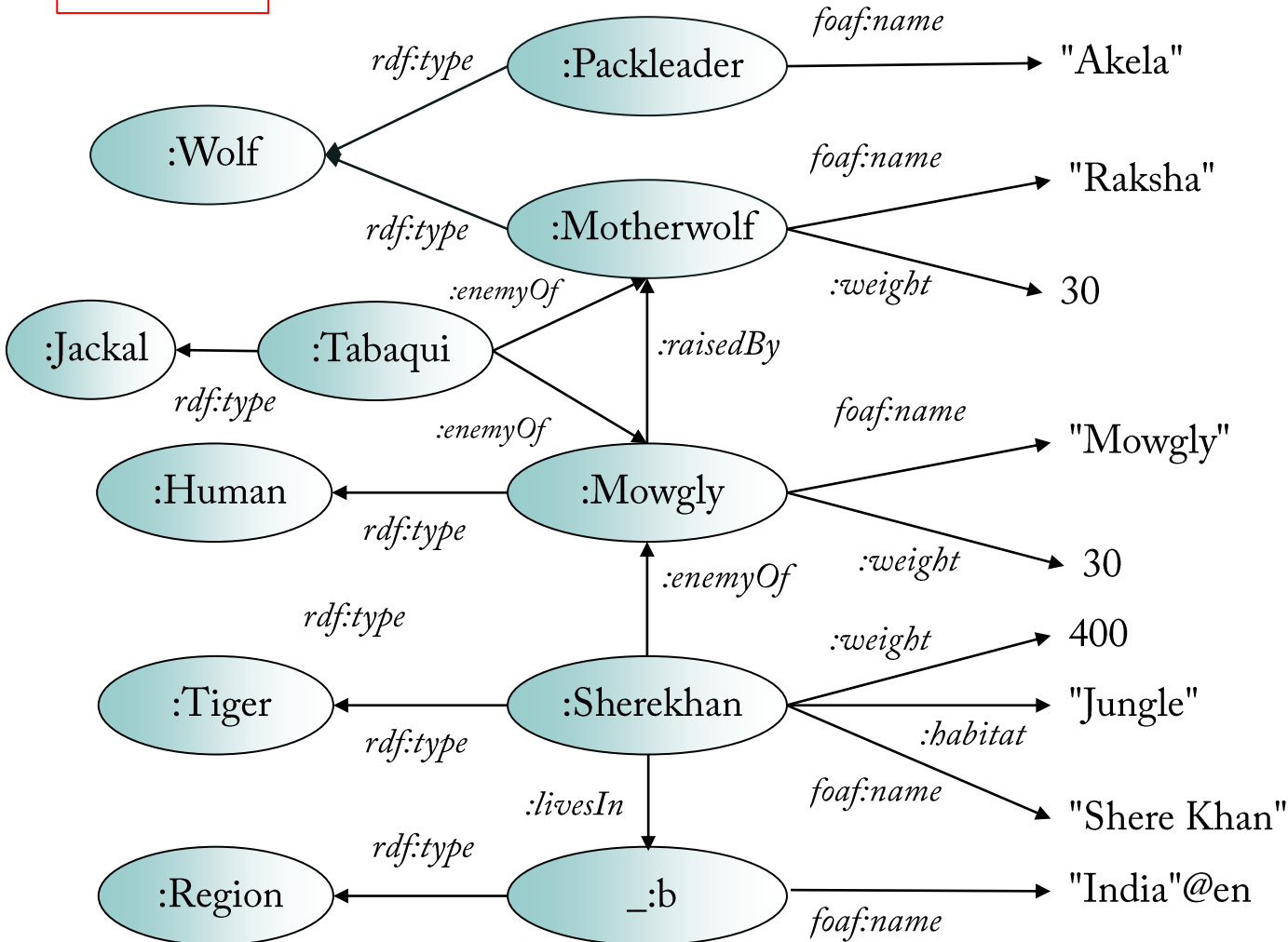
PREFIX : <<http://www.example.org/>>

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

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

# Section 1 – querying the example jungle book dataset

## Dataset



□ Exercise 1 : (to get started)

Find instances of Tiger.

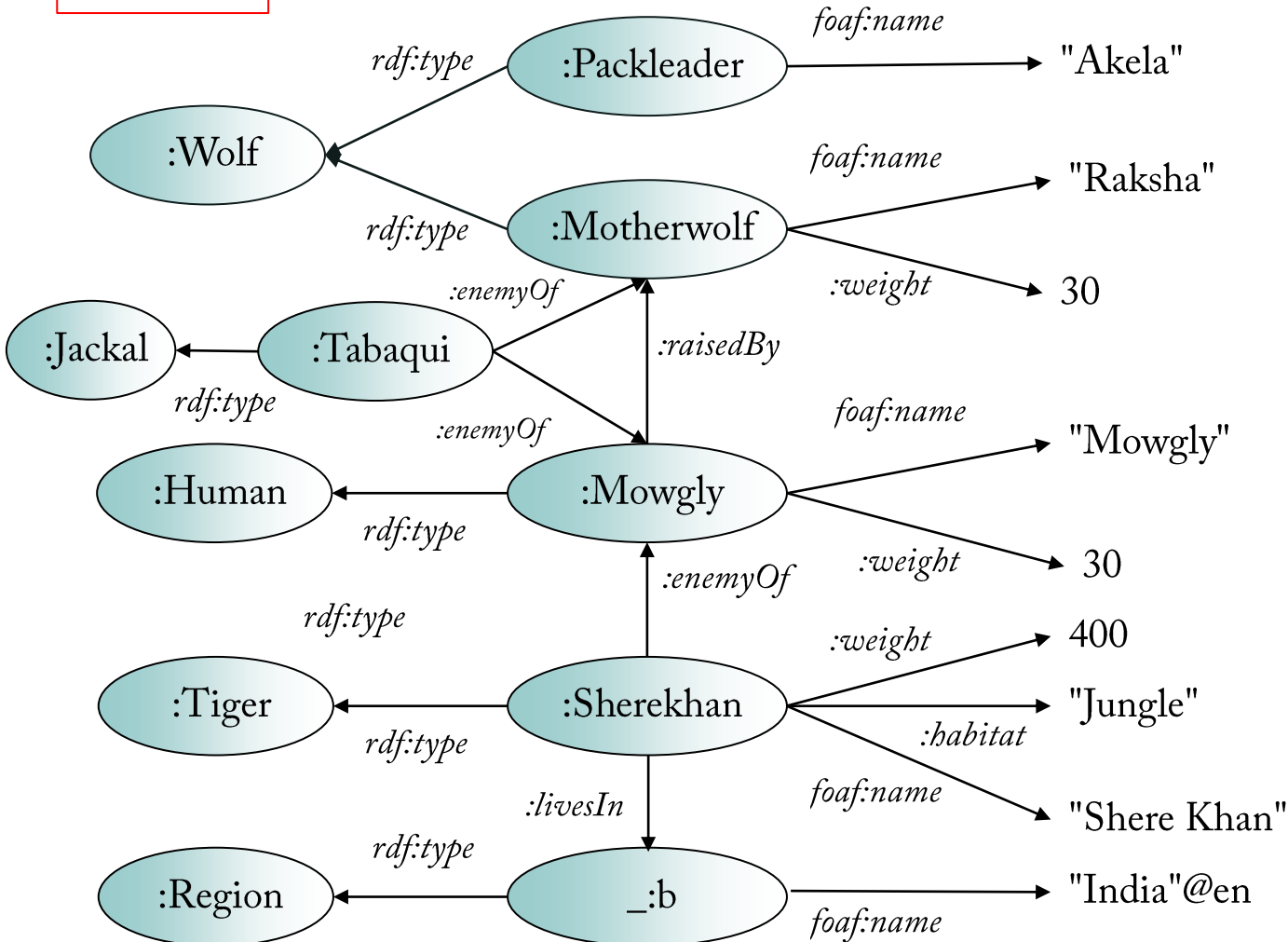
```

SELECT ?instance
WHERE {
  ?instance a :Tiger
}
  
```

Count	instance
1	<a href="http://www.example.org/Sherekhan">http://www.example.org/Sherekhan</a>

# Section 1 – querying the example jungle book dataset

## Dataset



## Exercise 2 :

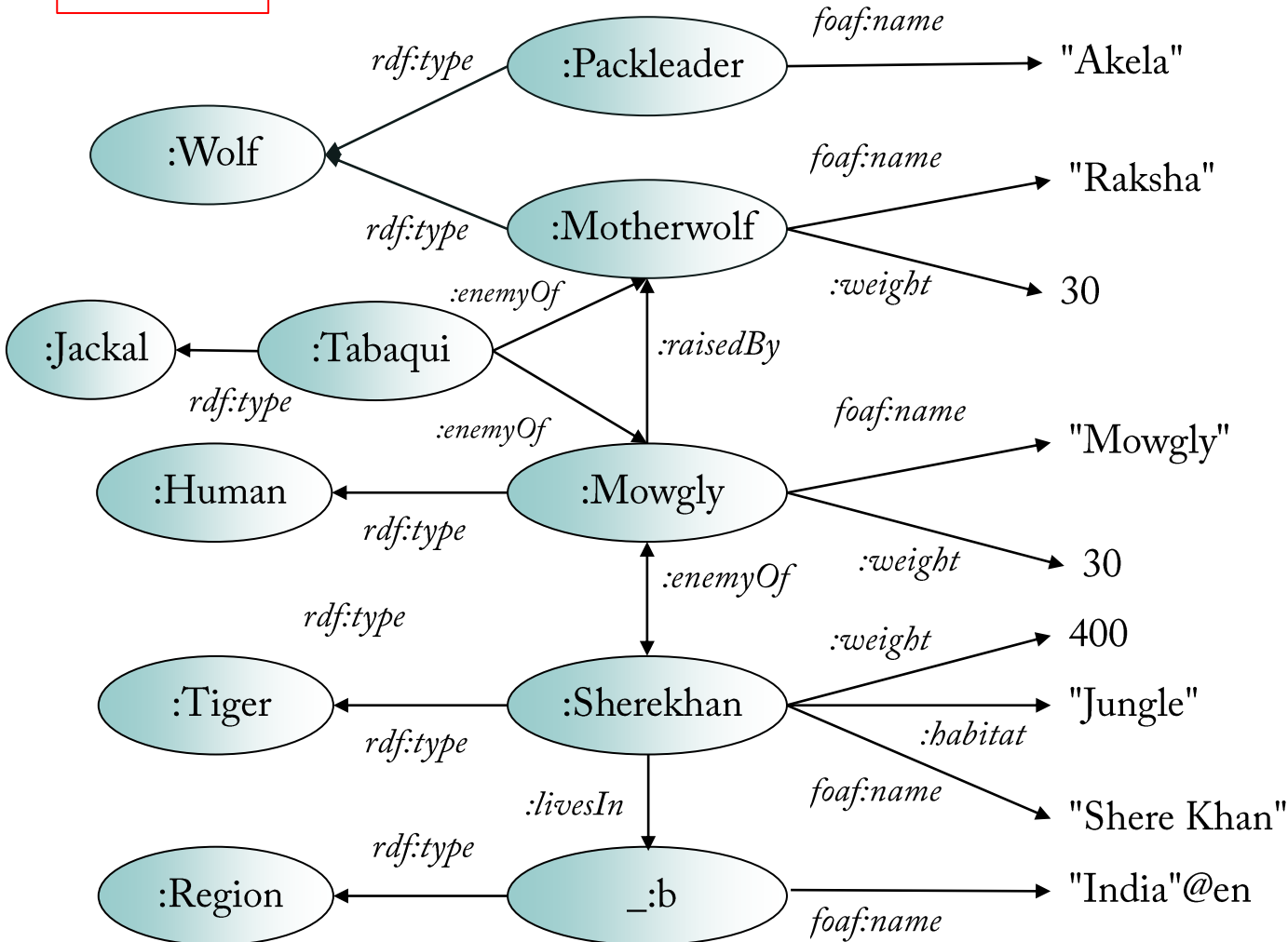
Find who (by name) raised "Mowgly".

```
SELECT ?name
WHERE {
  ?x foaf:name ?name .
  ?m :raisedBy ?x .
  ?m foaf:name "Mowgly" .
}
```

Count	name
1	Raksha

# Section 1 – querying the example jungle book dataset

## Dataset



### Exercise 3 :

Find which enemy of “Mowgly” is a tiger.

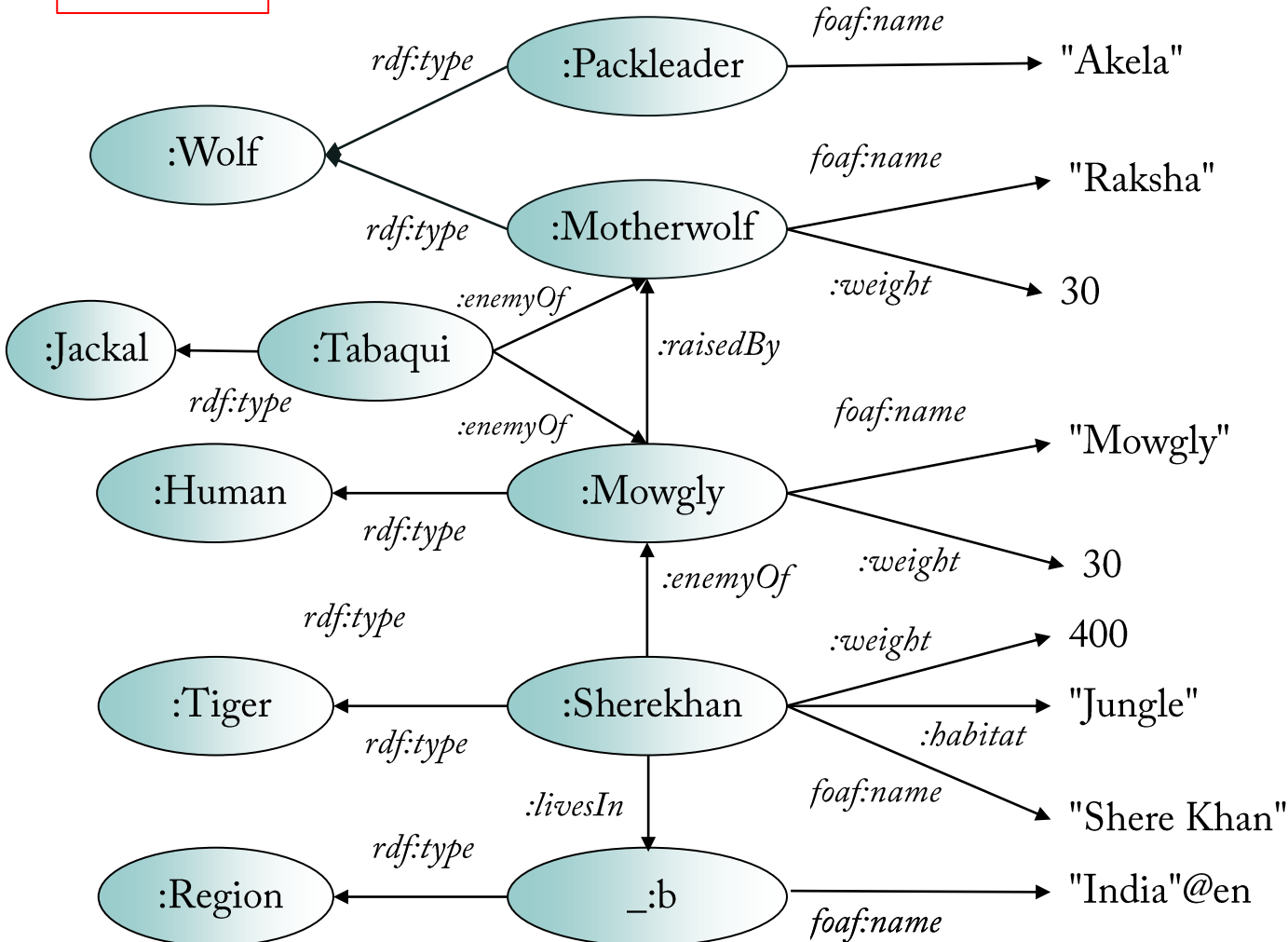
```

SELECT ?enemy
WHERE {
  ?m foaf:name "Mowgly" .
  ?enemy :enemyOf ?m .
  ?enemy a :Tiger .
}
    
```

Count	enemy
1	<a href="http://www.example.org/Sherekhan">http://www.example.org/Sherekhan</a>

# Section 1 – querying the example jungle book dataset

## Dataset



### Exercise 4 :

Find how many enemies “Mowgly” has.

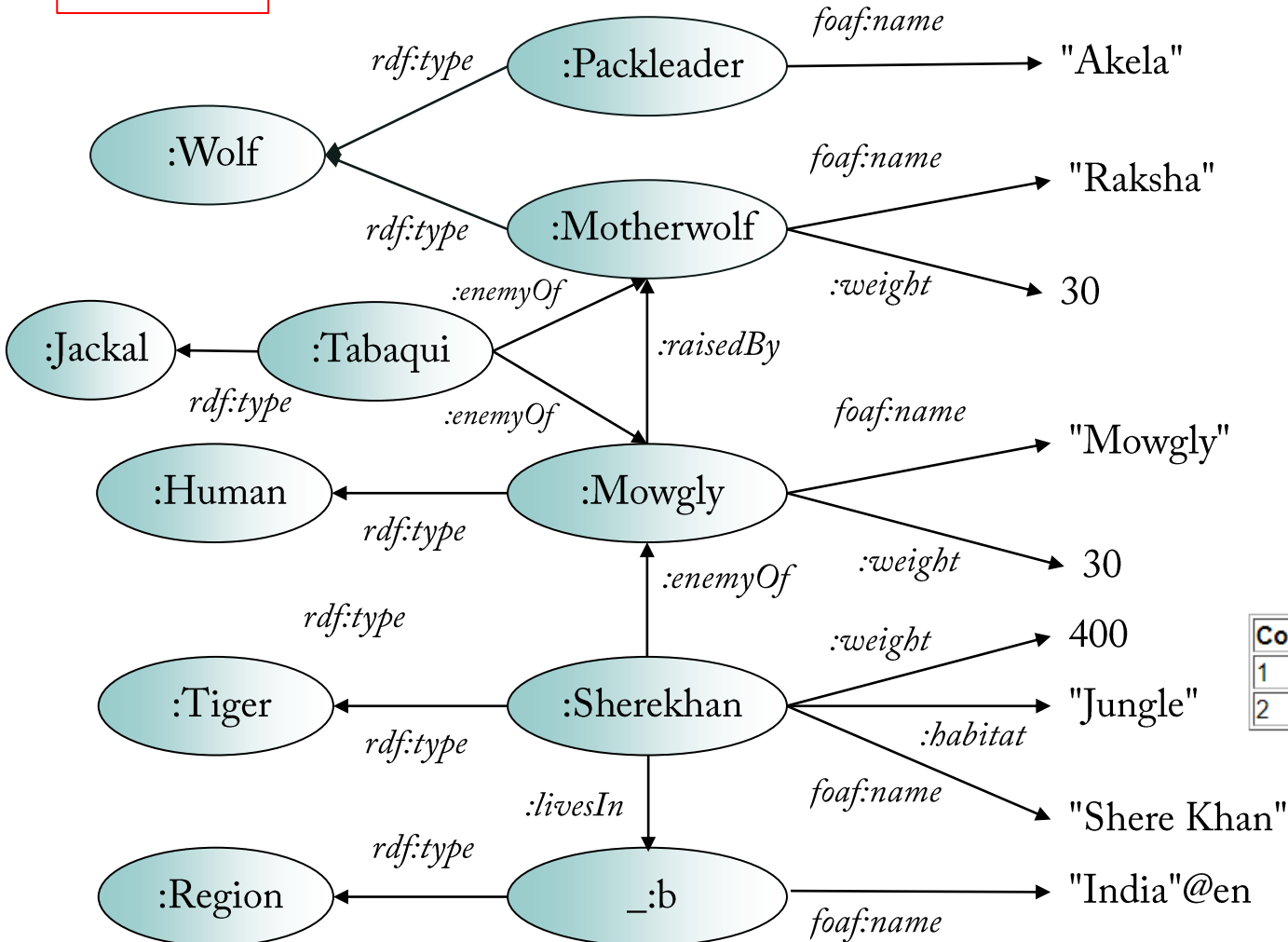
```

SELECT (COUNT(?enemy) AS ?enemies)
WHERE {
    ?person foaf:name "Mowgly".
    ?enemy :enemyOf ?person .
}
    
```

Count	enemies
1	2 <sup>^^</sup> <http://www.w3.org/2001/XMLSchema#integer>

# Section 1 – querying the example jungle book dataset

## Dataset



### Exercise 5 :

Find how many individual weights are in the dataset. Count them by weight value.

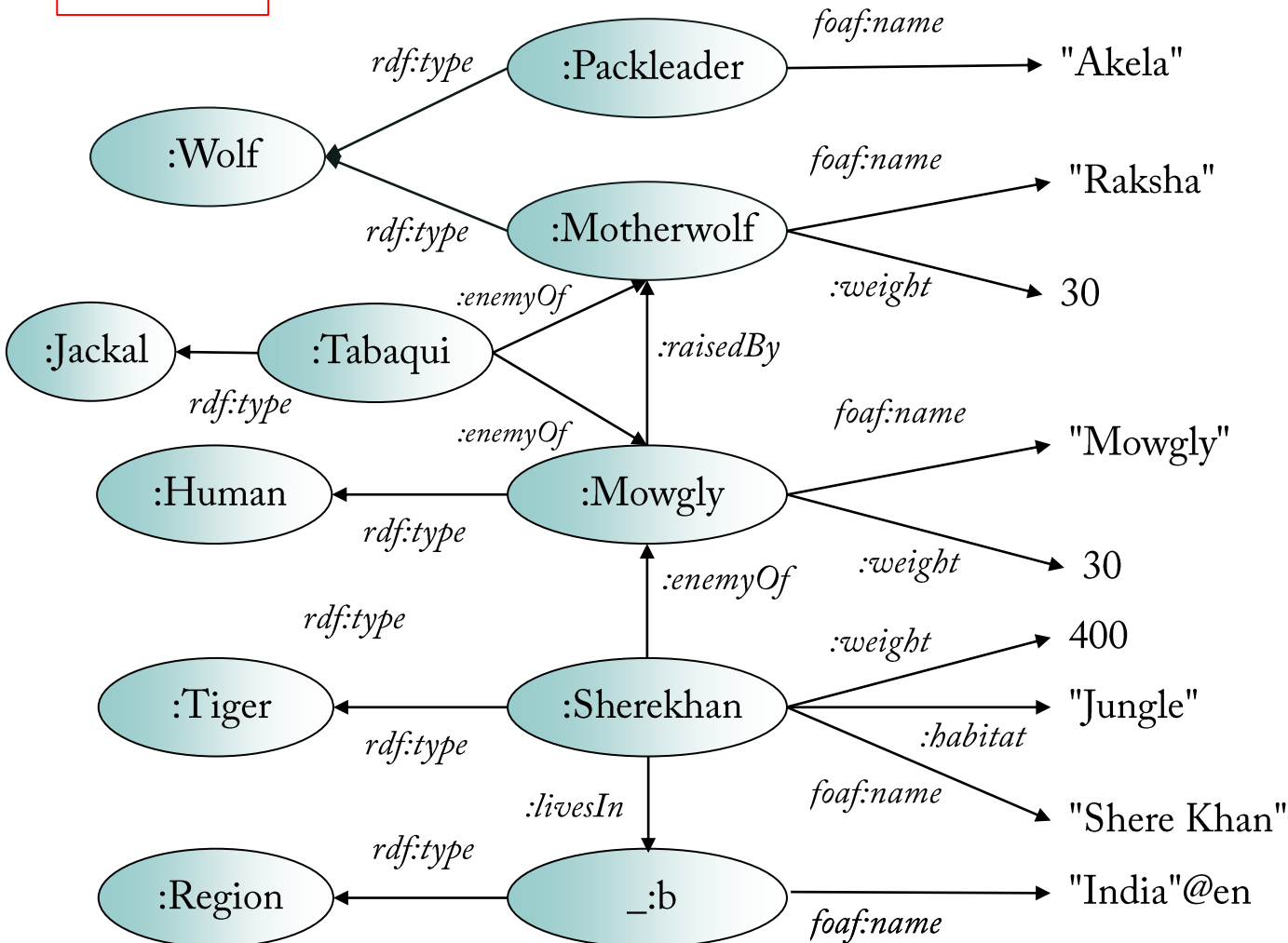
```

SELECT ?weight (COUNT(?weight) AS ?total)
WHERE {
    ?person :weight ?weight .
} GROUP BY ?weight
    
```

Count	weight	total
1	30^^<http://www.w3.org/2001/XMLSchema#integer>	2^^<http://www.w3.org/2001/XM
2	400^^<http://www.w3.org/2001/XMLSchema#integer>	1^^<http://www.w3.org/2001/XM

# Section 1 – querying the example jungle book dataset

## Dataset



### Exercise 6 :

Find how many enemies “Mowgly” has of each type.

```

SELECT ?type (COUNT(?enemy) AS ?total)
WHERE {
  ?person foaf:name "Mowgly" .
  ?enemy :enemyOf ?person .
  ?enemy a ?type .
} GROUP BY ?type
  
```

type	total
:Jackal	1 <sup>^^xsd:integer</sup>
:Tiger	1 <sup>^^xsd:integer</sup>

(Note: this result has been generated with Twinkle, which displays numbers in shorter format if the xsd prefix is added).



# Section 2 : querying DBpedia

❑ Exercise 1 : find out everything in DBpedia about Marie\_Curie

❑ Hints:

- Use [DBpedia SNORQL endpoint](#) to test the query.
- A number of prefixes are already defined.

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#>

results

SPARQL results:

s	p	
<a href="#">:Marie_Curie</a>	<a href="#">owl:sameAs</a>	<a href="http://wikidata.dbpedia.org/resource/Q7186">http://wikidata.dbpedia.org/resource/Q7186</a>
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia2:title</a>	"Marie Curie – More than Meets the Eye"
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia:ontology/abstract</a>	"Marie Skłodowska Curie (Varsóvia, 7 de novembro de 1867 — Passy, Sallanches, 4 de julho de 1935) foi uma física polonesa e química francesa. Ela se tornou a primeira mulher a ser enterrada por méritos próprios no Panteão de Paris. Seu marido Pierre Curie e o físico Henri Becquerel. A cientista também foi laureada com o Prêmio Nobel de Física em 1903 e o Nobel de Química em 1911. Hoje são grandes centros de pesquisa médica. Durante a Primeira Guerra Mundial, fundou o Serviço de Radiologia Móvel, na França, por conta de uma leucemia causada pela exposição a radiação alfa e beta." <a href="#">[editar]</a>
<a href="#">:Marie_Curie</a>	<a href="#">foaf:depiction</a>	<a href="http://commons.wikimedia.org/wiki/Special:FilePath/Marie_Curie_c1920.jpg">http://commons.wikimedia.org/wiki/Special:FilePath/Marie_Curie_c1920.jpg</a>
<a href="#">:Marie_Curie</a>	<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/Object10002684</a>
<a href="#">:Marie_Curie</a>	<a href="#">&lt;http://purl.org/dc/terms/subject&gt;</a>	<a href="#">:Category:Corresponding_Members_of_the_Russian_Academy_of_Sciences_(1917–25)</a>
<a href="#">:Marie_Curie</a>	<a href="#">&lt;http://purl.org/dc/terms/subject&gt;</a>	<a href="#">:Category:Nuclear_chemists</a>
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia:ontology/knownFor</a>	<a href="#">:Radioactivity</a>
<a href="#">:Marie_Curie</a>	<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatPolishNobelLaureates</a>
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia:ontology/wikiPageRedirects</a>	<a href="#">:Marie_Curie</a>
<a href="#">:Marie_Curie</a>	<a href="#">foaf:gender</a>	"female"
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia:ontology/award</a>	<a href="#">:Davy_Medal</a>
<a href="#">:Marie_Curie</a>	<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/Wikicat19th-centuryChemists</a>
<a href="#">:Marie_Curie</a>	<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatPolishScientists</a>
<a href="#">:Marie_Curie</a>	<a href="#">&lt;http://purl.org/dc/terms/description&gt;</a>	"French-Polish physicist and chemist" <a href="#">[editar]</a>
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia2:wordnet_type</a>	<a href="http://www.w3.org/2006/03/wn/w20/instances/synset-scientist-noun-1">http://www.w3.org/2006/03/wn/w20/instances/synset-scientist-noun-1</a>
<a href="#">:Pierre_Joliot</a>	<a href="#">dbpedia2:relatives</a>	<a href="#">:Marie_Curie</a>
<a href="#">:Marie_Curie</a>	<a href="#">dbpedia:ontology/birthName</a>	"Maria Salomea Skłodowska"

query

Describe [:Marie\\_Curie](#)

# Section 2 : querying DBpedia

Exercise 2 : find out all statements describing Marie\_Curie as subject.

results

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#>

query

Select \*  
Where  
{:Marie\_Curie ?p ?o}

SPARQL results:

p	
<a href="#">rdf:type</a>	<a href="#">owl:Thing</a>
<a href="#">rdf:type</a>	<a href="#">foaf:Person</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:ontology/Person</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#Agent&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.ontologydesignpatterns.org/ont/dul/DUL.owl#NaturalPerson&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.wikidata.org/entity/Q215627&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.wikidata.org/entity/Q24229398&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.wikidata.org/entity/Q5&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://www.wikidata.org/entity/Q901&gt;</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:ontology/Agent</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:ontology/Scientist</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://schema.org/Person&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://umbel.org/umbel/rc/PersonWithOccupation&gt;</a>
<a href="#">rdf:type</a>	<a href="#">&lt;http://umbel.org/umbel/rc/Scientist&gt;</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatExperimentalPhysicists</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFellowsOfTheGermanAcademyOfSciencesLeopoldina</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFrenchAgnostics</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFrenchChemists</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFrenchNobelLaureates</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFrenchPeople</a>
<a href="#">rdf:type</a>	<a href="#">dbpedia:class/yago/WikicatFrenchPeopleOfPolishDescent</a>

# Section 2 : querying DBpedia

Exercise 3 : find 15 example concepts in the DBpedia dataset.

```
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#>
```

query

```
SELECT DISTINCT ?concept
WHERE {
  ?s a ?concept .
} LIMIT 15
```

results

SPARQL results:

concept
<a href="#">owl:FunctionalProperty</a>
<a href="#">rdf:Property</a>
<a href="#">owl:Thing</a>
<a href="#">owl:Class</a>
<a href="#">owl:Ontology</a>
<a href="#">owl:ObjectProperty</a>
<a href="#">owl:DatatypeProperty</a>
<a href="#">foaf:Organization</a>
<a href="#">dbpedia:ontology/Company</a>
<a href="#">foaf:Person</a>
<a href="#">dbpedia:ontology/Activity</a>
<a href="#">dbpedia:ontology/Name</a>
<a href="#">dbpedia:ontology/Person</a>
<a href="#">dbpedia:ontology/Actor</a>
<a href="#">dbpedia:ontology/Place</a>

## Section 2 : querying DBpedia

- ❑ Exercise 4 : find in DBPedia +YAGO the islands with population above 1000000; names in English.
- ❑ Hints :
  - Yago is an ontology linked to DBPedia adding more spatial and temporal information
  - Islands have the type *WikicatIslandCountries*, defined in <http://dbpedia.org/class/yago>
  - Population can be accessed by property *populationEstimate* defined in <http://dbpedia.org/property>
  - Use the prefixes of previous slides and add those necessary.
  - Use `rdfs:label` as property for names;
  - To get names in English use a boolean condition `langMatches(lang(?name), "EN")` in a FILTER.

query

```
PREFIX type: <http://dbpedia.org/class/yago/>
PREFIX prop: <http://dbpedia.org/property/>
SELECT ?name ?population
WHERE {
    ?country a type:WikicatIslandCountries;
             rdfs:label ?name ;
             prop:populationEstimate ?population .
FILTER (?population > 1000000 && langMatches(lang(?name), "EN")) .}
```

results

SPARQL results:

name	population
"England"@en	54786300
"Northern Ireland"@en	1864000
"Scotland"@en	5373000

## Section 2 : querying other sources

Explanations for exercise 5 (stated next page) :

- ❑ **RSS (Rich Site Summary, originally RDF Site Summary, often called Really Simple Syndication)** is a type of [web feed](#) which allows users to access updates to online content in a standardized, computer-readable format. These feeds can, for example, allow a user to keep track of many different websites in a single news aggregator. The news aggregator will automatically check the RSS feed for new content, allowing the content to be automatically passed from website to website or from website to user. This passing of content is called web syndication. Websites usually use RSS feeds to publish frequently updated information, such as blog entries, news headlines, audio, video. An RSS document (called "feed", "web feed", or "channel") includes full or summarized text, and metadata, like publishing date and author's name.
- ❑ The **Atom Syndication Format** is an XML language used for web feeds.
- ❑ This exercises also illustrates the fact that you may have to investigate more in depth the structure of external data sources in order to build your query.

## Section 2 : querying other sources

- ❑ Exercise 5 : What are the last 10 updated items in an atom feed (Tim Bray's feed on music) ?
- ❑ How to build the solution :
  - Use Redland rascal SPARQL endpoint.
  - Data source : <http://www.tbray.org/ongoing/ongoing.atom>.
  - Use property *updated* defined in <http://www.w3.org/2005/Atom> (*Atom is part of the prefix !*).
  - Use property *title* defined in <http://purl.org/rss/1.0/>.

query

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

```
PREFIX rss: <http://purl.org/rss/1.0/>
```

```
PREFIX atom: <http://www.w3.org/2005/Atom>
```

```
SELECT ?item ?title ?date
```

```
WHERE {
```

```
?item rdf:type rss:item .
```

```
?item rss:title ?title .
```

```
?item atom:updated ?date }
```

```
ORDER BY DESC(?date)
```

```
LIMIT 10
```

Count	item	title	date
1	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/16/SotD-The-Dry-Cleaner">https://www.tbray.org/ongoing/When/201x/2018/03/16/SotD-The-Dry-Cleaner</a>	SotD: The Dry Cleaner	2018-03-16T09:20:36-07:00
2	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/15/SotD-Downpressor-Man">https://www.tbray.org/ongoing/When/201x/2018/03/15/SotD-Downpressor-Man</a>	SotD: Downpressor Man	2018-03-15T07:44:12-07:00
3	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/14/SotD-Barretts-Privateers">https://www.tbray.org/ongoing/When/201x/2018/03/14/SotD-Barretts-Privateers</a>	SotD: Barretts Privateers	2018-03-14T00:27:40-07:00
4	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/13/SotD-Gimme-Sympathy">https://www.tbray.org/ongoing/When/201x/2018/03/13/SotD-Gimme-Sympathy</a>	SotD: Gimme Sympathy	2018-03-13T16:52:33-07:00
5	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/12/SotD-K515">https://www.tbray.org/ongoing/When/201x/2018/03/12/SotD-K515</a>	SotD: K.515	2018-03-12T09:47:23-07:00
6	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/11/SotD-After-Midnight">https://www.tbray.org/ongoing/When/201x/2018/03/11/SotD-After-Midnight</a>	SotD: After Midnight	2018-03-11T10:19:25-07:00
7	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/10/SotD-Atomic">https://www.tbray.org/ongoing/When/201x/2018/03/10/SotD-Atomic</a>	SotD: Atomic	2018-03-10T09:08:54-08:00
8	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/09/SotD-Phase-Dance">https://www.tbray.org/ongoing/When/201x/2018/03/09/SotD-Phase-Dance</a>	SotD: Phase Dance	2018-03-09T15:41:24-08:00
9	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/08/SotD-Love-Me-Like-A-Man">https://www.tbray.org/ongoing/When/201x/2018/03/08/SotD-Love-Me-Like-A-Man</a>	SotD: Love Me Like a Man	2018-03-08T22:37:54-08:00
10	<a href="https://www.tbray.org/ongoing/When/201x/2018/03/07/SotD-Riding-the-Rocket">https://www.tbray.org/ongoing/When/201x/2018/03/07/SotD-Riding-the-Rocket</a>	SotD: Riding On The Rocket	2018-03-07T07:12:38-08:00

# Suggested exercises

## □ Suggested exercise 1 : querying DBpedia

Using the following name spaces and resources :

PREFIX : `<http://dbpedia.org/resource/>`

URIs of individual resources

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

PREFIX dbpedia2: `http://dbpedia.org/property/`

definitions of data properties

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

definitions of classes and object properties

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

Note : most prefix definitions, except `dbo`, are already predefined in the SNORQL endpoint.

Resources : `dbo:Person`, `foaf:name`, `dbo:SoccerPlayer`, `dbo:birthPlace`, `dbo:position`, `:Steve_Jobs`

- a) Find all information on Steve Jobs.
- b) Find 100 distinct names of persons in DBpedia.
- c) Find in alphabetic order all names of Belgian soccer players in DBpedia.  
Add their position on the field.

Test your result on the SNORQL SPARQL endpoint.

# Suggested exercises

## □ Suggested exercise 2 : querying DBpedia

Using the following name spaces and resources :

PREFIX : `<http://dbpedia.org/resource/>`

URIs of individual resources

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

PREFIX dbpedia2: `http://dbpedia.org/property/`

definitions of data properties

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

definitions of classes and object properties

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

Note : most prefix definitions, except `dbo`, are already predefined in the SNORQL endpoint.

Resources : `dbo:Book`, `dbo:author`, `dbo:publisher`, `dbpedia2:pubDate`, `:Jules_Verne`

- a) Find all books published in 1970.
- b) Find all books authored by Jules Verne, and their publisher.
- c) For the books found in b), find their publication date and sort and count them by publication date.

Test your result on the SNORQL SPARQL endpoint.



THANK YOU