COMP318: SPARQL

www.csc.liv.ac.uk/~valli/Comp318

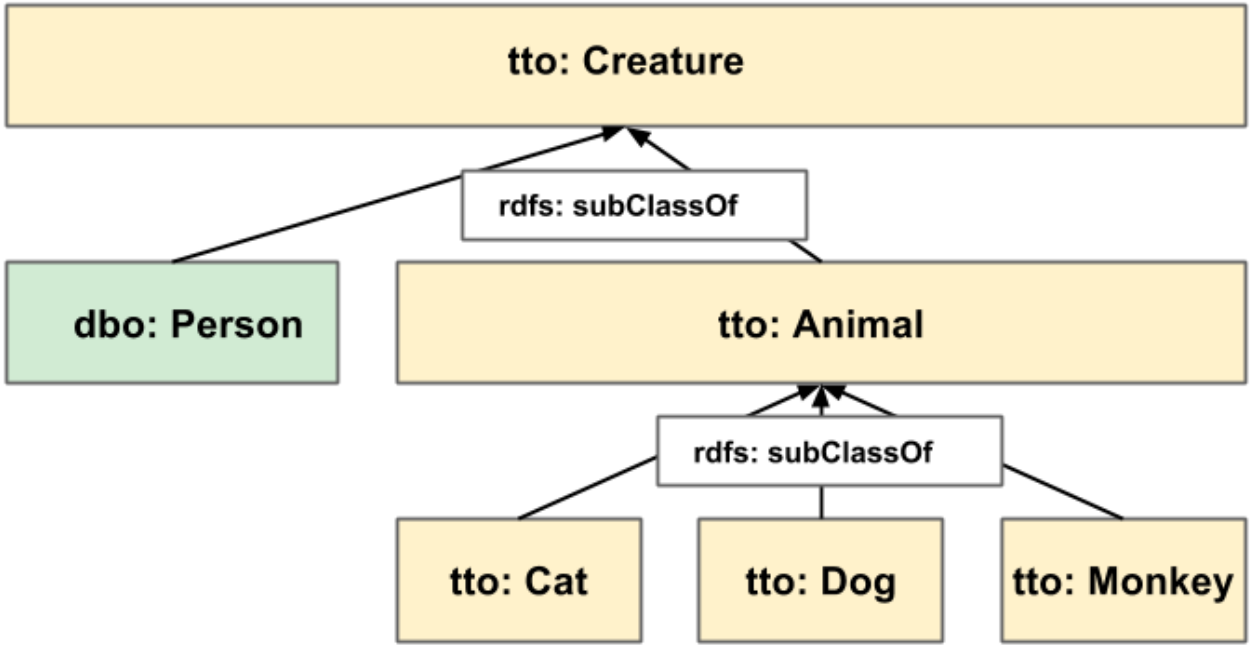
Dr Valentina Tamma
Room: Ashton 2.12
Dept of computer science
University of Liverpool
V.Tamma@liverpool.ac.uk

Where were we

- SPARQL
- Simple entailment in RDF

Data model for the exercises

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

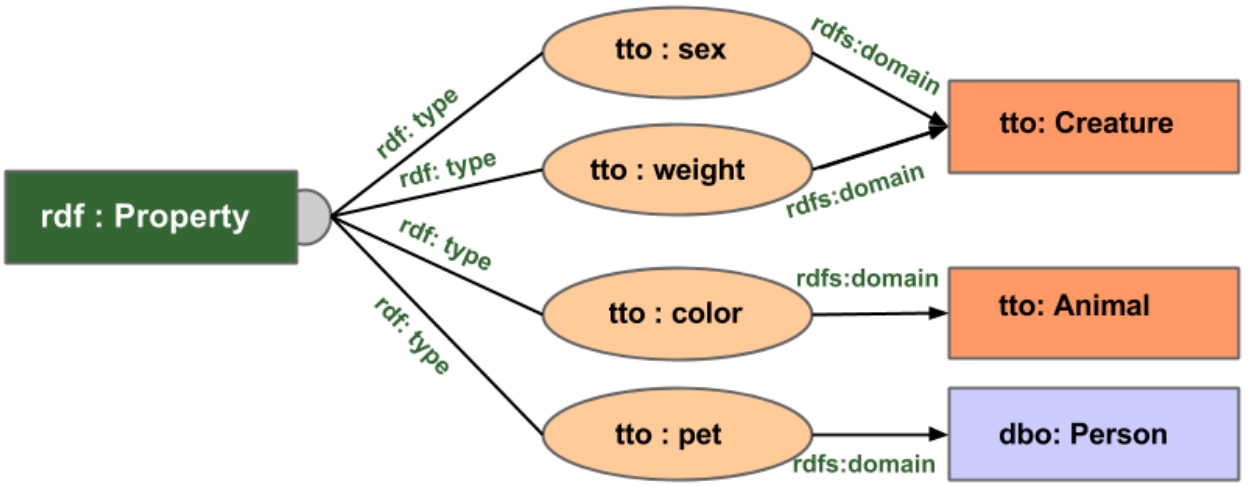


ONTOLOGY

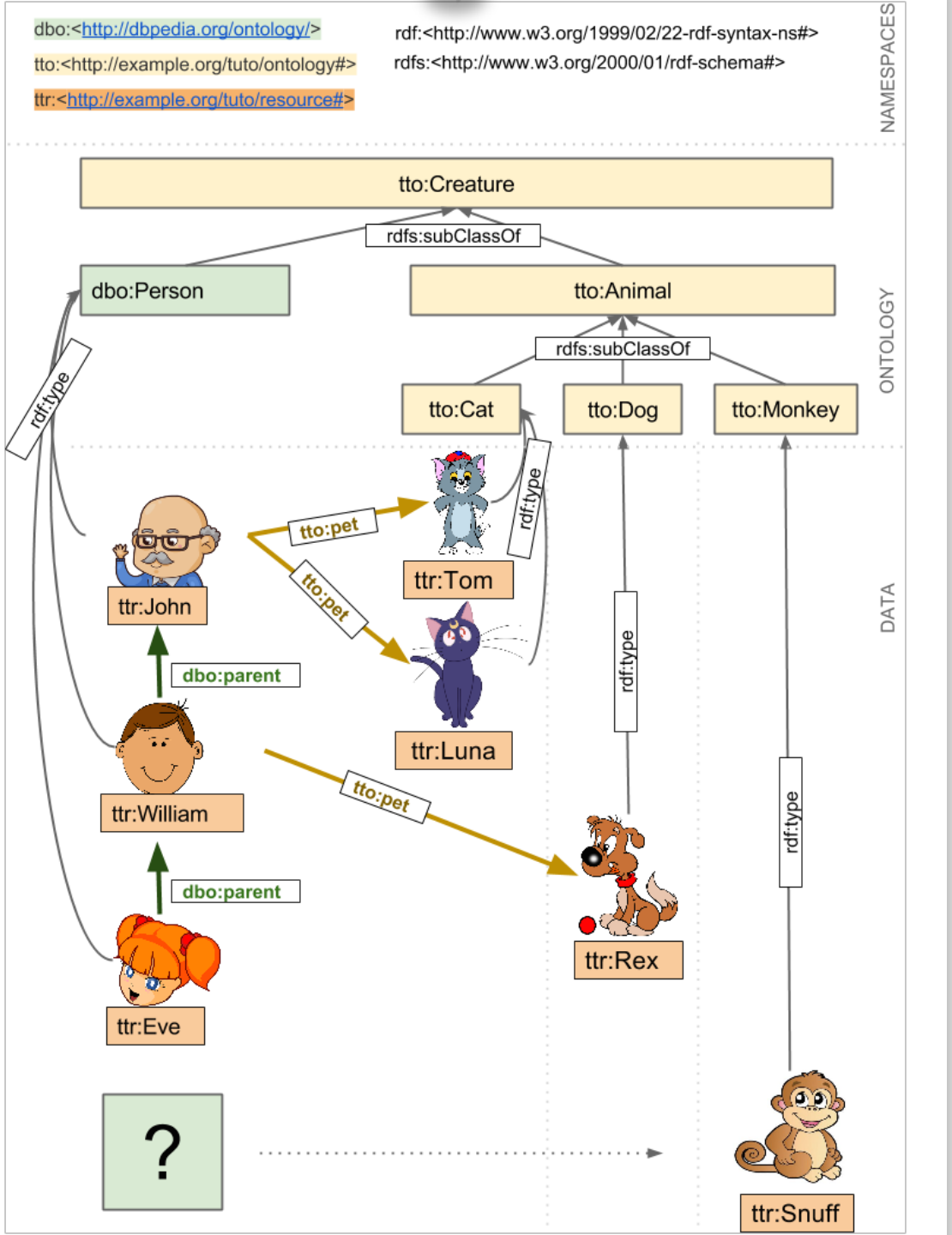
```
tto:Creature
      rdf:type rdfs:Class;
      rdfs:label "creature"^^xsd:string;
      rdfs:isDefinedBy tto: .

dbo:Person
      rdfs:subClassOf tto:Creature .

tto:Animal
      rdf:type rdfs:Class;
      rdfs:label "animal"^^xsd:string;
      rdfs:subClassOf tto:Creature ;
      rdfs:isDefinedBy tto: .

tto:Cat
      rdf:type rdfs:Class;
      rdfs:label "cat"^^xsd:string;
      rdfs:subClassOf tto:Animal ;
      rdfs:isDefinedBy tto: .

tto:Dog
      rdf:type rdfs:Class;
      rdfs:label "dog"^^xsd:string;
      rdfs:subClassOf tto:Animal ;
      rdfs:isDefinedBy tto: .

tto:Monkey
      rdf:type rdfs:Class;
      rdfs:label "monkey"^^xsd:string;
      rdfs:subClassOf tto:Animal ;
      rdfs:isDefinedBy tto: .
```

The properties in the data model

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




```
tto:sex
    rdf:type rdf:Property;
    rdfs:label "sex"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:pet
    rdf:type rdf:Property;
    rdfs:label "domestic animal"^^xsd:string;
    rdfs:domain dbo:Person ;
    rdfs:range tto:Animal ;
    rdfs:isDefinedBy tto: .

tto:color
    rdf:type rdf:Property;
    rdfs:label "hair of furr color"^^xsd:string;
    rdfs:domain dbo:Animal ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:weight
    rdf:type rdf:Property;
    rdfs:label "weight"^^xsd:string;
    rdfs:comment "weight in kilograms"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:decimal ;
    rdfs:isDefinedBy tto: .
```


dbo:<http://dbpedia.org/ontology/>
tto:<http://example.org/tuto/ontology#>
ttr:<http://example.org/tuto/resource#>
rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs:<http://www.w3.org/2000/01/rdf-schema#>
NAMESPACES
tto:Creature
rdfs:subClassOf
dbo:Person
tto:Animal
rdfs:subClassOf
ONTOLOGY
rdf:type
tto:Cat
tto:Dog
tto:Monkey
rdf:type
tto:pet
tto:pet
ttr:Tom
ttr:John
rdf:type
dbo:parent
ttr:Luna
rdf:type
tto:pet
ttr:William
rdf:type
dbo:parent
ttr:Rex
ttr:Eve
?
ttr:Snuff
DATA

The Data

Exploration queries

- What does the data look like?
- RDF triple <s, p, o> in Turtle syntax

```
SELECT *
WHERE {
     ?subject ?predicate ?object .
}
LIMIT 10
```

- Asks for a sample of the data available.
  - Always include the LIMIT keyword to avoid problems of resources and memory

Exploration queries

```
SELECT *
WHERE {
    ?subject ?predicate ?object .
}
LIMIT 10
```

s	p	o
dbo:Person	rdfs:subClassOf	tto:Creature
tto:Animal	rdf:type	rdfs:Class
tto:Animal	rdfs:isDefinedBy	tto:
tto:Animal	rdfs:label	"animal"
tto:Animal	rdfs:subClassOf	tto:Creature
tto:Cat	rdf:type	rdfs:Class
tto:Cat	rdfs:isDefinedBy	tto:
tto:Cat	rdfs:label	"cat"
tto:Cat	rdfs:subClassOf	tto:Animal
tto:Creature	rdf:type	rdfs:Class

Exploration queries

- What properties are used?

```
SELECT DISTINCT ?property
WHERE
{ ?s ?property ?o . }
LIMIT 30
```

Exploration queries

```
SELECT DISTINCT ?property
WHERE
{ ?s ?property ?o . }
LIMIT 30
```

Query time is 0.032[s] for 14 rows

property
rdfs:subClassOf
rdf:type
rdfs:isDefinedBy
rdfs:label
rdfs:domain
rdfs:range
rdfs:comment
dbo:parent
dbp:birthDate
dbp:name
tto:sex
tto:pet
tto:color
tto:weight

SPARQL queries

- We could be matching labels of resources

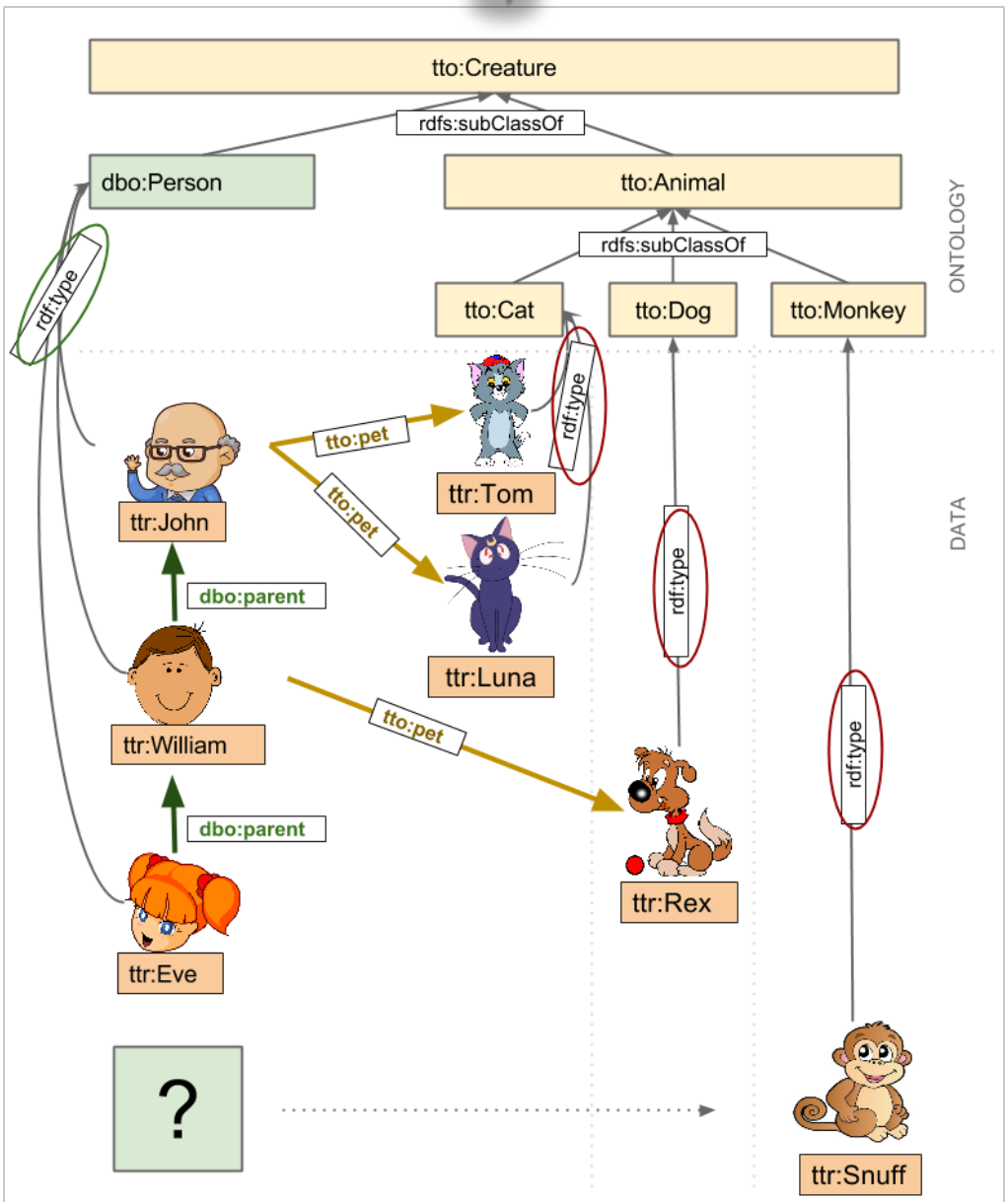
```
?subject rdfs:label ?label .
```

- All parts of a triple can be variables in the query

Example

- Select resources that are persons

```
SELECT ?thing WHERE {
   ?thing rdf:type dbo:Person .
}
```



Query answering wrt the data

Exploration queries

```
SELECT ?thing WHERE {
  ?thing rdf:type dbo:Person .
}
```

Query time is 0.06[s] for 3 rows

thing
ttr:Eve
ttr:John
ttr:William

Example

- Find the labels of the various resources

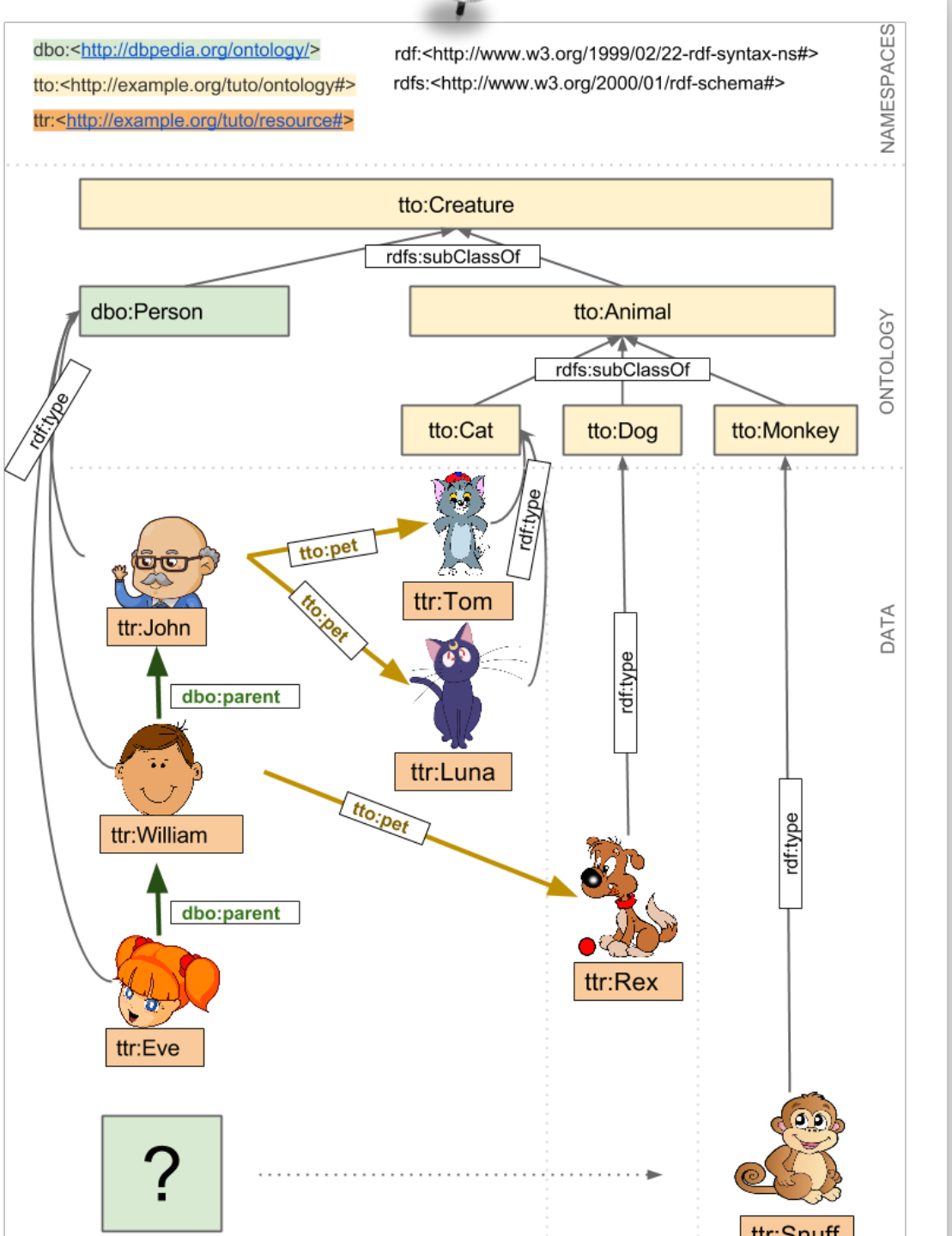
```
SELECT ?subject ?label WHERE {
  ?subject rdfs:label ?label .
}
```

Exploration queries

```
SELECT ?subject ?label WHERE {
  ?subject rdfs:label ?label .
}
```

Query time is 0.041[s] for 8 rows

subject	label
tto:Animal	"animal"
tto:Cat	"cat"
tto:Creature	"creature"
tto:Dog	"dog"
tto:Monkey	"monkey"
tto:pet	"domestic animal"
tto:sex	"sex"
tto:weight	"weight"



Now it's your turn

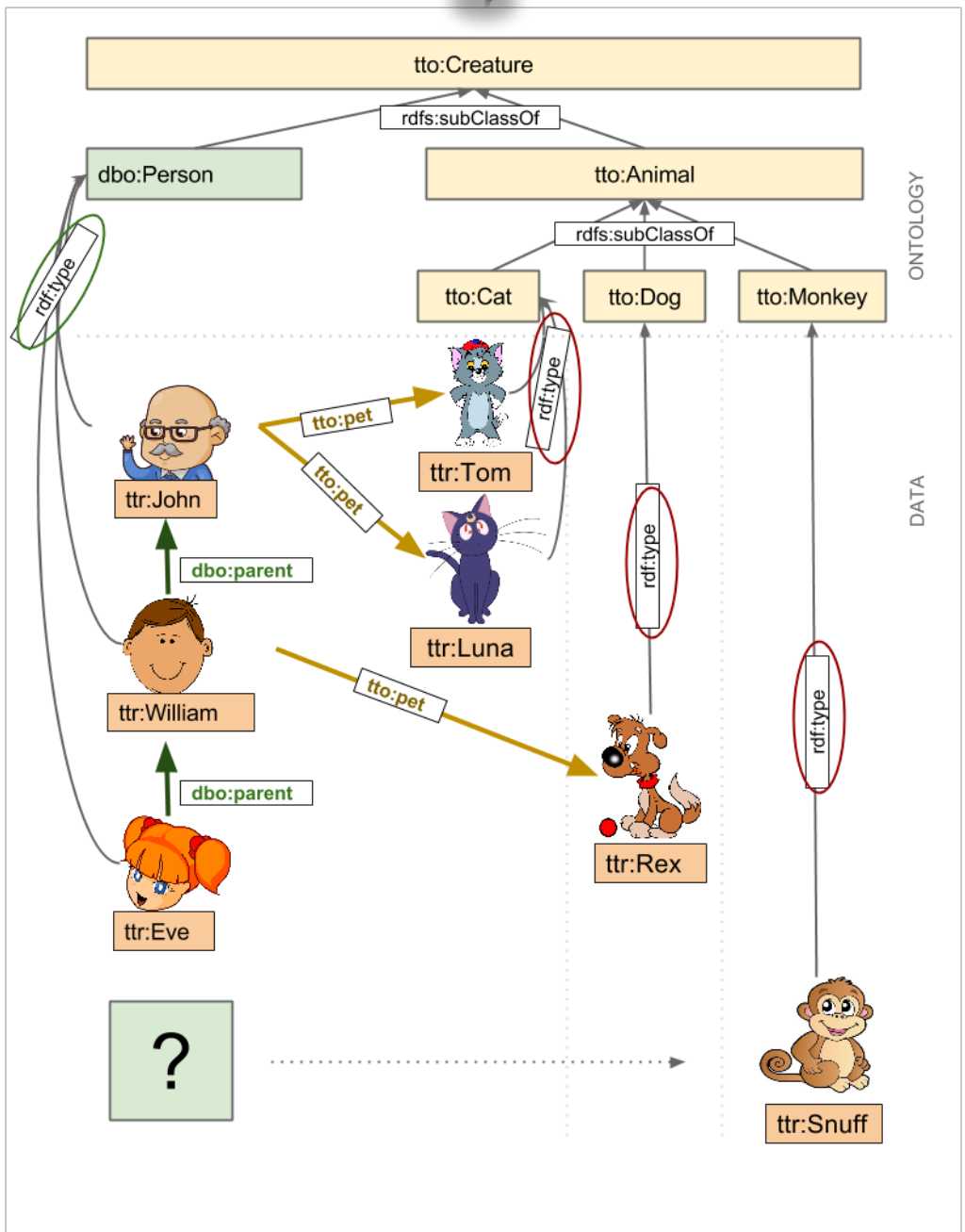
Query 1

- Select things that are cats

Query 1

- Select things that are cats

```
SELECT ?thing WHERE {
  ?thing rdf:type tto:Cat .
}
```



Query answering wrt the data

Query 1

```
SELECT ?thing WHERE {
  ?thing rdf:type tto:Cat .
}
```

Query time is 0.023[s] for 2 rows

thing
ttr:LunaCat
ttr:TomCat

Query 2

- Select all people who are female

Query 2

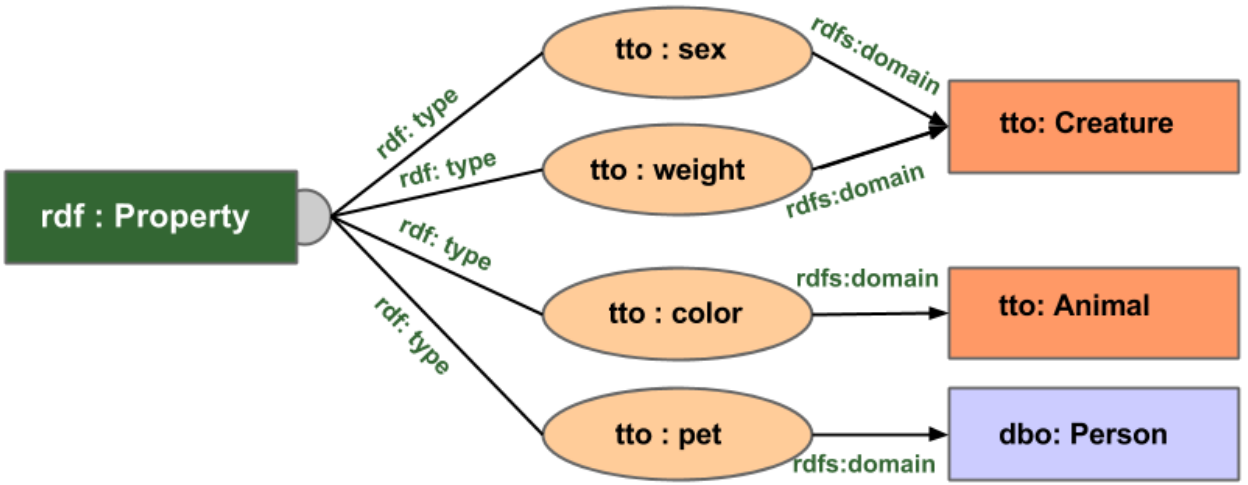
- Select all people who are female

```
SELECT ?thing WHERE {
  ?thing a dbo:Person .
  ?thing tto:sex "female" .
}
```

```
SELECT ?thing WHERE {
 ?thing a dbo:Person ;
         tto:sex "female" .
}
```

Properties

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




```
tto:sex
    rdf:type rdf:Property;
    rdfs:label "sex"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:pet
    rdf:type rdf:Property;
    rdfs:label "domestic animal"^^xsd:string;
    rdfs:domain dbo:Person ;
    rdfs:range tto:Animal ;
    rdfs:isDefinedBy tto: .

tto:color
    rdf:type rdf:Property;
    rdfs:label "hair of furr color"^^xsd:string;
    rdfs:domain dbo:Animal ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:weight
    rdf:type rdf:Property;
    rdfs:label "weight"^^xsd:string;
    rdfs:comment "weight in kilograms"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:decimal ;
    rdfs:isDefinedBy tto: .
```

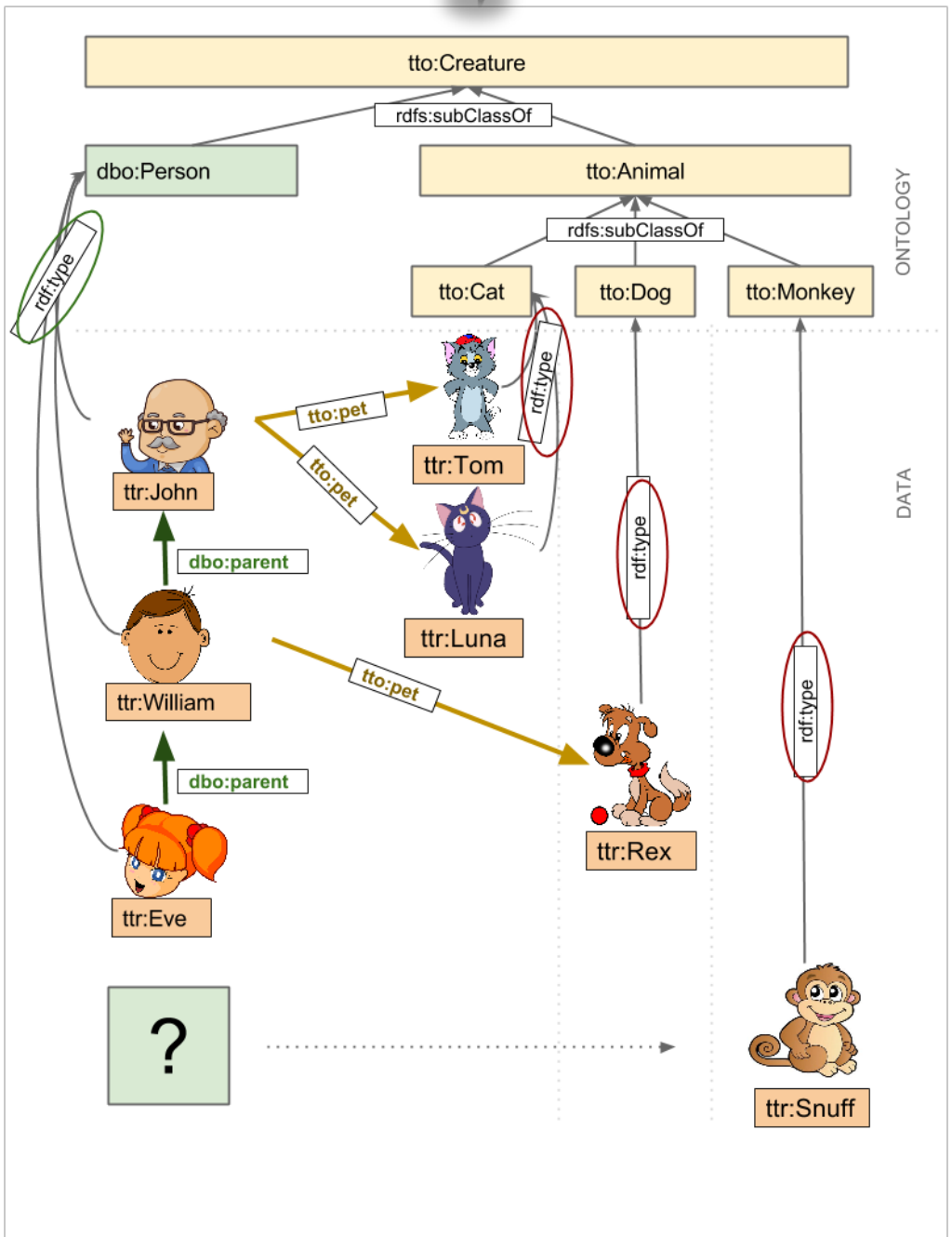
Query 2

```
SELECT ?thing WHERE {
 ?thing a dbo:Person ;
          tto:sex "female" .
}
```

thing
ttr:Eve

Query 3

- Select all people and their pets


tto:Creature
rdfs:subClassOf
dbo:Person
tto:Animal
rdfs:subClassOf
tto:Cat
tto:Dog
tto:Monkey
ONTOLOGY
DATA
rdf:type
rdf:type
rdf:type
rdf:type
tto:pet
tto:pet
tto:pet
ttr:John
ttr:Tom
ttr:Luna
ttr:William
ttr:Rex
ttr:Eve
ttr:Snuff
dbo:parent
dbo:parent
?

Query 3

Query 3

- Select all people and their pets

```
SELECT ?person ?pet WHERE {
     ?person rdf:type dbo:Person .
     ?person tto:pet ?pet .
}
```

Query 3

```
SELECT ?person ?pet WHERE {
     ?person rdf:type dbo:Person .
     ?person tto:pet ?pet .
}
```

Query time is 0.046[s] for 3 rows

person	pet
ttr:John	ttr:LunaCat
ttr:John	ttr:TomCat
ttr:William	ttr:RexDog

Query 4

- Select all pets and their owners

Query 4

- Select all animals that are pets and their owners

```
SELECT ?person ?pet WHERE {
     ?pet rdf:type ?x .
     ?x   rdfs:subClassOf tto:Animal.
     ?person tto:pet ?pet .
}
```

Query 4

```
SELECT ?person ?pet WHERE {
     ?pet rdf:type ?x .
     ?x   rdfs:subClassOf
tto:Animal.
     ?person tto:pet ?pet .
}
```

Query time is 0.046[s] for 3 rows

person	pet
ttr:John	ttr:LunaCat
ttr:John	ttr:TomCat
ttr:William	ttr:RexDog

Query 5

- Count all pets by owner

Query 5

- Count all pets by owner

```
SELECT ?owner (count(?pet) as ?count)
WHERE {
   ?owner tto:pet ?pet .
} GROUP BY ?owner
```

Query 5

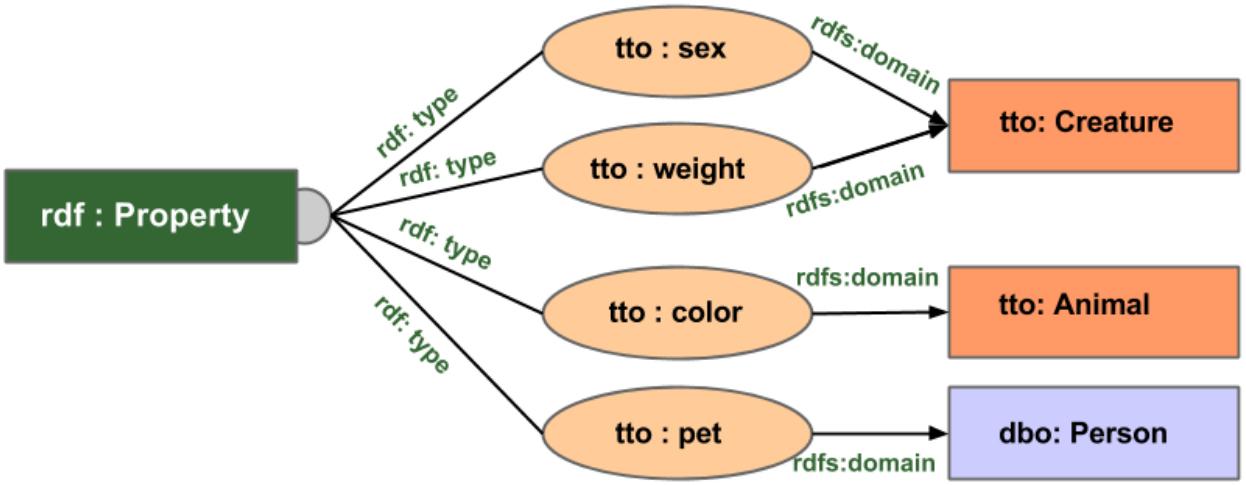
```
SELECT ?owner (count(?pet) as ?
count) {
  ?owner tto:pet ?pet .
} GROUP BY ?owner
```

Query time is 0.06[s] for 2 rows

owner	count
ttr:John	"2"
ttr:William	"1"

The properties in the data model

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




```
tto:sex
    rdf:type rdf:Property;
    rdfs:label "sex"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:pet
    rdf:type rdf:Property;
    rdfs:label "domestic animal"^^xsd:string;
    rdfs:domain dbo:Person ;
    rdfs:range tto:Animal ;
    rdfs:isDefinedBy tto: .

tto:color
    rdf:type rdf:Property;
    rdfs:label "hair of furr color"^^xsd:string;
    rdfs:domain dbo:Animal ;
    rdfs:range xsd:string ;
    rdfs:isDefinedBy tto: .

tto:weight
    rdf:type rdf:Property;
    rdfs:label "weight"^^xsd:string;
    rdfs:comment "weight in kilograms"^^xsd:string;
    rdfs:domain tto:Creature ;
    rdfs:range xsd:decimal ;
    rdfs:isDefinedBy tto: .
```

Query 6

- Select things that have a weight between 5 and 9 kg, order by weight and specify the type

Query6

- Select things that have a weight between 5 and 9 kg, order by weight and specify the type

```
SELECT ?thing ?weight ?type WHERE {
  ?thing tto:weight ?weight .
  ?thing a ?type .
  FILTER (?weight > 5 && ?weight < 9.0)
}ORDER BY ?weight ?type
```

Query 6

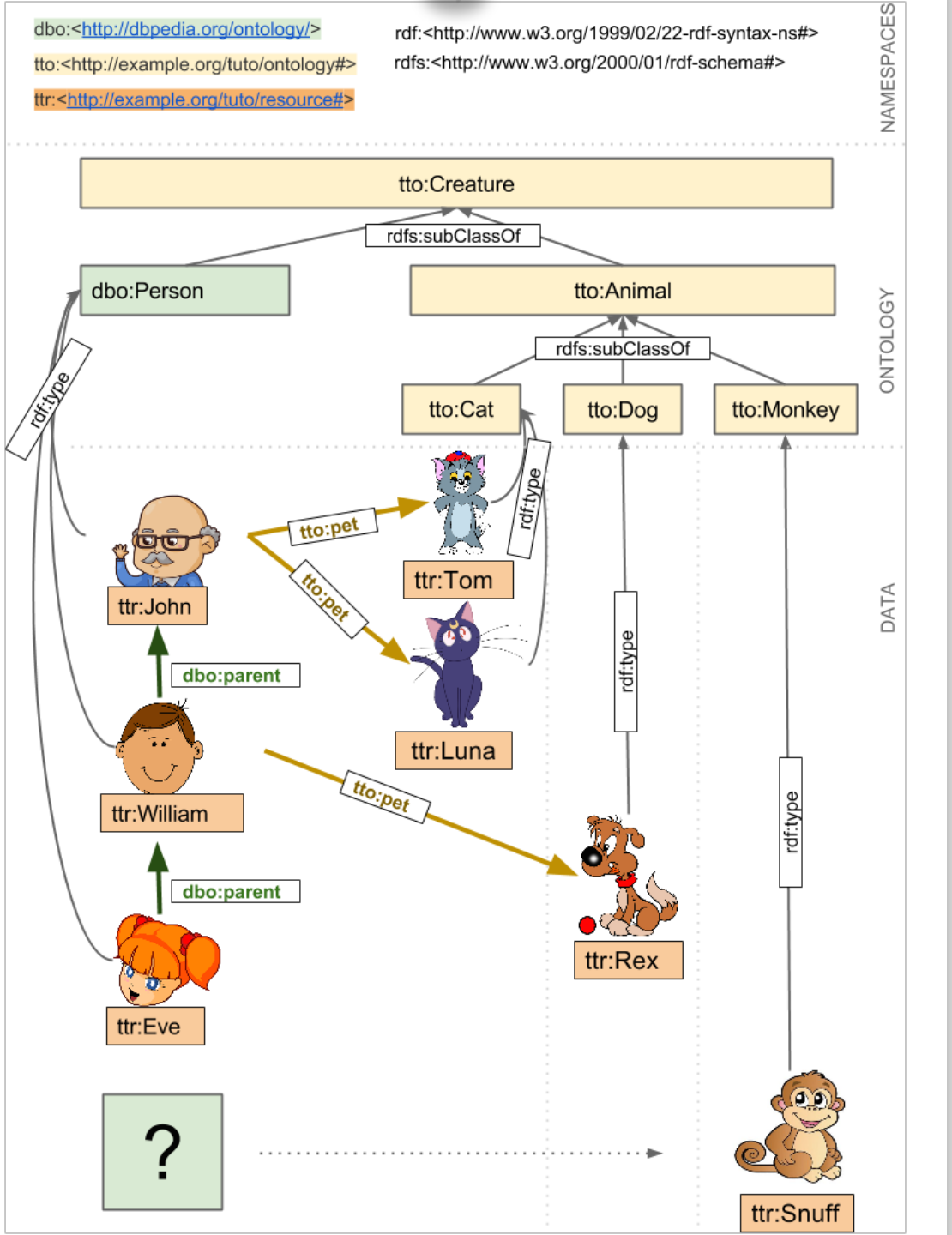
```
SELECT ?thing ?weight ?type WHERE {
  ?thing tto:weight ?weight .
  ?thing a ?type .
  FILTER (?weight > 5 && ?weight < 9.0)
}ORDER BY ?weight ?type
```

Query time is 0.029[s] for 2 rows

thing	weight	type
ttr:TomCat	"5.8"	tto:Cat
ttr:RexDog	"8.8"	tto:Dog

Query 7

- Select the name and year of birth for all the male people


dbo:<http://dbpedia.org/ontology/>
tto:<http://example.org/tuto/ontology#>
ttr:<http://example.org/tuto/resource#>
rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns#>
rdfs:<http://www.w3.org/2000/01/rdf-schema#>
NAMESPACES
tto:Creature
rdfs:subClassOf
dbo:Person
tto:Animal
rdfs:subClassOf
ONTOLOGY
rdf:type
tto:Cat
tto:Dog
tto:Monkey
rdf:type
tto:pet
tto:pet
ttr:Tom
ttr:John
dbo:parent
ttr:Luna
rdf:type
ttr:William
tto:pet
dbo:parent
ttr:Rex
rdf:type
ttr:Eve
?
ttr:Snuff
DATA

The Data

Query 7

- Select the name and year of birth for all the male people

```
select ?name ?yearBorn where {
  ?person rdf:type dbo:Person .
  ?person dbp:birthDate ?birth .
  ?person dbp:name ?name .
  ?person tto:sex "male" .
  bind (year(?birth) as ?yearBorn)
}
```

Query 7

```
select ?name ?yearBorn where {
  ?person rdf:type dbo:Person .
  ?person dbp:birthDate ?birth .
  ?person dbp:name ?name .
  ?person tto:sex "male" .
  bind (year(?birth) as ?yearBorn)
}
```

Query time is 0.025[s] for 2 rows

name	yearBorn
"John"	"1942"
"William"	"1978"

Query 8

- Select the direct and indirect subclass of the class Creature

Query 8

- Select the direct and indirect subclass of the class Creature

```
select ?subSpecies   where {
  ?subSpecies rdfs:subClassOf+ tto:Creature .
}
```

Query 8

```
select ?subSpecies   where {
  ?subSpecies rdfs:subClassOf+ tto:Creature .
}
```

Query time is 0.037[s] for 5 rows

subSpecies
dbo:Person
tto:Animal
tto:Cat
tto:Dog
tto:Monkey

Further reading (recommended)

- Jena SPARQL tutorial:

  `http://jena.sourceforge.net/ARQ/Tutorial/`
- SPARQL Query Language for RDF:

  `http://www.w3.org/TR/rdf-sparql-query/`
- Turtle - Terse RDF Triple Language:

  `http://www.dajobe.org/2004/01/turtle/`
- SPARQL FAQ:

  `http://www.thefigtrees.net/lee/sw/sparql-faq`
- Learn about SPARQL 1.1:

  `http://www.dajobe.org/talks/201105-sparql-11/`
- YASGUI, YASQE: SPARQL query editors:

  `http://yasgui.laurensrietveld.nl/`
  `http://yasgui.org/YASQE/`