

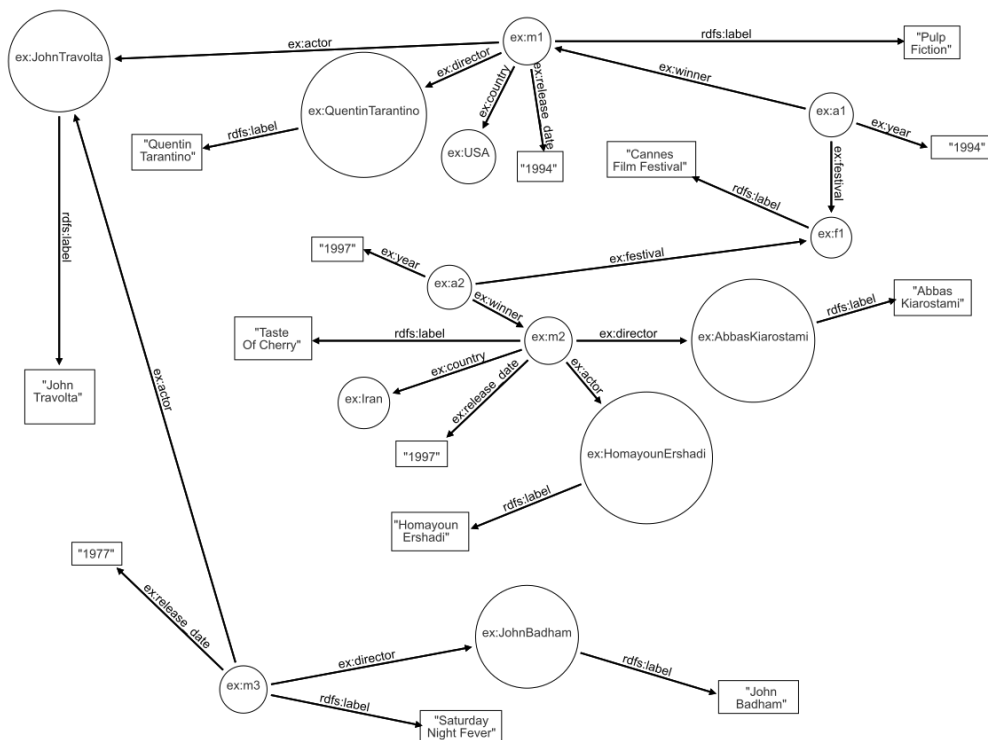
Knowledge Graph Analysis

Solutions to Exercise Sheet 1

Dr. Asja Fischer, Prof. Jens Lehmann

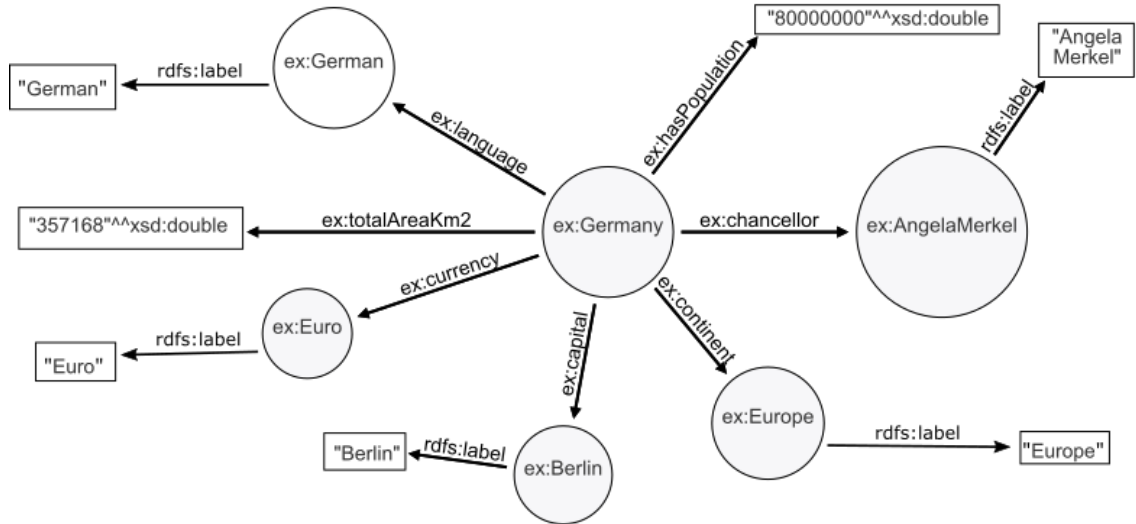
February 6, 2017

1 IN CLASS



1.

2. 1- URIs are used to uniquely identify resources.
- 2- Literals describe data values that do not have an own identity.



```

3. @prefix dbr: <http://dbpedia.org/resource/> .
   @prefix
     rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns-#> .
   @prefix dbo:<http://dbpedia.org/ontology/> .

   dbr:John_Travolta    rdf:type          dbo:Actor ,
                                                                dbo:Producer ;
                                                                dbo:starring    dbr:Basic_(film) ,
                                                                dbr:Shout_(film) .
  
```

```

4. SELECT ?movie_name
   WHERE {
     ?x rdf:type ex:Movie .
     ?x rdfs:label ?movie_name .
   }

   SELECT ?movie_name ?director
   WHERE {
     ?x rdf:type ex:Movie .
     ?x rdfs:label ?movie_name .
     ?x ex:year ?year .
     ?x ex:director ?director .
     ?director rdfs:label ?director_name .
   }
  
```

```

ORDER BY DESC(?year)
SELECT ?movie_name ?director
WHERE {
  ?x rdf:type ex:Movie .
  ?x rdfs:label ?movie_name .
  ?x ex:year ?year .
  ?x ex:director ?director .
  ?director rdfs:label ?director_name .
}
FILTER( ?year <= "1996"^^xsd:gYear )
SELECT ?movie_name
WHERE {
  ?x rdf:type ex:Movie .
  ?x rdfs:label ?movie_name .
  ?x ex:genre ex:Crime .
}
SELECT ?actor_name
WHERE {
  ?x rdf:type ex:Actor .
  ?x ex:birthYear ?year .
  ?x rdfs:label ?actor_name .
}
FILTER ( ?year < "1966"^^xsd:gYear )
SELECT ?movie_name
WHERE {
  ?x rdf:type ex:Movie .
  ?x rdfs:label ?movie_name .
  ?x ex:director ?director .
  ?director ex:birthYear ?year .
  ?director rdfs:label ?director_name .
}
FILTER( ?year <= "1946"^^xsd:gYear )
5. PREFIX ex: <http://example.org/>
SELECT ?object
WHERE
  {{ ex:Sun      ex:satellite  ?object . } UNION
  { ex:Sun      ex:satellite  ?satellite .
    ?satellite  ex:satellite  ?object .}}
PREFIX ex: <http://example.org/>
SELECT ?object
WHERE {
?object ex:satellite ?satellite .
?object ex:name ?name .
FILTER(lang(?name)='en')) . }
PREFIX ex: <http://example.org/>
SELECT ?object

```

```

WHERE {
?object ex:satellite ?satellite .
?object2 ex:satellite ?object . }

PREFIX ex: <http://example.org/>
SELECT ?object
WHERE {
?object ex:satellite ?satellite .
?object ex:name ?name .
FILTER(lang(?name)='en')) .
?object2 ex:satellite ?object .
?object2 ex:radius ?radius .
FILTER(?radius >= 1500) . }

PREFIX ex: <http://example.org/>
SELECT ?object
WHERE {
?object ex:satellite ?satellite1 .
?object ex:satellite ?satellite2 .
FILTER (?satellite1 != ?satellite2 ). }

```

2 AT HOME

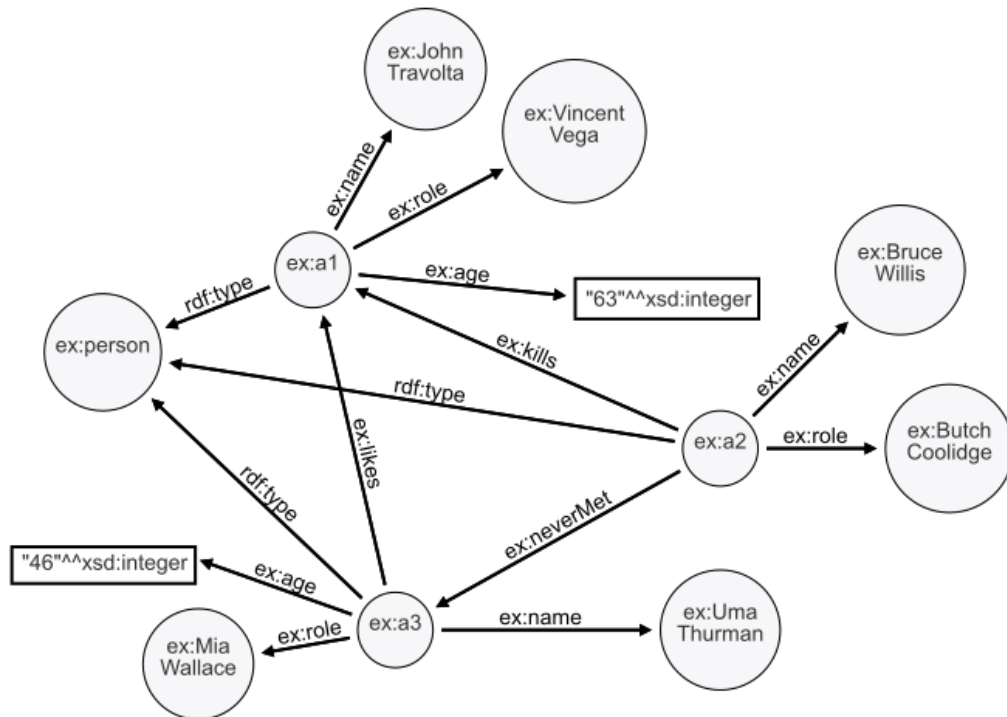
1. No solution necessary.

```

2.
@prefix rdf:<http://www.w3.org/1999/02/22-rdf-syntax-ns-type#>
.
@prefix xsd :<http://www.w3.org/2001/XMLSchema> .
@prefix ex :<http://www.example.org/> .

ex:KGA16    ex:lecturer    ex:JL , ex:AF;
            ex:semester    ex:SW1617 ;
            ex:track      ex:IntelligentSystems ;
            ex:department ex:SDA.
ex:JL      rdfs:label    "Prof. Dr. Jens Lehmann"^^xsd:String .
ex:AF      rdfs:label    "Dr. Asja Fischer"^^xsd:String .
ex:SDA     rdfs:label    "Smart Data Analytics"^^xsd:String.

```



3.

4. a):How tall is Claudia Schiffer?

```
SELECT DISTINCT ?height
WHERE {
    res:Claudia_Schiffer dbo:height ?height .
}
```

b):Give me all female Russian astronauts.

```
SELECT DISTINCT ?uri ?string
WHERE {
    ?uri rdf:type yago:RussianCosmonauts .
    ?uri rdf:type yago:FemaleAstronauts .
    ?uri rdfs:label ?string .
}
```

c):How many monarchical countries are there in Europe?

```
SELECT COUNT(DISTINCT ?uri)
WHERE {
    ?uri rdf:type yago:EuropeanCountries .
    ?uri dbo:governmentType ?govern .
    FILTER regex(?govern, 'monarchy') .
}
```

d): Which states of Germany are governed by the Social Democratic Party?

```
SELECT DISTINCT ?uri ?string
WHERE {
  ?uri rdf:type yago:StatesOfGermany .
  { ?uri dbp:rulingParty 'SPD'@en . }
  UNION
  { ?uri dbp:rulingParty
    res:Social_Democratic_Party_of_Germany.}
  ?uri rdfs:label ?string .
}
```

e): Which monarchs of the United Kingdom were married to a German?

```
SELECT DISTINCT ?uri ?string
WHERE {
  ?uri rdf:type yago:MonarchsOfTheUnitedKingdom .
  ?uri dbo:spouse ?spouse .
  ?spouse dbo:birthPlace res:Germany.
  ?uri rdfs:label ?string.
}
```

f): Which countries have places with more than two caves?

```
SELECT DISTINCT ?uri ?string
WHERE {
  ?cave rdf:type dbo:Cave .
  ?cave dbo:location ?uri .
  ?uri rdf:type dbo:Country .
  ?uri rdfs:label ?string .
}
GROUP BY ?uri ?string
HAVING (COUNT(?cave) > 2)
```

g): Give me all cities in New Jersey with more than 100000 inhabitants.

```
SELECT DISTINCT ?uri ?string
WHERE {
  ?uri rdf:type dbo:City .
  ?uri dbo:isPartOf res:New_Jersey .
  ?uri dbp:populationTotal ?inhabitants .
  FILTER (?inhabitants > 100000) .
  ?uri rdfs:label ?string.
}
```

h): Is proinsulin a protein?

```
ASK
```

```
WHERE {  
  res:Proinsulin rdf:type dbo:Protein .  
}
```

i):Is Frank Herbert still alive?

```
ASK  
WHERE {  
  OPTIONAL { res:Frank_Herbert dbo:deathDate ?date . }  
  FILTER (!BOUND(?date))  
}
```

j):Which mountain is the highest after the Annapurna?

```
SELECT DISTINCT ?uri ?string  
WHERE {  
  res:Annapurna dbo:elevation ?elevation .  
  ?uri rdf:type dbo:Mountain .  
  ?uri dbo:elevation ?otherelevation .  
  FILTER (?otherelevation < ?elevation) .  
  ?uri rdfs:label ?string.  
}  
ORDER BY DESC(?otherelevation) LIMIT 1
```