

Домашно бр. 2  
**SPARQL**

Моника Јовевска 203028

**A) Домашна задача**

**1. Што е SPARQL?**

SPARQL е прашален јазик (query language) за структурирани податоци на веб, поточно податоци достапни во RDF.

**2. Кои типови на SPARQL прашања постојат?**

SELECT, DESCRIBE, ASK, CONSTRUCT, SPARQL UPDATE

**3. Во каква логичка релација се наоѓаат меѓусебно triple pattern-ите кои се наведени во стандарден WHERE дел од SPARQL прашање?**

Triple patterns се поврзани со логичкото AND. Ова значи дека сите triple patterns во WHERE делот мора да биде задоволени, а резултатот ќе вклучува решенија каде што сите patterns се вистинити. Во SQL за таа цел го користиме AND операторот во WHERE делот, а во SPARQL, ова се постигнува автоматски преку наведувањето на повеќе triple patterns во WHERE делот на query-то.

**4. Како можеме да означиме во едно SPARQL прашање дека бараниот triple pattern не е задолжителен при пребарувањето?**

Со OPTIONAL блок

**5. Дали еден SPARQL endpoint може да се користи и од страна на апликации? Објаснете.**

SPARQL може да се користи од страна на апликации, Ова го овозможуваат параметрите “format” и “default-graph-uri”.

## Б) Вовед

DBpedia (<http://www.dbpedia.org>) е проект кој ги извлекува структурираните информации од Wikipedia, семантички ги аотира (со RDF) и ги објавува како јавно достапни информации. DBpedia дозволува овие податоци да се поврзуваат со други RDF податоци низ WWW, да се користат за сопствени бази на знаење, а дозволува и поставување на SPARQL прашања врз податоците со кои располага.

За да го овозможи последново, DBpedia нуди пристапна точка за пишување SPARQL прашања преку веб, преку нивниот SPARQL Endpoint, поставен на: <http://dbpedia.org/sparql>.

Напомена: Доколку некогаш имате проблеми со SPARQL Endpoint-от, можете да го користите SPARQL Explorer-от, достапен на: <http://dbpedia.org/snorql>. Доколку DBpedia е недостапна во моментот, нејзините податоци можете да ги пребарувате и преку <http://lod.openlinksw.com/sparql>.

## В) Практична задача

### I. Работа со податоци за личност, режисер

1. DBpedia своите ресурси ги означува со URI-а кои се изведени од оригиналните адреси на англиските страни на Wikipedia. Тие се добиваат на следниот начин:
  - a. Се одбира Wikipedia страна (на пр: [http://en.wikipedia.org/wiki/Woody\\_Allen](http://en.wikipedia.org/wiki/Woody_Allen))
  - b. Префиксот <http://en.wikipedia.org/wiki/> се заменува со префиксот <http://dbpedia.org/resource/>
  - c. На тој начин, се добива DBpedia URI-то за ресурсот: [http://dbpedia.org/resource/Woody\\_Allen](http://dbpedia.org/resource/Woody_Allen)
  - d. Доколку го отворите овој ресурс преку интернет прелистувач, ќе ви се отвори страната [http://dbpedia.org/page/Woody\\_Allen](http://dbpedia.org/page/Woody_Allen), на која се излистани сите RDF тројки кои DBpedia ги извлекла од Wikipedia за дадениот ресурс.
  - e. Првата половина RDF тројки се оние во кои ресурсот се среќава како Subject. Втората половина RDF тројки, во кои релациите се означени како 'is relation of', се оние во кои ресурсот се среќава како Object.

2. Разгледајте ги тројките (релациите и вредностите) кои ги има на DBpedia за ресурсот [http://dbpedia.org/resource/Woody\\_Allen](http://dbpedia.org/resource/Woody_Allen). Секоја од релациите е означена линк кој носи кон уникатно URI кое ја идентификува релацијата и на кое е поставена нејзината дефиниција.

Преку SPARQL Endpoint-от (поставен на адресата <http://dbpedia.org/sparql>), напишете SPARQL прашање со кое ќе го добиете вистинското име на Вуди Ален (dbo:birthName).

Напомена: На линкот Namespace Prefixes (поставен на SPARQL Endpoint страната) можете да ги погледнете префиксите кои може да ги користите кај ресурсите од различни извори, без да ги наведувате експлицитно во самото SPARQL прашање. На пример, ресурсот [http://dbpedia.org/resource/Woody\\_Allen](http://dbpedia.org/resource/Woody_Allen) може да се наведе како **dbr:Woody\_Allen**, а ресурсот <http://dbpedia.org/ontology/birthName> како **dbo:birthName** во SPARQL прашањата, без да ги вклучите овие два префикси експлицитно. Доколку префиксот кој сакате да го користите не е прикажан во Namespace Prefixes, дефинирајте го експлицитно.

```
select ?name
where {
dbr:Woody_Allen dbo:birthName ?name
}
```

3. Напишете SPARQL прашање со кое ќе ги добиете сите филмови на Вуди Ален (филмови кои ги режирал, филмови кои ги напишал и филмови во кои глумел).

```
select ?film
where {
?film (dbo:director|dbo:writer|dbo:starring) dbr:Woody_Allen .
}
```

4. Проширете го претходното SPARQL прашање за да ги добиете сите филмови на Вуди Ален, но подредени од најновите кон најстарите (според датумот на објавување).

```
select ?film ?releaseDate
where {
?film (dbo:director|dbo:writer|dbo:starring) dbr:Woody_Allen .
?film dbo:releaseDate ?releaseDate .
}
order by desc(?releaseDate)
```

5. Проширете го претходното SPARQL прашање за да ги добиете сите филмови на Вуди Ален, со буџет поголем од \$500.000 долари.

```
select ?film ?releaseDate ?budget
where {
?film (dbo:director|dbo:writer|dbo:starring) dbr:Woody_Allen .
?film dbo:releaseDate ?releaseDate .
?film dbo:budget ?budget .
filter (xsd:decimal(?budget) >5000000)
}
order by ?releaseDate
```

6. Напишете SPARQL прашање со кое ќе ги добиете имињата (label, на англиски) и сликите (thumbnail) на сите актери кои играле (starring) во филм на Вуди Ален. Ограничете го излезот на само 10 актери. Отстранете ги дупликатите од резултатите.

```
select distinct ?name ?pic
where {
?film (dbo:director|dbo:writer|dbo:starring) dbr:Woody_Allen .
?film dbo:starring ?starring .
?starring rdfs:label ?name .
?starring dbo:thumbnail ?pic .
filter (lang(?name) = "en")
}
limit 10
```

## II. Работа со податоци за книга

7. Отворете го ресурсот [http://dbpedia.org/resource/The\\_Little\\_Prince](http://dbpedia.org/resource/The_Little_Prince). Погледнете ги својствата (тројките) кои ги има оваа книга. Напишете SPARQL прашање со кое ќе ги добиете авторот и јазикот на книгата.

```
select ?author ?language
where {
  dbr:The_Little_Prince dbo:author ?author ;
  dbp:language ?language
}
```

8. Напишете SPARQL прашање со кое ќе го добиете апстрактот на овој ресурс. Апстрактот нека биде на англиски јазик.

```
select ?abstract
where {
  dbr:The_Little_Prince dbo:abstract ?abstract .
  filter (lang(?abstract)= "en")
}
```

9. Преку истиот ресурс (книгата „Малиот Принц“), освен името на авторот извлекете ги информациите и за датумот и местото на раѓање на авторот, како и датумот и местото на умирање.

```
select ?author ?birthDate ?birthPlace ?deathDate ?deathPlace
where {
  dbr:The_Little_Prince dbo:author ?author .
  ?author dbo:birthDate ?birthDate .
  ?author dbo:birthPlace ?birthPlace .
  ?author dbo:deathDate ?deathDate .
  ?author dbo:deathPlace ?deathPlace .
}
```

10. Преку истиот ресурс (книгата „Малиот Принц“), извлекете ги имињата и на останатите книги кои ги напишал нејзиниот автор, нивните години на издавање и сортирајте ги според годината.

```
select ?books, ?releaseDate
where {
  dbr:The_Little_Prince dbo:author ?author .
  ?books dbo:author ?author .
  optional {?books dbp:releaseDate ?releaseDate}.
  filter( dbr:The_Little_Prince != ?books)
}
```

11. Напишете SPARQL прашање за да добиете одговор во која година прв пат е издадена оваа книга, од која издавачка куќа и под кој оригинален наслов.

```
select ?releaseDate ?publisher ?titleOrig
where {
  dbr:The_Little_Prince dbp:releaseDate ?releaseDate .
  dbr:The_Little_Prince dbp:publisher ?publisher .
  dbr:The_Little_Prince dbp:titleOrig ?titleOrig .
}
```

```
}  
order by asc(?releaseDate)  
limit 1
```

### III. Работа со податоци за ресурс по слободен избор

12. Отворете Wikipedia article по ваш избор. Отворете го соодветниот DBpedia ресурс и разгледајте ги тројките кои постојат за тој ресурс. Напишете барем три SPARQL прашања по ваш избор, во кој ќе ги пробате останатите опции од аудиториски вежби.

Пример идеи:

- Музички бенд / артист:
  - прикажете ги имињата и датумите на објава на албумите;  

```
select distinct ?albumName ?releasedDate  
where{  
  ?songs dbo:artist dbr:Pink_Floyd .  
  ?songs dbo:artist ?artist .  
  ?songs dbo:album ?album .  
  ?album dbp:name ?albumName .  
  ?album dbp:released ?releasedDate .  
  filter(?artist = dbr:Pink_Floyd)  
}
```
  - прикажете ги имињата и местата на раѓање на членовите на бендот;  

```
select ?bandMemberName, ?birthPlace  
where{  
  dbr:Pink_Floyd (dbo:bandMember|dbo:formerBandMember)  
  ?bandMember .  
  ?bandMember dbp:birthName ?bandMemberName .  
  ?bandMember dbp:birthPlace ?birthPlace .  
}
```
- Личност:
  - прикажете краток опис и биографија, на англиски;  

```
select ?abstract  
where{  
  dbr:Igor_Džambazov dbo:abstract ?abstract  
  filter(lang(?abstract)="en")  
}
```
  - прикажете ги датумите и местата на раѓање / смрт;  

```
select ?birthDate (group_concat(?birthPlace; separator=", " ) as  
  ?birthPlace)  
where{  
  dbr:Igor_Džambazov dbo:birthDate ?birthDate.  
  dbr:Igor_Džambazov dbo:birthPlace ?birthPlace.  
}
```
  - прикажете ги линковите до неговиот / нејзиниот веб сајт, социјален профил,  
итн.;  

```
select (group_concat(?link; separator=", " ) as ?link)  
where{  
  dbr:Elon_Musk dbo:wikiPageExternalLink ?link  
}
```
- Локација:

- прикажете краток опис и линк кон слика од локацијата;  
**select ?abstract ?thumbnail**  
**where{**  
**dbr:Pehčevo dbo:abstract ?abstract.**  
**dbr:Pehčevo dbo:thumbnail ?thumbnail.**  
**filter (lang(?abstract)= "en")**  
**}**
- прикажете ги longitude и latitude координатите, како и надморската височина;  
**select ?long ?lat**  
**where{**  
**dbr:Pehčevo geo:long ?long.**  
**dbr:Pehčevo geo:lat ?lat.**  
**}**
- ИТН.

#### IV. Работа со податоци за потребите на конкретна апликација

13. Размислете за некоја мобилна или веб апликација која би сакале да ја развивате. Согласно темата, пробајте да напишете едно или повеќе SPARQL прашања со кои од DBpedia ќе ги извлечете сите потребни податоци за нејзино реализирање.

Пример идеи:

- најдете ги сите филмови објавени во тековната година и прикажете ги нивните имиња, краток опис, жанр, како и датумот на објава; сортирајте ги по датум на објава;  
**select ?films ?name ?abstract ?genre ?releaseDate**  
**where{**  
**?films dbo:wikiPageWikiLink dbr:Category:2023\_films.**  
**?films dbp:name ?name.**  
**?films dbo:abstract ?abstract.**  
**optional {?films dbp:genre ?genre}.**  
**optional {?films dbp:releaseDate ?releaseDate}.**  
**filter (lang(?abstract)= "en")**  
**}**
- најдете ги музичките албуми кои се објавени во тековната година и прикажете го артистот, името на албумот, жанрот, датумот на објава и линкот кон слика од албумот; сортирајте ги по датум на објава;  
**select ?album\_of\_2023\_songs ?releaseDate**  
**where{**  
**dbr:List\_of\_2023\_albums dbo:wikiPageWikiLink ?artists.**  
**?songs dbp:artist ?artists.**  
**?songs dbo:album ?album\_of\_2023\_songs.**  
**?album\_of\_2023\_songs (dbo:releaseDate|dbp:released) ?releaseDate.**  
**filter (strstarts(str(?releaseDate), "2023"))**  
**}**  
**order by ?releaseDate**
- ИТН.