

Домашно бр. 2 SPARQL

А) Домашна задача

1. Што е SPARQL?

SPARQL е прашален јазик кој се користи за испрашување на графови на знаења односно RDF графови

2. Кои типови на SPARQL прашања постојат?

Select, Describe, Ask, Construct и Update.

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

Се наоѓаат во логичка релација “И” (AND).

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

Со тоа што ќе го ставиме во OPTIONAL блок

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

Да може затоа што SPARQL ендпоинтот можеме лесно да го исправаме со користење на HTTP GET барање каде што SPARQL прашањето ќе го поставиме во query string или пак со користење на HTTP POST барање каде што SPARQL прашањето ќе го поставиме во сабото body на барањето. Дополнително може да избереме во кој формат да ни биде вратен одговорот.

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:

- Се одбира Wikipedia страна (на пр: http://en.wikipedia.org/wiki/Woody_Allen)
- Префиксот <http://en.wikipedia.org/wiki/> се заменува со префиксот <http://dbpedia.org/resource/>
- На тој начин, се добива DBpedia URI-то за ресурсот: http://dbpedia.org/resource/Woody_Allen
- Доколку го отворите овој ресурс преку интернет прелистувач, ќе ви се отвори страната http://dbpedia.org/page/Woody_Allen, на која се излистани сите RDF тројки кои DBpedia ги извлекла од Wikipedia за дадениот ресурс.
- 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. 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).

```
select ?BirthName where {dbr:Woody_Allen dbo:birthName ?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 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.

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 ?Movie

WHERE {

  {?Movie dbo:director dbr:Woody_Allen}

  UNION {?Movie dbo:author dbr:Woody_Allen}

  UNION {?Movie 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 ?Movie

WHERE {

  {?Movie dbo:director dbr:Woody_Allen}

  UNION {?Movie dbo:author dbr:Woody_Allen}

  UNION {?Movie dbo:starring dbr:Woody_Allen}

  OPTIONAL {?Movie dbp:released ?Date}

}

ORDER BY ?Date
```

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

```
SELECT ?Movie

WHERE {

  {?Movie dbo:director dbr:Woody_Allen}

  UNION {?Movie dbo:author dbr:Woody_Allen}

  UNION {?Movie dbo:starring dbr:Woody_Allen}

  ?Movie dbo:budget ?Budget

  FILTER(?Budget > 500000 )

}
```

```
OPTIONAL { ?Movie dbp:released ?Date }  
  
}
```

```
ORDER BY ?Date
```

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 ?Thumb  
  
WHERE {  
  
  { ?Movie dbo:director dbr:Woody_Allen. }  
  
  UNION { ?Movie dbo:author dbr:Woody_Allen. }  
  
  UNION { ?Movie dbo:starring dbr:Woody_Allen. }  
  
  ?Movie dbp:starring ?Actor.  
  
  ?Actor rdfs:label ?Name;  
  
          dbo:thumbnail ?Thumb  
  
}  
  
LIMIT 10
```

II. Работа со податоци за книга

7. Отворете го ресурсот 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 ?Lang  
  
WHERE {  
  
  dbr:The_Little_Prince dbo:author ?Author;  
  
          dbp:language ?Lang.  
  
}
```

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

```
SELECT  ?Abs  
  
WHERE {  
  
  dbr:The_Little_Prince dbo:abstract ?Abs.
```

```
FILTER(lang(?Abs) = 'en')
```

```
}
```

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

```
SELECT  ?Name ?BirthDate ?BirthPlace ?DeathDate ?DeathPlace
```

```
WHERE {
```

```
  dbr:The_Little_Prince dbo:author ?Author.
```

```
  ?Author dbo:birthName ?Name;
```

```
    dbo:birthDate ?BirthDate;
```

```
    dbo:birthPlace ?BirthPlace;
```

```
    dbo:deathDate ?DeathDate;
```

```
    dbo:deathPlace ?DeathPlace.
```

```
}
```

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

```
SELECT  ?BookName ?Release
```

```
WHERE {
```

```
  dbr:The_Little_Prince dbo:author ?Author.
```

```
  ?Book dbo:author ?Author.
```

```
  ?Book rdfs:label ?BookName;
```

```
    dbp:releaseDate ?Release.
```

```
}
```

```
ORDER BY ?Release
```

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  ?Release ?Publisher ?OriginalTitle
```

```
WHERE {
```

```
  dbr:The_Little_Prince dbp:releaseDate ?Release;
```

```
dbp:publisher ?Publisher;  
dbp:titleOrig ?OriginalTitle.
```

```
}
```

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;
 - прикажете ги имињата и местата на раѓање на членовите на бендот;
- Личност:
 - прикажете краток опис и биографија, на англиски;
 - прикажете ги датумите и местата на раѓање / смрт;
 - прикажете ги линковите до неговиот / нејзиниот веб сајт, социјален профил, итн.;
- Локација:
 - прикажете краток опис и линк кон слика од локацијата;
 - прикажете ги longitude и latitude координатите, како и надморската височина;
- итн.

Јас го како ресурс го избрав спортскиот клуб Реал Мадрид

- a. Прво SPARQL прашање: Најди ги имињата на сите моментални играчи на Реал Мадрид и подреди ги според нивниот број на дрес

```
SELECT ?PlayerName ?ShirtNumber  
WHERE {  
  ?Player dbp:currentclub dbr:Real_Madrid_CF;  
          rdfs:label ?PlayerName;  
          dbo:number ?ShirtNumber.  
  FILTER(lang(?PlayerName) = 'en')  
}  
ORDER BY ?ShirtNumber
```

- b. Второ SPARQL прашање Најди ги имињата на сите менаџери на Реал Мадрид низ историјата, сортирани според вкупниот број на натпревари во опфаќачки редослед.

```
SELECT ?MenagerName ?TotalCaps  
WHERE {  
  ?Menager dbo:managerClub dbr:Real_Madrid_CF.  
  ?Menager rdfs:label ?MenagerName;  
           dbp:totalcaps ?TotalCaps.  
  FILTER(lang(?MenagerName) = 'en')  
}
```

ORDER BY DESC(?TotalCaps)

- с. Трето SPARQL прашање Од овој ресурс најди ја лигата во која Реал Мадрид се натпреварува и прикажи ги имињата на нај успешниот клуб во лигата и играчот со најмногу постигнати голови

```
SELECT ?LeagueName ?BestClub ?BestPlayerName
WHERE {
  dbr:Real_Madrid_CF dbo:league ?League.
  ?League dbp:mostSuccessfulClub ?BestClub;
    dbp:topGoalscorer ?BestPlayer;
    rdfs:label ?LeagueName.
  ?BestPlayer rdfs:label ?BestPlayerName.
  FILTER(lang(?LeagueName) = 'en')
  FILTER(lang(?BestPlayerName) = 'en')
}
```

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;
- 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;
- итн.

Јас се одлучив за оваа идеја најдете ги сите филмови објавени во 2022 година и прикажете ги нивните имиња, краток опис и датумот на објава; сортирајте ги по датум на објава;

```
SELECT ?Name ?Abstract ?ReleaseDate
WHERE {
  ?Movie rdf:type dbo:Film;
    dbp:date ?ReleaseDate.
  FILTER (regex(str(?ReleaseDate), "2022" , "i"))
  ?Movie rdfs:label ?Name;
    dbo:abstract ?Abstract.
  FILTER (lang(?Name) = 'en').
  FILTER (lang(?Abstract) = 'en').
}
ORDER BY ?ReleaseDate
```