Домашно бр. 2 SPARQL

А) Домашна задача

1. IIITO e SPARQI

SPARQL e query јазик за Linked Data. Аналогно како што e SQL за Relational DB, SPARQL e за Linked Data.

2. Кои типови на SPARQL прашања постојат?

SELECT, CONSTRUCT, DESCRIBE, ASK, UPDATE

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

Resource property Resource или пак Resource property literal

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

Bo WHERE делот може да ставиме OPTIONAL{triple pattern} кој ако нема некоја вредност за него ќе биде изоставен.

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

Може бидејќи можеме да правиме GET и POST request до ендоинтите, каде што GET ги користиме за да добиеме информации од ендпоинтот, а POST за да поставиме нешто на ендпоинтот.

Б) Вовед

DBpedia (http://www.dbpedia.org) е проект кој ги извлекува структурираните информации од Wikipedia, семантички ги анотира (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.

<u>Напомена</u>: 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.

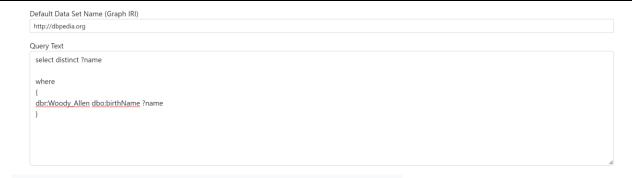
В) Практична задача

І. Работа со податоци за личност, режисер

- 1. DBреdia своите ресурси ги означува со URI-а кои се изведени од оригиналните адреси на англиските страни на Wikipedia. Тие се добиваат на следниот начин:
 - а. Се одбира Wikipedia страна (на пр: 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
 - d. 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.
 - е. 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).

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.

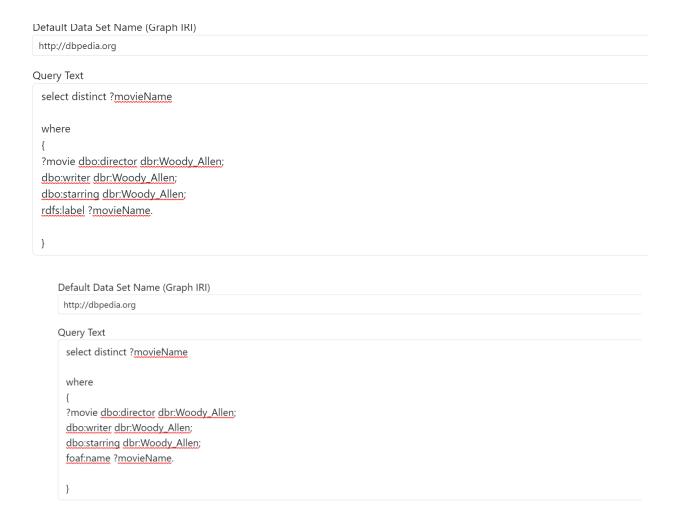


SPARQL | HTML5 table

name

"Allan Stewart Konigsberg"@en

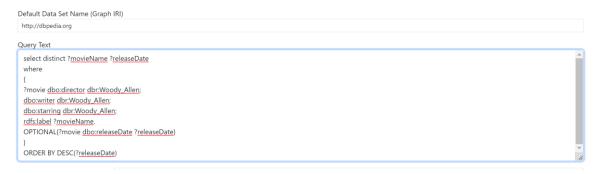
3. Напишете SPARQL прашање со кое ќе ги добиете сите филмови на Вуди Ален (филмови кои ги режирал, филмови кои ги напишал и филмови во кои глумел).



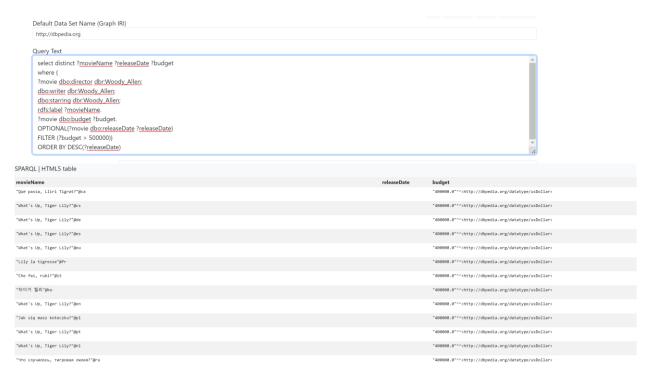
movieName	
"Què passa, Lliri Tigrat?"@ca	
"What's Up, Tiger Lily?"@cs	
"What's Up, Tiger Lily?"@de	
"What's Up, Tiger Lily?"@es	
"What's Up, Tiger Lily?"@eu	
"Lily la tigresse"@fr	
"Che fai, rubi?"@it	
"타이거 릴리"@ko	
"What's Up, Tiger Lily?"@en	
"Jak się masz koteczku?"@pl	
"What's Up, Tiger Lily?"@pt	
"What's Up, Tiger Lily?"@nl	
"Что случилось, тигровая лилия?"@ru	
"野貓爐春"@zh	
ar@"الكل يقول أجبك"	
"Tothom diu "I love you""@ca	
movieName	
"What's Up, Tiger Lily?"@en	
"Everyone Says I Love You"@en	
Everyone Says I Love You wen	
"Everything You Always Wanted to Know About Sex*"@en	
WARNE House Association Astronomy	
"(*But Were Afraid to Ask)"@en	
"Scoop"@en	
"A Midsummer Night's Sex Comedy"@en	
"Deconstructing Harry"@en	
"Hannah and Her Sisters"@en	
"Love and Death"@en	
"Stardust Memories"@en	
"Zelig"@en	
"Annie Hall"@en	
"Anything Else"@en	
CHYCHING LISE WELL	

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).

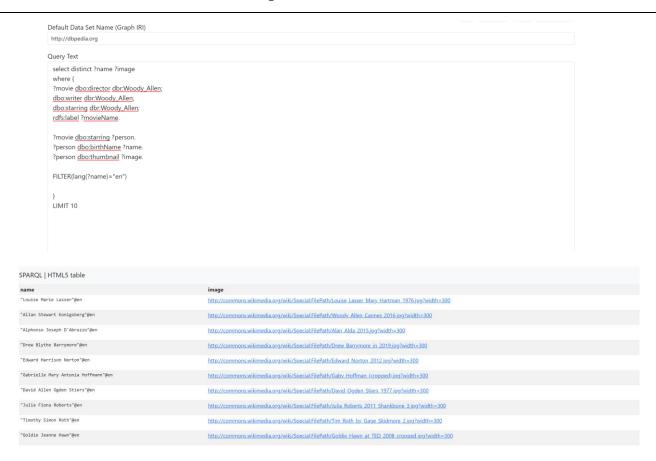
При разгледување на филмовите забележав дека немаат никаков Release Date па направив некое претпоставено query ако би го имале



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

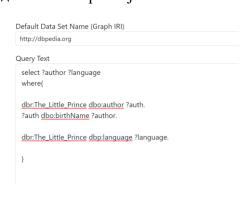


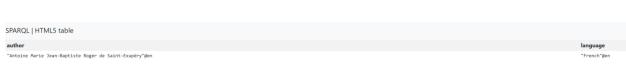
6. Напишете SPARQL прашање со кое ќе ги добиете имињата (label, на англиски) и сликите (thumbnail) на сите актери кои играле (starring) во филм на Вуди Ален. Ограничете го излезот на само 10 актери. Отстранете ги дупликатите од резултатите.



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.





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

```
select ?author ?abstract ?language where{

dbr:The_Little_Prince dbo:author ?auth;
dbo:abstract ?abstract.

?auth_dbo:birthName ?author.

dbr:The_Little_Prince_dbp:language ?language.

FILTER(lang(?abstract)="en")
}
```

("head": ("lind": [], "merm": ["sethor", "abstract", "language"]);
"results: ("distinct": false, "medered" true, "bindings": ["
'athor": "melling": "men, "value": "me Little Prince (French: Le Petit Prince, pronounced []\u00f3259 p(\u00f3255)tis
["athor": "melling": "men, "value": "me Little Prince (French: Le Petit Prince, pronounced []\u00f3259 p(\u00f3255)tis
["athor": "posterior ("special posterior ("special posterior

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 ?author ?birthDate ?passingDate ?abstract ?language where{

dbr:The Little Prince dbo:author ?auth;
dbo:abstract ?abstract.

?auth dbo:birthName ?author;
dbo:birthDate ?birthDate;
dbo:deathDate ?passingDate.

dbr:The Little Prince dbp:language ?language.

FILTER(lang(?abstract)="en")
}
```

```
birthDate passingDate a

@en 1900-06-29 1944-07-31 "
```

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.

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.



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.

Ќе направам queries за музички бенд Silk Sonic.

За артистите ќе излистам албуми кои ги имаат дати и слики од албумите.

Биографија за артистите и дата на раѓање и слики

1) Квери за име абстракт и ислика од бендот.

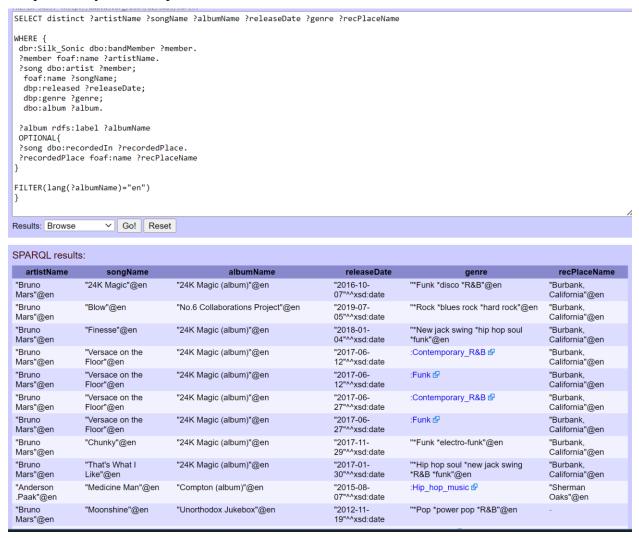


2) Квери за членовите на албумот



Powered by OpenLink Virtuoso and obpedia

3) Квери за албумите на артистите



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.

```
Query Text
 select distinct ?name ?director ?dirName ?producer ?producerName ?country ?rDate ?abstract
  where {
 ?p rdf:type dbo:Film;
 dbo:releaseDate ?rDate;
 dbp:name ?name:
 dbo:abstract ?abstract;
 dbp:director ?director.
 OPTIONAL(
 ?dir foaf:name ?director.
 OPTIONAL(
  ?p dbo:producer ?producer.
  ?producer foaf:name ?producerName.
 ?director foaf:name ?dirName
 OPTIONAL (
 ?p dbo:country ?c.
 ?c foaf:name ?country.
 FILTER (year(?rDate)=2022)
 FILTER(lang(?abstract)="en")
 ORDER BY DESC(?rDate)
```

Sparql Query кое ги враќа сите филмови излезени во 2022 година што се на dbedia нивен продуцер директор датум на излегување на филмот абстрактот и име. Сортирани по датум.

