

Leveraging ChatGpt for Natural Language Query to SPARQL on Virtuoso

--Using TheSession, Wikidata, MusicBrainz as example



Junjun Cao
Postdoctor of DDMAL
Email: junjun.cao@mail.mcgill.ca

- **Section 1** TheSession Webpage Queries vs. Using SPARQL on Virtuoso
- **Section 2** Queries Only Executable Using SPARQL on Virtuoso but not on TheSession Webpage
- **Section 3** Queries across TheSession and Wikidata
- **Section 4** Queries across TheSession, MusicBrainz and Wikidata

TheSession Webpage Queries vs. Using SPARQL on Virtuoso

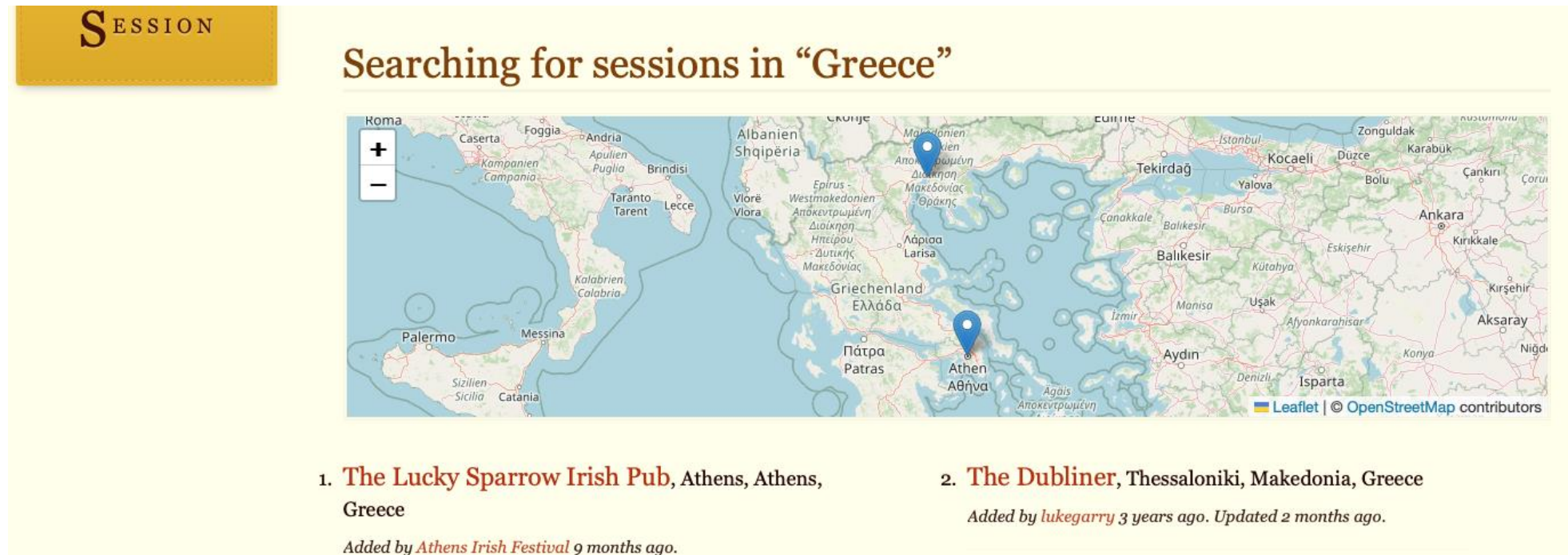
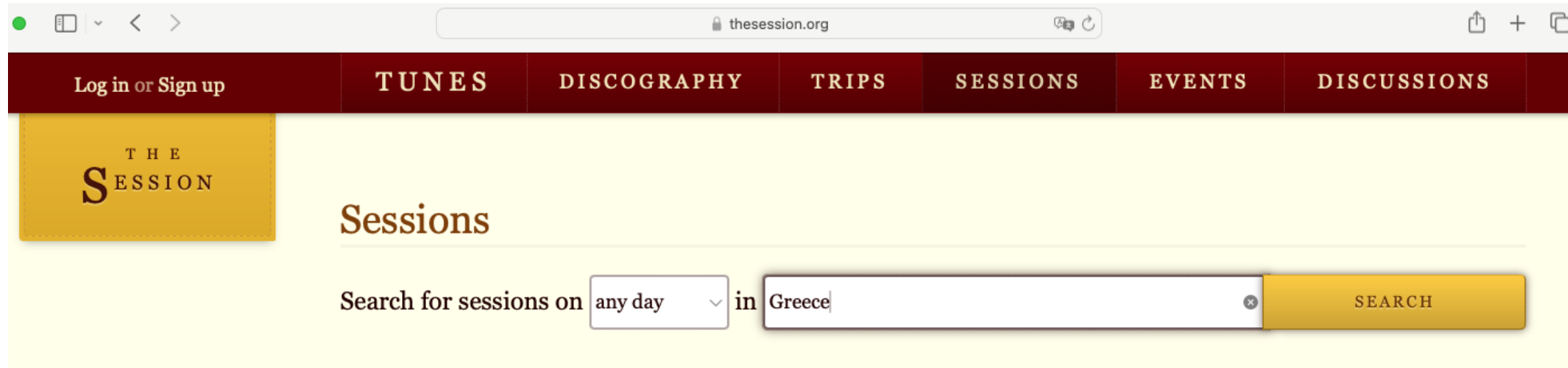


The “session”: mostly informal gatherings at which people play Irish traditional music.

- Q1-1: Find all the sessions that took place in Greece
- Q1-2: Find the tunes of genre “jig,” in “D major,” having “Malcolm” in its name
- Q1-3: Find albums containing “Tola” in its name or its artist’s name
- Q1-4: Find albums related to “Mirella”

Searching on TheSession webpage (Q1-1)

<https://thesession.org/sessions>



Prompting ChatGPT to generate SPARQL query (Q1-1)

SPARQL: SPARQL Protocol and RDF Query Language

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given a database reconciled with Wikidata, knowing `?session a
<<https://thesession.org/sessions>>`, for question “Find the sessions that took place
(wdt:P17) in Greece”, please generate a SPARQL query. **Don't forget namespace.**

Generated SPARQL by ChatGPT (Q1-1)

Find the sessions that took place in Greece

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT DISTINCT ?session

WHERE {

 ?session rdf:type <https://thesession.org/sessions> .

 ?session wdt:P17 wd:Q41 . # P17 is the 'country' property, and Q41 is the Wikidata entity for Greece

}

Virtuoso SPARQL query interface (Q1-1)

The screenshot displays the Virtuoso SPARQL query interface in a web browser. The browser's address bar shows the URL `virtuoso.staging.simssa.ca`. The interface has a light gray header with the title "SPARQL Query Editor" and navigation links "About" and "Tables". On the right side of the header are links for "Conductor", "Facet Browser", and "Permalink". Below the header, there is a section for "Extensions" with buttons for "cxml", "save to dav", "sponge", and "User: SPARQL". A label "Default Data Set Name (Graph IRI)" is positioned above a text input field. The main area is a large text editor labeled "Query Text" with a blue border and a vertical cursor. At the bottom, there is a "Results Format" dropdown menu currently set to "HTML". Below the dropdown are two buttons: "Execute Query" (blue) and "Reset" (gray).

SPARQL Query Editor About Tables Conductor Facet Browser Permalink

Extensions: cxml save to dav sponge User: SPARQL

Default Data Set Name (Graph IRI)

Query Text

Results Format HTML

Execute Query Reset

Copy SPARQL code then Execute (Q1-1)

<https://virtuoso.staging.simssa.ca/sparql>

Default Data Set Name (Graph IRI)

http://sample/thesession/reconciled

Query Text

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?session
WHERE {
  ?session rdf:type <https://thesession.org/sessions> .
  ?session wdt:P17 wd:Q41 . # P17 is the 'country' property, and Q41 is the Wikidata entity for Greece
}
```

Results Format

HTML

Execute Query

Reset

The result from Virtuoso (Q1-1)

SPARQL | HTML5 table

session

<https://thesession.org/sessions/7243>

<https://thesession.org/sessions/8020>

This is why Linked Data is associated with
“Semantic Network”?

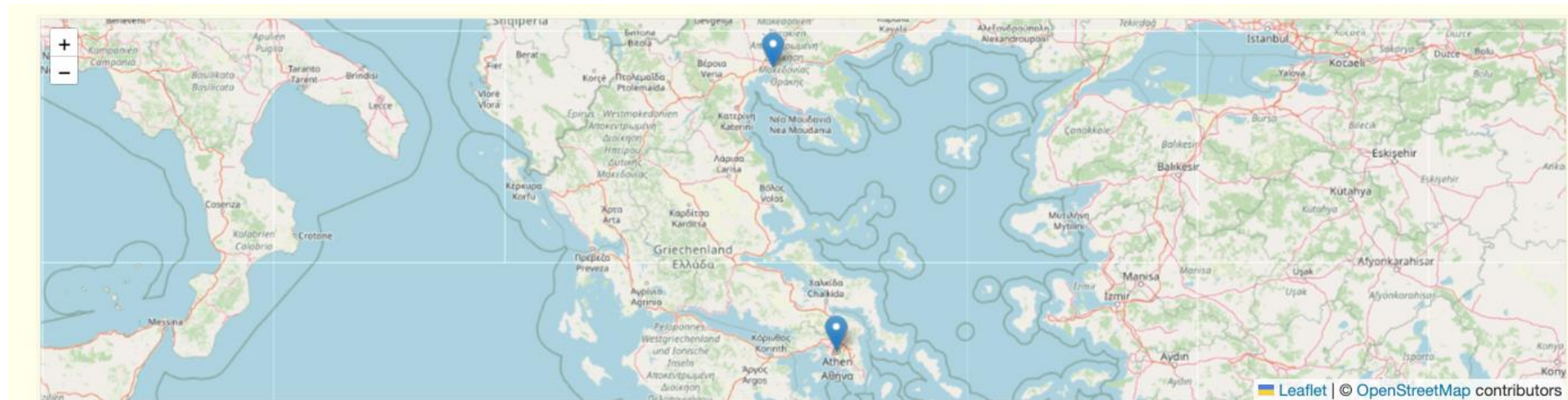
Comparison with The Session result (Q1-1)

SPARQL | HTML5 table

session

<https://thesession.org/sessions/7243>

<https://thesession.org/sessions/8020>



1. **The Lucky Sparrow Irish Pub**, Athens, Athens, Greece

Added by Athens Irish Festival 9 months ago.

2. **The Dubliner**, Thessaloniki, Makedonia, Greece

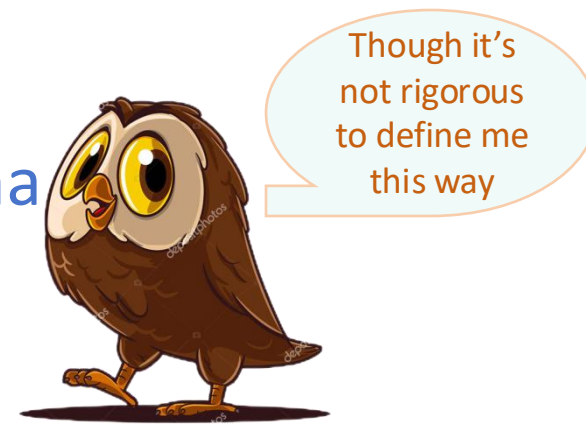
Added by lukegarry 3 years ago. Updated 1 month ago.

Search for sessions on any day in Greece

SEARCH

Prompting ChatGPT to generate SPARQL query (Q1-1)

Using Ontology (OWL, Web Ontology Language) as a schema



Enter the following in ChatGPT (**the text in red will be provided to users**):

Given that the properties of an RDF database are reconciled with Wikidata,
and ``

@prefix wd: <http://www.wikidata.org/entity/> .

@prefix wdt: <http://www.wikidata.org/prop/direct/> .

<https://thesession.org/sessions> rdf:type owl:Class .

wdt:P2561 rdfs:label "name"; rdfs:domain <https://thesession.org/sessions> .

wdt:P17 rdfs:domain <https://thesession.org/sessions>.

``

Make sure to generate a correct SPARQL for the following question:

Find all the sessions that took place in Greece

Generated SPARQL by ChatGPT (Q1-1)

Find all the sessions that took place in Greece

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
SELECT DISTINCT ?session ?name
```

```
WHERE {
```

```
  ?session rdf:type <https://thesession.org/sessions> ;
```

```
    wdt:P17 wd:Q41 ; # Assuming wd:Q41 is the Wikidata entity for Greece
```

```
    wdt:P2561 ?name .
```

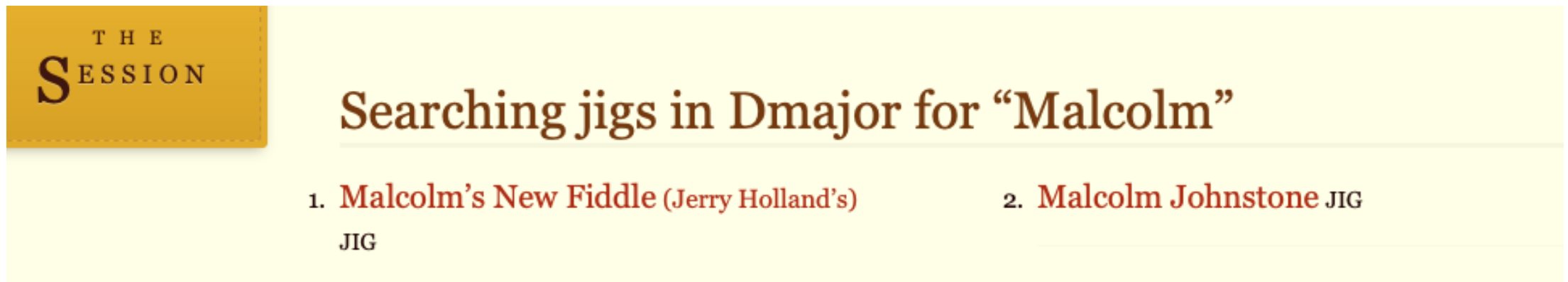
```
}
```

Searching on TheSession webpage (Q1-2)

<https://thesession.org/tunes>



The screenshot shows the top navigation bar of the TheSession.org website. The navigation bar is dark red with white text for the following links: "Log in or Sign up", "TUNES", "DISCOGRAPHY", "TRIPS", "SESSIONS", "EVENTS", and "DISCUSSIONS". Below the navigation bar is a yellow header section. On the left of this section is a logo that says "THE SESSION" in a stylized font. To the right of the logo is the word "Tunes" in a large, dark brown serif font. Below "Tunes" is a search bar with the text "Search jigs in Dmajor for Malcolm". The search bar has three dropdown menus: "jigs", "Dmajor", and "Malcolm". To the right of the search bar is a yellow button with the text "SEARCH".

- 
- The screenshot shows the search results page on TheSession.org. The page has a yellow background. On the left is a logo that says "THE SESSION" in a stylized font. To the right of the logo is the title "Searching jigs in Dmajor for 'Malcolm'" in a large, dark brown serif font. Below the title is a list of two search results:
1. **Malcolm's New Fiddle (Jerry Holland's)**
JIG
 2. **Malcolm Johnstone JIG**

Prompting ChatGPT to generate SPARQL query (Q1-2)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database reconciled with Wikidata, and

...

wd:Q170412 rdfs:type owl:Class ; rdfs:label "tune" .

wdt:P136 rdfs:type owl:DatatypeProperty ; rdfs:label "genre" .

wdt:P826 rdfs:type owl:DatatypeProperty ; rdfs:comment "This property is also used about the tonality of a tune etc" .

wdt:P2561 rdfs:label "name" .

...

Generate SPARQL for the following[Note: SPARQL be syntactically right such as `filter(contains(lcase(str(?x)))`. For string values, please add @en]:

Find the tunes of genre “jig”, in “D major”, having “Malcolm” in its name

Generated SPARQL by ChatGPT (Q1-2)

Find the tunes of genre “jig”, in “D major”, having “Malcolm” in its name

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?tune ?name

WHERE {

 ?tune rdf:type wd:Q170412 ;

 wdt:P136 "jig"@en ;

 wdt:P826 "D major"@en ;

 wdt:P2561 ?name .

 FILTER (CONTAINS(LCASE(STR(?name)), "malcolm")) }

Copy SPARQL code then Execute (Q1-2)

Query Text

```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

SELECT ?tune ?name
WHERE {
    ?tune rdf:type wd:Q170412 ;
        wdt:P136 "jig" ;
        wdt:P826 "D major" ;
        wdt:P2561 ?name .
    FILTER (CONTAINS(LCASE(STR(?name)), "malcolm"))
}
```

Results Format

HTML



Execute Query

Reset

The result from Virtuoso (Q1-2)

SPARQL | HTML5 table

tune	name
https://thesession.org/tunes/9347	"Malcolm's New Fiddle"@en
https://thesession.org/tunes/14626	"Malcolm Johnstone"@en

Comparison with The Session result (Q1-2)

SPARQL | HTML5 table

tune	name
https://thesession.org/tunes/9347	"Malcolm's New Fiddle"@en
https://thesession.org/tunes/14626	"Malcolm Johnstone"@en

THE
SESSION

Searching jigs in Dmajor for “Malcolm”

1. **Malcolm's New Fiddle** (Jerry Holland's)
JIG
2. **Malcolm Johnstone** JIG

Searching on TheSession webpage (Q1-3)

< > thesession.org

Log in or Sign up TUNES DISCOGRAPHY TRIPS SESSIONS EVENTS DISCUSSIONS

THE
SESSION

Discography

Search discography for Tola SEARCH

Searching recordings for “Tola”

1. **Setting Free** by Tola Custy And Cyril O'Donoghue
2. **Three Sunsets** by Mirella Murray And Tola Custy
3. **Dísirt Tola : Traditional Music From Clare And Dublin** by Various Artists

Search discography for Tola SEARCH

Prompting ChatGPT to generate SPARQL query (Q1-3)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database partially reconciled with Wikidata, and:

wd:Q482994 a owl:Class . wdt:P175 a owl:DatatypeProperty .

wdt:P2561.

Generate SPARQL for the following question:

[Note: (1)The value of wdt:P175 is only rdf:literal instead of a URI, so it mustn't be subject of other property.

(2)The subject of wdt:P2561 can only be instances of wd:Q482994.

(3)Use filter(contains(lcase(str(?x))). Any parts joined by UNION should be in {}.

(4)Don't forget namespace]

Question: Find albums containing "Tola" in its name or its artist's name

Generated SPARQL by ChatGPT (Q1-3)

Find albums containing “Tola” in its name or its artist’s name

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

```
SELECT ?album ?name ?artistName WHERE { {  
    ?album rdf:type wd:Q482994 ;  
        wdt:P2561 ?name . FILTER(contains(lcase(str(?name)), "tola"))  
} UNION {  
    ?album rdf:type wd:Q482994 ;  
        wdt:P175 ?artistName .  
    FILTER(contains(lcase(str(?artistName)), "tola"))}}
```

Copy SPARQL code then Execute (Q1-3)

Query Text

```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
SELECT ?album ?name ?artistName WHERE { {
    ?album rdf:type wd:Q482994 ;
            wdt:P2561 ?name . FILTER(contains(lcase(str(?name)), "tola"))
} UNION {
    ?album rdf:type wd:Q482994 ;
            wdt:P175 ?artistName .
    FILTER(contains(lcase(str(?artistName)), "tola"))}}
```

Results Format

HTML

Execute Query

Reset

The result from Virtuoso (Q1-3)

SPARQL | HTML5 table

album	name	artistName
https://thesession.org/recordings/4016	"Dísirt Tola : Traditional Music From Clare And Dublin"@en	
https://thesession.org/recordings/448		"Tola Custy And Cyril O'Donoghue"@en
https://thesession.org/recordings/1525		"Mirella Murray And Tola Custy"@en

Comparison with The Session result (Q1-3)

SPARQL | HTML5 table

album	name	artistName
https://thesession.org/recordings/4016	"Dísirt Tola : Traditional Music From Clare And Dublin"@en	
https://thesession.org/recordings/448		"Tola Custy And Cyril O'Donoghue"@en
https://thesession.org/recordings/1525		"Mirella Murray And Tola Custy"@en

Searching recordings for “Tola”

1. **Setting Free** by Tola Custy And Cyril O'Donoghue
2. **Three Sunsets** by Mirella Murray And Tola Custy
3. **Dísirt Tola : Traditional Music From Clare And Dublin** by Various Artists

Search discography for

Tola



SEARCH

Searching on TheSession webpage (Q1-4)



The screenshot shows the top navigation bar of TheSession.org with links for Log in or Sign up, TUNES, DISCOGRAPHY, TRIPS, SESSIONS, EVENTS, and DISCUSSIONS. The Discography section is active, displaying a search bar with the text "Search discography for" and a search input field containing "Mirella". A yellow "SEARCH" button is located to the right of the input field. The TheSession logo is visible in the top left corner.

Log in or Sign up

TUNES DISCOGRAPHY TRIPS SESSIONS EVENTS DISCUSSIONS

THE
SESSION

Discography

Search discography for Mirella

SEARCH

Searching recordings for “Mirella”

1. **Three Sunsets** by Mirella Murray And Tola Custy

2. **Celtic Breeze** by Celtic Breeze

Prompting ChatGPT to generate SPARQL query (Q1-4)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database partially reconciled with Wikidata, and:

...

wd:Q482994 rdf:type owl:Class .

ex:Shape a sh:NodeShape ; sh:targetClass wd:Q482994 ; sh:not [sh:property [sh:path rdfs:label]].

...

Generate SPARQL for the following question:

[Caution: Use filter(contains(lcase(str(?x))); “related to” means some of one’s attributes contains “Mirella”]

Find albums related to “Mirella”

Generated SPARQL by ChatGPT (Q1-4)

Find albums related to “Mirella”

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

```
SELECT DISTINCT ?album ?attribute ?value WHERE {  
    ?album rdfs:type wd:Q482994 .  
    ?album ?attribute ?value .  
    FILTER (contains(lcase(str(?value)), "mirella")).  
}
```

Copy SPARQL code then Execute (Q1-4)

Query Text

```
PREFIX wd: <http://www.wikidata.org/entity/>
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
PREFIX rdf: <http://www.w3.org/1999/02/22-rdf-syntax-ns#>
PREFIX owl: <http://www.w3.org/2002/07/owl#>
PREFIX sh: <http://www.w3.org/ns/shacl#>

SELECT DISTINCT ?album ?attribute ?value WHERE {
  ?album rdfs:type wd:Q482994 .
  ?album ?attribute ?value .
  FILTER (contains(lcase(str(?value)), "mirella")).
}
```

Results Format

HTML

Execute Query

Reset

The result from Virtuoso (Q1-4)

SPARQL | HTML5 table

album	attribute	value
https://thesession.org/recordings/3355	http://www.wikidata.org/prop/direct/P1625	"Mirella Murray's"@en
https://thesession.org/recordings/1525	http://www.wikidata.org/prop/direct/P175	"Mirella Murray And Tola Custy"@en

Comparison with The Session result(Q1-4)

SPARQL | HTML5 table

album	attribute	value
https://thesession.org/recordings/3355	http://www.wikidata.org/prop/direct/P1625	"Mirella Murray's"@en
https://thesession.org/recordings/1525	http://www.wikidata.org/prop/direct/P175	"Mirella Murray And Tola Custy"@en

Searching recordings for “Mirella”

1. **Three Sunsets** by Mirella Murray And Tola Custy

2. **Celtic Breeze** by Celtic Breeze

A comprehensive context for several questions (from Q1-1 to Q1-4)

- We can merge the different contexts into a single prompt for ChatGPT to generate SPARQL queries for various questions, such as those previously mentioned:
- Q1-1: Find all the sessions that took place in Greece
- Q1-2: Find the tunes of genre “jig,” in “D major,” having “Malcolm” in its name
- Q1-3: Find albums containing “Tola” in its name or its artist’s name
- Q1-4: Find albums related to “Mirella”

The entire context for generating various SPARQL queries (Sec1)

Given that the entities of an RDF database are reconciled with Wikidata, and

```
`` @prefix wd: <http://www.wikidata.org/entity/> .
@prefix wdt: <http://www.wikidata.org/prop/direct/> .
@prefix thesession: <https://thesession.org/> .
thesession:sessions rdf:type owl:Class . wdt:P17
wdt:P136 a owl:DatatypeProperty ; rdfs:label "genre" .
wdt:P175 a owl:DatatypeProperty .
wdt:P826 a owl:DatatypeProperty ; rdfs:label "mode" ;
    rdfs:comment "also used for the tonality of a tune" .
wdt:P2561 rdfs:label "name" .
    rdfs:domain thesession:sessions .
wd:Q170412 a owl:Class ;
    rdfs:label "tune" .
wd:Q482994 a owl:Class .
ex:Shape a sh:NodeShape ; sh:targetClass wd:Q482994 ; sh:not [ sh:property [ sh:path rdfs:label, wdt:P1476 ]].
``
```

Note:

1. In terms of SPARQL generation, it must be syntactically right such as:
 - (1) `filter(contains(lcase(str(?x)))`--Don't forget str() function
 - (2) Any parts joined by `UNION` should be in {}
 - (3) All the namespace prefixes should be defined for each SPARQL query
2. SPARQL generation should strictly adhere to the given context associated with OWL or SHACL, etc.
For example, distinguish between ObjectProperty and DataProperty!
As above, pay attention: wdt:P136, wdt:P175 and wdt:P826 are DataProperties! So, for example, the value of wdt:P175 are only strings in our context!
3. "related to" means some of one's attributes contains "Mirella"
Make sure to generate correct SPARQL for the following
4. For string values, please add @en
question:

2. Queries only Executable Using
SPARQL on Virtuoso but not on
TheSession Webpage



- Q2-1: Find a session named "[Hurley's Irish Pub](#)"
- Q2-2: What genres appear in the recording whose artist is "Adam Agee & Jon Sousa"?
- Q2-3: Find the session that took place within a 20-kilometer radius centered on the session "[Hurley's Irish Pub](#)"
- Q2-4: 找到在希腊的演奏会 (mandarin / Chinese, translated as "Find sessions in Greece")

Log in or Sign up

TUNES

DISCOGRAPHY

TRIPS

SESSIONS

EVENTS

DISCUSSIONS

THE
SESSION

Hurley's Irish Pub

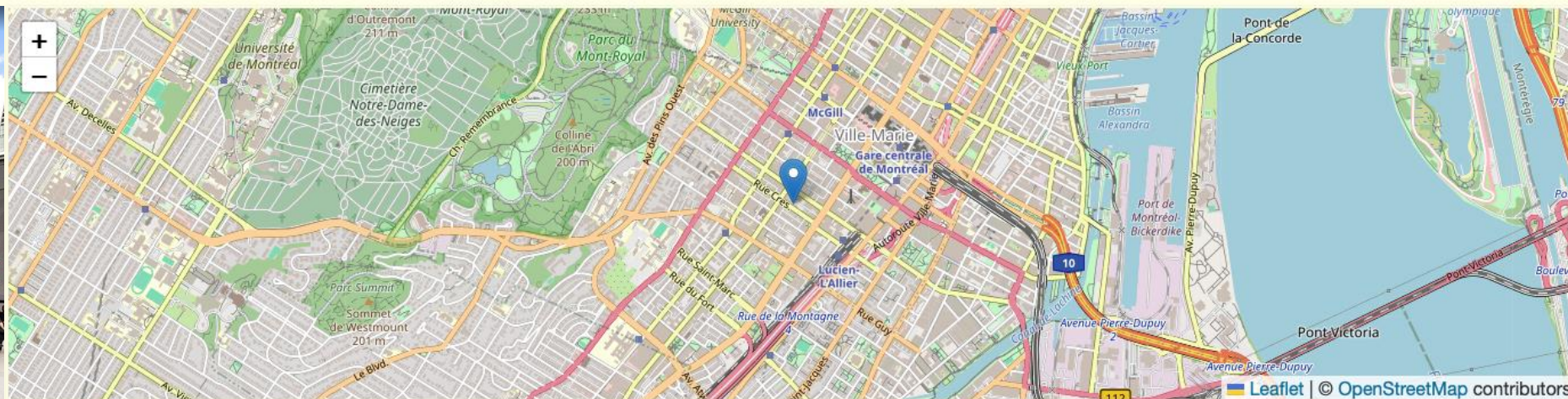


SHARE

1225 Crescent Street, Montreal, Quebec, Canada

Added by [Loa](#) 21 years ago.

Last updated [with a comment](#) by [GW](#) 9 months ago.



nearby members



nearby sessions



nearby events

Schedule: Saturday, Tuesday.

Telephone: 514.861.4111

Website: <http://www.hurleysirishpub.com/>

Latest: <https://www.siamsa.org/en/sessions>

➡ Posted by [GW](#) 9 months ago.

Hurley's Irish Pub

Prompting ChatGPT to generate SPARQL query (Q2-1)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database with properties(wdt:P2561) reconciled with Wikidata, generate SPARQL for the following (Use wdt:P2561 instead of rdfs:label. For string values, please add @en):

Find a session named "Hurley's Irish Pub"

Generated SPARQL by ChatGPT (Q2-1)

Find a session named "Hurley's Irish Pub"

PREFIX wdt: <<http://www.wikidata.org/prop/direct/>>

SELECT ?session

WHERE {

?session wdt:P2561 "Hurley's Irish Pub"@en .

}	Subject	predicate	object
		(property)	(property value)

The result from Virtuoso (Q2-1)

SPARQL | HTML5 table

session

<https://thesession.org/sessions/411>

<https://thesession.org/sessions/411>

Prompting ChatGPT to generate SPARQL query (Q2-2)

Enter the following in ChatGPT (the text in red will be provided to users):

Given an RDF database reconciled with Wikidata(such as Q482994)
and...

@PREFIX thesession: <https://thesession.org/> .

wd:Q170412 rdfs:label "tune" .

wd:Q482994 rdfs:label "album" .

wd:Q483394 rdfs:label "genre" .

wdt:P175 rdf:type owl:DatatypeProperty . #even if it's an

ObjectProperty in Wikidata, here we take it as DataProperty

thesession:tunes rdfs:domain wd:Q482994 ;

rdfs:range wd:Q170412 .

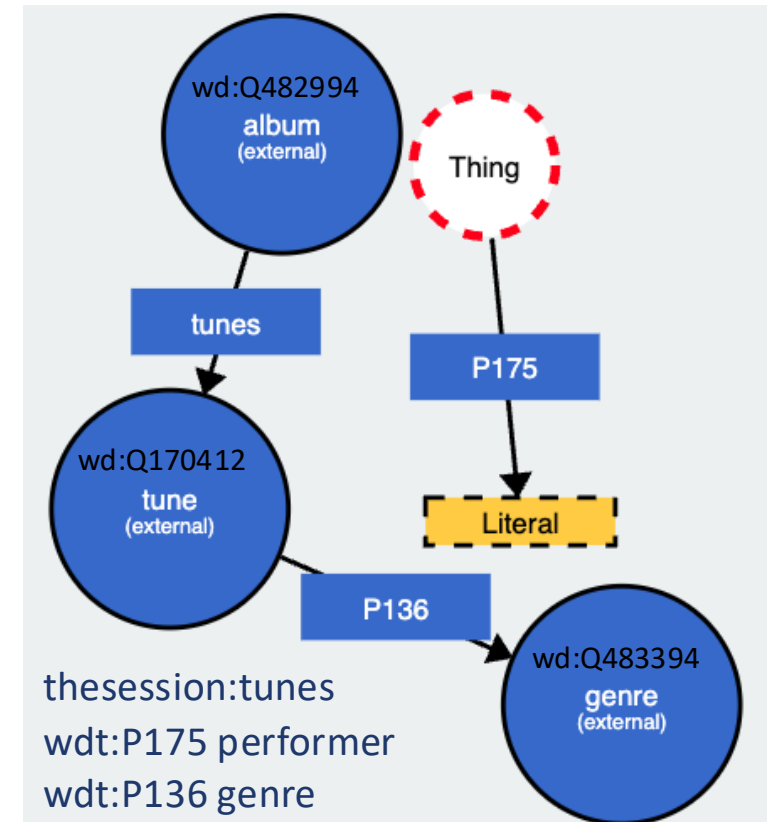
wdt:P136 rdfs:domain wd:Q170412 ;

rdfs:range wd:Q483394

Generate SPARQL for the following[note: No need to return

the label of an entity. For string property values, please add @en]:

What genres appear in the album whose artist is "Adam Agee & Jon Sousa"?



Generated SPARQL by ChatGPT (Q2-2)

What genres appear in the album whose artist is "Adam Agee & Jon Sousa"?

PREFIX thesession: <<https://thesession.org/>>

PREFIX wd: <<http://www.wikidata.org/entity/>>

PREFIX wdt: <<http://www.wikidata.org/prop/direct/>>

SELECT DISTINCT ?genre

WHERE {

Match the album of the artist "Adam Agee & Jon Sousa"

?album wdt:P175 "Adam Agee & Jon Sousa"@en .

Find all tunes in the matched albums

?album thesession:tunes ?tune .

Find genres of the matched tunes

?tune wdt:P136 ?genre . }

The result from Virtuoso (Q2-2)

SPARQL | HTML5 table

genre

<http://www.wikidata.org/entity/Q118120>

"reel"

<http://www.wikidata.org/entity/Q1079270>

"jig"

<http://www.wikidata.org/entity/Q211025>

"march"

<http://www.wikidata.org/entity/Q8701407>

"waltz"

Prompting ChatGPT to generate SPARQL query (Q2-3)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database reconciled with Wikidata and

...

wdt:P625 rdfs:domain <https://thesession.org/sessions> .

wdt:P2561 rdfs:label "sessionName" ; rdfs:domain <https://thesession.org/sessions> .

...

Use bif:st_distance(). Remember to confine the type for entities .

Generate SPARQL for the following (For string property values, please add @en):

Find other sessions that took place within a 20-kilometer radius centered on the session "Hurley's
Irish Pub"

Generated SPARQL by ChatGPT (Q2-3)

Find other sessions that took place within a 20-kilometer radius centered on the session "Hurley's Irish Pub"

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?session ?sessionName ?location ?distance

WHERE {

Find the location of "Hurley's Irish Pub"

?centerSession wdt:P2561 "Hurley's Irish Pub"@en ;

wdt:P625 ?centerLocation .

Find other sessions and their locations

?session a <https://thesession.org/sessions> ;

wdt:P2561 ?sessionName ;

wdt:P625 ?location .

Calculate distance between "Hurley's Irish Pub" and other sessions

BIND(bif:st_distance(?centerLocation, ?location) AS ?distance)

Filter sessions within a 20-kilometer radius

FILTER(?distance <= 20)} ORDER BY ?distance

The result from Virtuoso (Q2-3)

SPARQL | HTML5 table

session	sessionName	location	distance
https://thesession.org/sessions/411	"Hurley's Irish Pub"@en	"POINT(45.4969101 -73.57475281)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	0.0
https://thesession.org/sessions/410	"McKibbin's"@en	"POINT(45.49675751 -73.57733154)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	0.286621
https://thesession.org/sessions/8805	"Lord William Pub"@en	"POINT(45.48695374 -73.5670166)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	0.91493
https://thesession.org/sessions/2779	"Les Îles En Ville"@en	"POINT(45.45341873 -73.56800842)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	1.55896
https://thesession.org/sessions/7202	"La Petite Marche"@en	"POINT(45.52675629 -73.58866119)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	1.80777
https://thesession.org/sessions/2780	"Aux Vices & Versa"@en	"POINT(45.53131866 -73.61063385)"^^<http://www.openlinksw.com/schemas/virtrdf#Geometry>	4.13126

geographical coordinates

Log in or Sign up

TUNES

DISCOGRAPHY

TRIPS

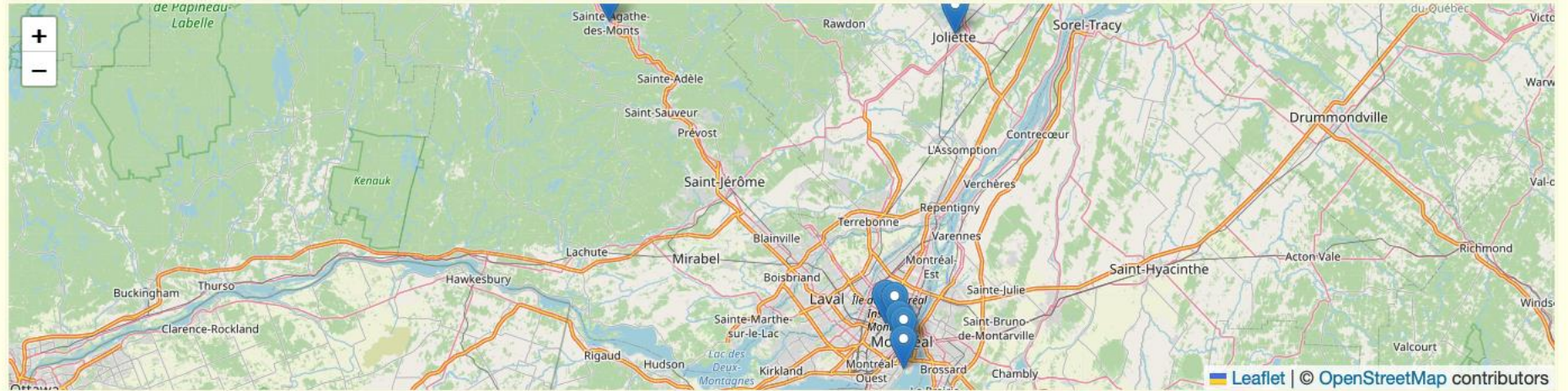
SESSIONS

EVENTS

DISCUSSIONS

THE
SESSION

Sessions near Montreal, Quebec



📍 nearby members

📍 nearby sessions

📍 nearby events

A similar result from
TheSession

<https://thesession.org/sessions/411>

Hurley's Irish Pub, Montreal, Quebec, Canada

Added by *Loa* 21 years ago. Updated 9 months ago.

Aux Vices & Versa, Montreal, Quebec, Canada

Added by *cpbmtl* 14 years ago. Updated 2 years ago.

McKibbin's, Montreal, Quebec, Canada

Added by *Loa* 21 years ago. Updated 2 months ago.

Les Îles En Ville, Montreal, Quebec, Canada

Added by *cpbmtl* 14 years ago. Updated 2 years ago.

Lord William Pub, Montreal, Quebec, Canada

Added by *Jeremy* 6 months ago.

Brasserie Artisanale Albion, Joliette, Quebec, Canada

Added by *Dana Whittle* 11 years ago. Updated 6 years ago.

La Petite Marche, Montreal, Quebec, Canada

Added by *Patrice Bouchertère* 3 years ago.

Le P'tit Refrain Du Nord, Sainte-Agathe-des-Monts, Quebec, Canada

Added by *Geneviève Poulin* 2 years ago.

Prompting ChatGPT to generate SPARQL query (Q2-4)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF database reconciled with Wikidata, and

```

@prefix wd: <http://www.wikidata.org/entity/> .

@prefix wdt: <http://www.wikidata.org/prop/direct/> .

<https://thesession.org/sessions> a owl:Class .

wdt:P2561 rdfs:label "name";

          rdfs:domain <https://thesession.org/sessions> .

wdt:P17 rdfs:domain <https://thesession.org/sessions>.

``` (use wdt:P2561 instead of rdfs:label)

Make sure to generate a correct SPARQL for the following question:

找到在希腊的演奏会 (mandarin / Chinese, translated as “Find sessions in Greece”)

Generated SPARQL by ChatGPT (Q2-4)

找到在希腊的演奏会 (mandarin / Chinese, translated as “Find sessions in Greece”)

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
SELECT ?session ?name
```

```
WHERE {
```

```
    ?session a <https://thesession.org/sessions> ;
```

```
        wdt:P17 wd:Q41 ; # Q41 is the Wikidata entity ID for Greece
```

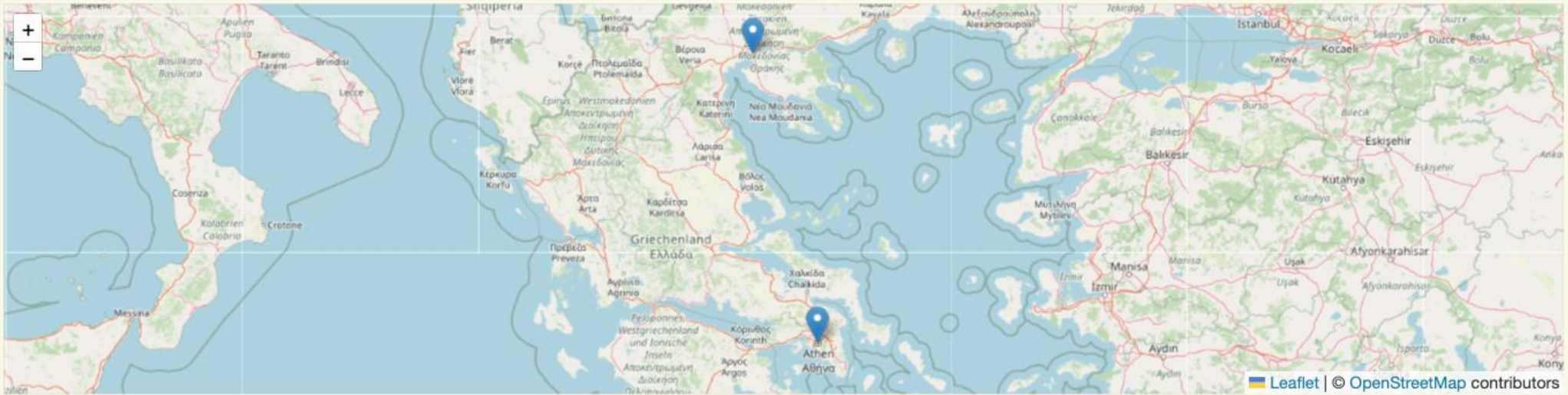
```
        wdt:P2561 ?name .
```

```
}
```


Comparison with The Session result (Q2-4)

SPARQL | HTML5 table

| session | name |
|---|-------------------------------|
| https://thesession.org/sessions/7243 | "The Dubliner" |
| https://thesession.org/sessions/8020 | "The Lucky Sparrow Irish Pub" |



The map displays Greece with two blue location pins. One pin is in Athens, labeled 'The Lucky Sparrow Irish Pub', and the other is in Thessaloniki, labeled 'The Dubliner'. The map includes various Greek cities and regions, with a legend at the bottom right indicating 'Leaflet | © OpenStreetMap contributors'.

1. **The Lucky Sparrow Irish Pub**, Athens, Athens, Greece

Added by *Athens Irish Festival* 9 months ago.

2. **The Dubliner**, Thessaloniki, Makedonia, Greece

Added by *lukegarrry* 3 years ago. Updated 1 month ago.

Search for sessions on

any day

in

Greece

SEARCH

A comprehensive context for several questions (from Q2-1 to Q2-4)

- We can merge the different contexts into a single prompt for ChatGPT to generate SPARQL queries for various questions, such as those previously mentioned:
- Q2-1: Find a session named "Hurley's Irish Pub"
- Q2-2: What genres appear in the recording whose artist is "Adam Agee & Jon Sousa"?
- Q2-3: Find the session that took place within a 20-kilometer radius centered on the session "Hurley's Irish Pub"
- Q2-4: 找到在希腊的演奏会 (mandarin / Chinese, translated as Find sessions in Greece)

The entire context for generating various SPARQL queries (Sec2)

Given an RDF database with properties reconciled with Wikidata, and

@PREFIX thesession: <https://thesession.org/> .

@PREFIX wd: <http://www.wikidata.org/entity/> .

@PREFIX wdt: <http://www.wikidata.org/prop/direct/> .

thesession:sessions a owl:Class .

thesession:tunes rdfs:domain wd:Q482994 ;

rdfs:range wd:Q170412 ;

rdfs:comment "For albums have tunes" .

wd:Q170412 rdfs:label "tune" .

wd:Q482994 rdfs:label "album" .

wd:Q483394 rdfs:label "genre" .

wdt:P17 rdfs:domain thesession:sessions .

wdt:P175 rdf:type owl:DatatypeProperty .

wdt:P136 rdfs:domain wd:Q170412 ;

rdfs:range wd:Q483394 .

wdt:P625 rdfs:domain thesession:sessions .

wdt:P2561 rdfs:label "sessionName" ; rdfs:domain thesession:sessions .

Before generating SPARQL, note: 1. No need to return the label of an entity

2. Use `bif:st_distance()`, the unit in which is kilometer and `bif:` is not a namespace prefix

3. Remember to confine the type for entities

4. Don't forget the namespace prefix for each SPARQL query!!

5. Don't use property that's not provided above

6 .Albums don't relate genres directly, but Albums have tunes which relate genres

7. Even if `wdt:P175` is an `ObjectProperty` in Wikidata, here we take it as `DataProperty`

8. For string property values, please add `@en`

Generate correct SPARQL for the following question:

3. Queries across TheSession and Wikidata

Some information is not available solely through TheSession. Given that we have already done reconciliation with Wikidata, we can extend our query to external data sources (e.g., Wikidata) for more specific questions such as...



- Q3-1: Which sessions took place in Eastern Europe?
- Q3-2: From TheSession, please find recordings by Altan, which were known to be released between 2007 and 2012
- Q3-3: Find in TheSession performers who are Canadians. And find the recordings they performed in TheSession

The information about the green part comes from Wikidata.

Prompting ChatGPT to generate SPARQL query (Q3-1)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF graph <http://sample/thesession/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible, so regarding SPARQL queries across different databases, we use `graph <...>` and `service <...>`. and Given:

@prefix thesession: <https://thesession.org/>. thesession:sessions a owl:Class.

wdt:P17 rdfs:domain <https://thesession.org/sessions>.

wdt:P2561 rdfs:domain <https://thesession.org/sessions>.

--for graph <http://sample/thesession/reconciled>

wdt:P361 rdfs:domain <ex:country>. wd:Q27468 rdfs:label "Eastern Europe".

--for service <https://query.wikidata.org/sparql>

Make sure to generate a correct SPARQL for the following question (Note: Use the properties or classes as above primarily; traverse wikidata before local graph):

Which sessions took place in Eastern Europe?

Generated SPARQL by ChatGPT (Q3-1)

Which sessions took place in Eastern Europe?

PREFIX thesession: <https://thesession.org/>

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

SELECT ?session ?location WHERE {

SERVICE <https://query.wikidata.org/sparql> {

?location wdt:P361 wd:Q27468 . # Q27468 is the Wikidata item for Eastern Europe

}

GRAPH <http://sample/thesession/reconciled> {

?session wdt:P17 ?location .

?session a thesession:sessions . }}



The result from Virtuoso (Q3-1)

| session | location | https://thesession.org/sessions/6721 | http://www.wikidata.org/entity/Q159 |
|---|---|---|---|
| https://thesession.org/sessions/1218 | http://www.wikidata.org/entity/Q36 | https://thesession.org/sessions/6918 | http://www.wikidata.org/entity/Q36 |
| https://thesession.org/sessions/1896 | http://www.wikidata.org/entity/Q213 | https://thesession.org/sessions/7549 | http://www.wikidata.org/entity/Q36 |
| https://thesession.org/sessions/1915 | http://www.wikidata.org/entity/Q212 | https://thesession.org/sessions/8013 | http://www.wikidata.org/entity/Q213 |
| https://thesession.org/sessions/2206 | http://www.wikidata.org/entity/Q159 | https://thesession.org/sessions/8014 | http://www.wikidata.org/entity/Q213 |
| https://thesession.org/sessions/2521 | http://www.wikidata.org/entity/Q36 | https://thesession.org/sessions/8731 | http://www.wikidata.org/entity/Q159 |
| https://thesession.org/sessions/2760 | http://www.wikidata.org/entity/Q213 | https://thesession.org/sessions/8775 | http://www.wikidata.org/entity/Q219 |
| https://thesession.org/sessions/3241 | http://www.wikidata.org/entity/Q36 | https://thesession.org/sessions/8806 | http://www.wikidata.org/entity/Q213 |
| https://thesession.org/sessions/3452 | http://www.wikidata.org/entity/Q213 | https://thesession.org/sessions/8823 | http://www.wikidata.org/entity/Q159 |
| https://thesession.org/sessions/3631 | http://www.wikidata.org/entity/Q184 | https://thesession.org/sessions/8824 | http://www.wikidata.org/entity/Q159 |
| https://thesession.org/sessions/5941 | http://www.wikidata.org/entity/Q36 | https://thesession.org/sessions/8905 | http://www.wikidata.org/entity/Q213 |

Q36: Poland, Q159: Russia, Q184: Belarus, Q213: Czech Republic, Q219: Bulgaria

Prompting ChatGPT to generate SPARQL query (Q3-2)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF graph <http://sample/thesession/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible, so regarding SPARQL queries across different databases, we use `graph <...>` and `service <...>`. and Given:

For graph <http://sample/thesession/reconciled> :

wd:Q482994 a owl:Class. wdt:P175 rdfs:domain wd:Q482994; a owl:DataProperty.

For service <https://query.wikidata.org/sparql> :

wdt:P577 rdfs:range xsd:dateTime.

Make sure to generate a correct SPARQL for the following question

(Note: Use the properties or classes as above primarily; Even if on Wikidata, wdt:P175 is an ObjectProperty, we see it as a DataProperty here; Use such as `?sessionRecording wdt:P2888 ?wikidataRecording .`; For string property values, please add @en):

From TheSession, please find recordings by Altan, which were known to be released between 2007 and 2012

Generated SPARQL by ChatGPT (Q3-2)

From TheSession, please find recordings by Altan, which were known to be released between 2007 and 2012

PREFIX wdt: <http://www.wikidata.org/prop/direct/>

PREFIX wd: <http://www.wikidata.org/entity/>

PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX owl: <http://www.w3.org/2002/07/owl#>

PREFIX xsd: <http://www.w3.org/2001/XMLSchema#>

```
SELECT ?sessionRecording ?wikidataRecording ?releaseDate WHERE {  
  GRAPH <http://sample/thesession/reconciled> { ?sessionRecording a wd:Q482994 ;  
                                                    wdt:P175 "Altan"@en .  
                                                    ?sessionRecording wdt:P2888 ?wikidataRecording .}  
  SERVICE <https://query.wikidata.org/sparql> { ?wikidataRecording wdt:P577 ?releaseDate .  
  FILTER (?releaseDate >= "2007-01-01T00:00:00Z"^^xsd:dateTime && ?releaseDate <= "2012-12-31T23:59:59Z"^^xsd:dateTime) }}
```

The result from Virtuoso (Q3-2)

SPARQL | HTML5 table (faceted browsing links)

| sessionRecording | wikidataRecording | releaseDate |
|---|---|----------------------|
| https://thesession.org/recordings/3560 | http://www.wikidata.org/entity/Q4632227 | 2010-01-01T00:00:00Z |
| https://thesession.org/recordings/4142 | http://www.wikidata.org/entity/Q5567392 | 2012-01-01T00:00:00Z |

Prompting ChatGPT to generate SPARQL query (Q3-3-0)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF graph <http://sample/thesession/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible, so regarding SPARQL queries across different databases, we use `graph <...>` and `service <...>`. and Given:

The value of wdt:P175 is performer

--for graph <http://sample/thesession/reconciled>

wdt:P175 performer

wdt:P27

wdt:P27 country of citizenship

--for service <https://query.wikidata.org/sparql>

Make sure to generate a correct SPARQL for the following question:

Find in TheSession performers who are Canadians ...

Expected SPARQL (Q3-3-0)

Find in TheSession performers who are Canadians

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
SELECT ?performer
```

```
WHERE {
```

```
  GRAPH <http://sample/thesession/reconciled> {
```

```
    ?x a wd:Q482994 ;
```

```
        wdt:P175 ?performer .
```

```
  FILTER isIRI(?performer)
```

```
}
```

```
SERVICE <https://query.wikidata.org/sparql> {
```

```
  ?performer wdt:P27 wd:Q16
```

```
}}
```

SPARQL | HTML5 table

performer

<http://www.wikidata.org/entity/Q725953>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q7417356>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q7417356>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q645562>

<http://www.wikidata.org/entity/Q4960771>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q725953>

<http://www.wikidata.org/entity/Q7417356>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q6183761>

<http://www.wikidata.org/entity/Q4960771>

<http://www.wikidata.org/entity/Q6968206>

<http://www.wikidata.org/entity/Q6183761>

Prompting ChatGPT to generate SPARQL query (Q3-3)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given an RDF graph <http://sample/thesession/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible, so regarding SPARQL queries across different databases, we use

(1) `graph <http://sample/thesession/reconciled>` with:

wd:Q482994 rdfs:label "recording". wdt:P175 # the value of it can be either rdfs:Literal or URI

(2) `service <https://query.wikidata.org/sparql>` with:

wdt:P27

[Note: Use isIRI(?performer); Don't use `rdfs:label` as a predicate in SPARQL anyway!]

Make sure to generate a correct SPARQL for the following question:

Find in TheSession performers who **are Canadians**. And find the recordings they performed in TheSession

Generated SPARQL by ChatGPT (Q3-3)

Find in TheSession performers who are Canadians. And find the recordings they performed in TheSession

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
SELECT ?performer ?recording WHERE {
```

```
  GRAPH <http://sample/thesession/reconciled> {
```

```
    ?recording wdt:P175 ?performer .
```

```
  }
```

```
SERVICE <https://query.wikidata.org/sparql> {
```

```
  ?performer wdt:P27 wd:Q16 .
```

```
}
```

```
FILTER(isIRI(?performer)) }
```

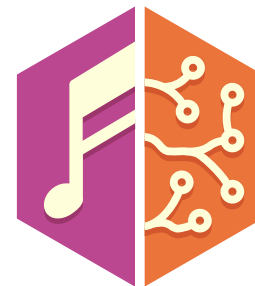


Note: It probably takes a long time to get the result from running the code on localhost; sometimes, it reports timeout. Recently, there is timeout on staging

The result from Virtuoso (Q3-3)

| performer | recording |
|---|---|
| http://www.wikidata.org/entity/Q725953 | https://thesession.org/recordings/5709 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/662 |
| http://www.wikidata.org/entity/Q7417356 | https://thesession.org/recordings/6621 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/6747 |
| http://www.wikidata.org/entity/Q7417356 | https://thesession.org/recordings/7463 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/7677 |
| http://www.wikidata.org/entity/Q645562 | https://thesession.org/recordings/3369 |
| http://www.wikidata.org/entity/Q4960771 | https://thesession.org/recordings/3617 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/4075 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/4190 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/427 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/430 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/434 |
| http://www.wikidata.org/entity/Q6968206 | https://thesession.org/recordings/5059 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/560 |
| http://www.wikidata.org/entity/Q725953 | https://thesession.org/recordings/1215 |
| http://www.wikidata.org/entity/Q7417356 | https://thesession.org/recordings/1606 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/1664 |
| http://www.wikidata.org/entity/Q6183761 | https://thesession.org/recordings/1704 |

4. Queries across TheSession, MusicBrainz and Wikidata



- Q4-1: How many / Which recording performers on TheSession (who) are **not on MusicBrainz**?
- Q4-2: List **top five most popular record labels** used by musicians on TheSession?
- Q2-3: List all albums **recorded by bands** in TheSession, that was **released in 2003 in America**

Prompting ChatGPT to generate SPARQL query (Q4-1-1)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given RDF graphs <http://sample/thesession/reconciled> and <http://musicbrainz/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible,

so regarding SPARQL queries across different databases, we use

(1) `graph <http://sample/thesession/reconciled>` with:

wdt:P175 rdfs:range ex:performer.

(2) `graph <http://musicbrainz/reconciled>` with:

wdt:P2888 rdfs:range ex:performer.

[note: DISTINCT; The value of performer must be IRI, that should be embodied earlier in SPARQL, please use FILTER isIRI()]

Make sure to generate a correct SPARQL for the following question:

How many recording performers on TheSession are **not on MusicBrainz**?

Generated SPARQL by ChatGPT (Q4-1-1)

How many recording performers on TheSession are not on MusicBrainz?

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
SELECT (COUNT(DISTINCT ?performer)
```

```
AS ?performerCount) WHERE {
```

```
  GRAPH <http://sample/thesession/reconciled>
```

```
    {?recording wdt:P175 ?performer .
```

```
    FILTER isIRI(?performer)
```

```
  }
```

```
  FILTER NOT EXISTS {
```

```
    GRAPH <http://musicbrainz/reconciled> {
```

```
      ?otherRecording wdt:P2888 ?performer .
```

```
      FILTER isIRI(?performer)
```

```
    }}
```

The result from Virtuoso (Q4-1-1)

SPARQL | HTML5 table

performerCount

15

Note: Why is the result such a small number? This is because only a small fraction of artists/performers from TheSession have been reconciled with Wikidata

Prompting ChatGPT to generate SPARQL query (Q4-1-2)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given RDF graphs <http://sample/thesession/reconciled> and <http://musicbrainz/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible,

so regarding SPARQL queries across different databases, we use

(1) `graph <http://sample/thesession/reconciled>` with:

wdt:P175 rdfs:range ex:performer.

(2) `graph <http://musicbrainz/reconciled>` with:

wdt:P2888 rdfs:range ex:performer.

[note: DISTINCT; The value of ex:performer must be IRI, that should be embodied in SPARQL]

Make sure to generate a correct SPARQL for the following question:

Which recording performers on TheSession **are not on MusicBrainz?**

Generated SPARQL by ChatGPT (Q4-1-2)

Which recording performers on TheSession are not on MusicBrainz?

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
SELECT DISTINCT ?performer
```

```
WHERE {
```

```
  GRAPH <http://sample/thesession/reconciled> {
```

```
    ?recording wdt:P175 ?performer .
```

```
  }
```

```
  MINUS {
```

```
    GRAPH <http://musicbrainz/reconciled> {
```

```
      ?otherRecording wdt:P2888 ?performer .
```

```
    }}
```

```
  FILTER(isIRI(?performer)) }
```

The result from Virtuoso (Q4-1-2)

SPARQL | HTML5 table

<http://www.wikidata.org/entity/Q101325026>

<http://www.wikidata.org/entity/Q121384360>

<http://www.wikidata.org/entity/Q2198713>

<http://www.wikidata.org/entity/Q2286403>

<http://www.wikidata.org/entity/Q2784150>

<http://www.wikidata.org/entity/Q3418793>

<http://www.wikidata.org/entity/Q35489573>

<http://www.wikidata.org/entity/Q3860276>

<http://www.wikidata.org/entity/Q539531>

<http://www.wikidata.org/entity/Q594693>

<http://www.wikidata.org/entity/Q7171905>

<http://www.wikidata.org/entity/Q7417356>

<http://www.wikidata.org/entity/Q7490578>

<http://www.wikidata.org/entity/Q7557090>

<http://www.wikidata.org/entity/Q93448268>

Prompting ChatGPT to generate SPARQL query (Q4-2)

Enter the following in ChatGPT (**the text in red will be provided to users**):

Given RDF graphs <http://sample/thesession/reconciled> and <http://musicbrainz/reconciled>, the properties and classes of instances are reconciled with Wikidata as much as possible.

1. For graph <http://sample/thesession/reconciled/noLangLabel>:

wd:Q482994 a owl:Class; rdfs:label "Record". wdt:P2561 rdfs:label "name".

2. For graph <http://musicbrainz/reconciled>

wd:Q3972943 a owl:Class; rdfs:label "Record".

wdt:P1476 rdfs:label "title"; rdfs:domain wd:Q3972943.

wdt:P264 rdfs:label "record label"; rdfs:domain wd:Q3972943; rdfs:range wd:Q18127.

wdt:P2561 rdfs:domain wd:Q18127; rdfs:comment "It's the name of 'record label'. **Don't use rdfs:label** as a predicate in the triple pattern of the generated SPARQL!"!!!

wd:Q18127 rdfs:label "Label".

Make sure to generate an absolutely correct SPARQL for the following question

(Note: Primarily use the properties or classes mentioned above for each graph. You can use wdt:P2561 of wd:Q482994 to match wdt:P1476 of wd:Q3972943. Return both the URI of the record labels and the name of record labels): *List top five most popular record labels used by musicians on TheSession?* ->**deceptive, Modification:**

List the **top five most popular labels from MusicBrainz** whose records are also on TheSession database

Generated SPARQL by ChatGPT (Q4-2)

List the **top five most popular labels from MusicBrainz** whose records are also on TheSession database.



List top five most popular record labels used by musicians on TheSession?

```
PREFIX wd: <http://www.wikidata.org/entity/>
```

```
PREFIX wdt: <http://www.wikidata.org/prop/direct/>
```

```
PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>
```

```
SELECT ?labelURI ?labelName (COUNT(?record) AS ?recordCount) WHERE {
```

```
  # Records from TheSession
```

```
  GRAPH <http://sample/thesession/reconciled/noLangLabel> { ?record a wd:Q482994.
```

```
    ?record wdt:P2561 ?recordName. }
```

```
  # Matching Records from MusicBrainz
```

```
  GRAPH <http://musicbrainz/reconciled> { ?musicBrainzRecord a wd:Q3972943.
```

```
    ?musicBrainzRecord wdt:P1476 ?recordName.
```

```
  # Link records to their labels
```

```
    ?musicBrainzRecord wdt:P264 ?labelURI. ?labelURI a wd:Q18127. ?labelURI wdt:P2561 ?labelName. }}
```

```
GROUP BY ?labelURI ?labelName ORDER BY DESC(?recordCount) LIMIT 5
```

The result from Virtuoso (Q4-2)



SPARQL | HTML5 table

| labelURI | labelName | recordCount |
|---|----------------------|-------------|
| https://musicbrainz.org/label/011d1192-6f65-45bd-85c4-0400dd45693e | "Columbia" | 62 |
| https://musicbrainz.org/label/8f638ddb-131a-4cc3-b3d4-7ebdac201b55 | "Epic" | 49 |
| https://musicbrainz.org/label/1b3b52a5-ef89-4f8b-8d5f-e15a7a58863b | "Vertigo" | 33 |
| https://musicbrainz.org/label/29d7c88f-5200-4418-a683-5c94ea032e38 | "BMG" | 29 |
| https://musicbrainz.org/label/da631842-b2b2-4b65-a526-592f1b776c18 | "Supersonic Records" | 28 |

Prompting ChatGPT to generate SPARQL query (Q4-3)

Enter the following in ChatGPT (the text in red will be provided to users):

Given 2 RDF graphs, the properties, instances, classes of which are reconciled with Wikidata as much as possible, regarding SPARQL queries across different databases, refer to:

album **wd:Q482994** **owl:equivalentClass** **wd:Q3972943**. The values of wdt:P175 have property wdt:P31 to indicate whether the performer is a band. **release**

1. For `GRAPH <http://sample/thesession/reconciled/noLangLabel>`:

wd:Q482994 a owl:Class; rdfs:label "Album".

wdt:P175 rdfs:label "performer"; rdfs:domain wd:Q482994. wdt:P2561 rdfs:label "name".

2. For `GRAPH <http://musicbrainz/reconciled>`:

wd:Q3972943 a owl:Class; rdfs:label "Release".

wdt:P17 rdfs:label "country". wdt:P585 rdfs:label "date". wdt:P1476 rdfs:label "title".

3. For `SERVICE <https://query.wikidata.org/sparql>`:

wd:Q215380 rdfs:label "bands". wdt:P31

Make sure to generate a correct SPARQL for the following question[note: match value of P2561 and of P1476 in order to match the instances of wd:Q482994 and of wd:Q3972943(Please use `FILTER(?date=2003)`in SPARQL)]:

List all albums **recorded by bands** in TheSession, that was **released in 2003 in America**

Generated SPARQL by ChatGPT (Q4-3)

List all albums recorded by bands in TheSession, that was released in 2003 in America

PREFIX owl: <http://www.w3.org/2002/07/owl#> PREFIX rdfs: <http://www.w3.org/2000/01/rdf-schema#>

PREFIX wdt: <http://www.wikidata.org/prop/direct/> PREFIX wd: <http://www.wikidata.org/entity/>

SELECT ?albumName ?album ?performer

WHERE { GRAPH <http://sample/thesession/reconciled/noLangLabel> {

?album a wd:Q482994 ; #recording of TheSession

wdt:P2561 ?albumName ;

wdt:P175 ?performer . }

GRAPH <http://musicbrainz/reconciled> {

?release a wd:Q3972943 ;

wdt:P1476 ?releaseTitle ;

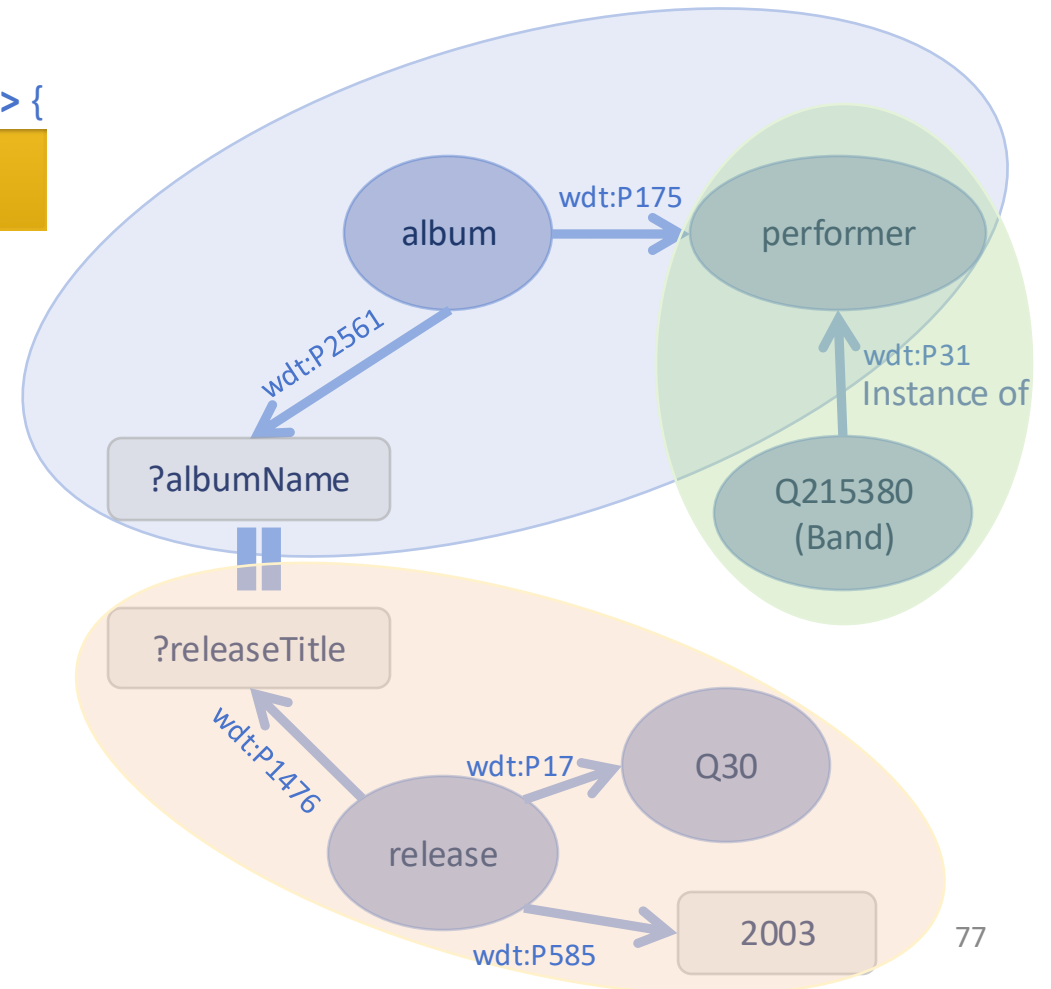
wdt:P17 wd:Q30 ; #

wdt:P585 ?date .

FILTER(?date = 2003)} FILTER(?albumName = ?releaseTitle)

SERVICE <https://query.wikidata.org/sparql> {

?performer wdt:P31 wd:Q215380 . }}



The result from Virtuoso (Q4-3)

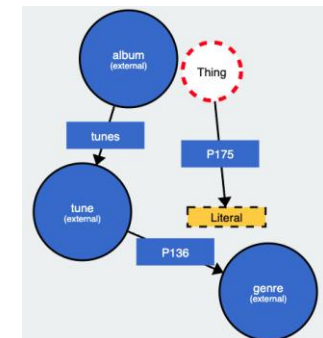
SPARQL | HTML5 table

| albumName | album | performer |
|---------------|---|---|
| "The Best Of" | https://thesession.org/recordings/245 | http://www.wikidata.org/entity/Q1410132 |

<https://thesession.org/recordings/245>

Discussion or Issue

- Should we or how to understand the ontology?
 - Ontology can be automatically generated during CSV2RDF in Virtuoso
 - Understanding OWL (Web Ontology Language) isn't essential right now
 - It can be substantially seen as a graph structure, serving as a framework for the DB
 - A familiarity of ontology will contribute to a more professional way of asking questions. Refer to:
 - Q1-1 "Find the sessions that took place in Greece." (Interoperable)
 - Q2-2 What genres appear in the album whose artist is "Adam Agee..."?
 - Q4-2 List top 5 most popular record labels used *by musicians* on TheSession?
 - Editing or visualization tools for ontology: (1) protégé (2) WebVOWL
- Issue: The accuracy of SPARQL generation
 - What if there is a complex context (e.g., an oversized ontology)?



Not a "black box"

Future Work

- Further reconciliation Q4-3
- Knowing more about Wiki-Project Music
(https://www.wikidata.org/wiki/Wikidata:WikiProject_Music)
- E.g. “Find albums containing “Tola” in its name or its artist’s name” wdt:P175 performer
- Prompt engineering or In-context learning:
 - Intelligent agents designing and RAGs
 - E.g., ChatGPT can even read the visualization of OWL
- Knowledge-Reasoning, Recommendation, Visualization, Network Analysis... Q4-2 if ... a lazy searcher, asking natural language queries in a ...

Thank you!

Junjun Cao
Postdoctor of DDMAL
Email: junjun.cao@mail.mcgill.ca