Task 1 Apache Jena

1.1

> Get names of actors whose name starts with the letter 'T', and the names of films they starred in (total of 2 columns).

Execution result:

```
15  PREFIX lmdb_country: <http://data.linkedmdb.org/country/>
16
17  SELECT DISTINCT ?actor_name ?movie_name
18 ▼ WHERE {
19      ?movie a lmdb:film ;
20          lmdb:actor/lmdb:actor_name ?actor_name ;
21          dcterms:title ?movie_name .
22      FILTER regex(?actor_name, "^T")
23  }
24
25  LIMIT 10
26
27
28
```

QUERY RESULTS

[Table] [Raw Response] [↓]

Showing 1 to 10 of 10 entries   Search: [_____] Show [50] entries

	actor_name	movie_name
1	"Timothy Dalton"	"Flash Gordon"
2	"Tito Larriva"	"True Stories"
3	"Troy Garity"	"Sunshine"
4	"Taylor Hanson"	"Strong Enough to Break"
5	"Theo Sebeko"	"Kirikou et la sorciÃ¨re"
6	"The B-52's"	"The B-52's 1979-1989"
7	"The B-52's"	"The B-52's Time Capsule: Videos for a Future Generation 1979-1998"
8	"Teresa Wright"	"Somewhere in Time"
9	"Tom Courtenay"	"King Rat"
10	"Tom Hollander"	"Enigma"

Showing 1 to 10 of 10 entries

1.2

> Get URIs and names of **actors** who are also **writers** but are not **producers** (total of 2columns).

While I'm pretty sure different people can have duplicate names, there's no way I can differentiate between them. So for people with the same name, I can only assume it's the same person.

Execution result:

```
15   PREFIX lmdb_country: <http://data.linkedmdb.org/country/>
16
17   SELECT ?actor_uri ?actor_name
18 ▼ WHERE {
19       ?actor_uri a lmdb:actor ;
20           lmdb:actor_name ?actor_name .
21       ?writer_uri a lmdb:writer ;
22           lmdb:writer_name ?actor_name .
23 ▼   FILTER NOT EXISTS { ?_ a lmdb:producer; lmdb:producer_name ?actor_name. }
24   }
25
26   LIMIT 10
27
28
```

QUERY RESULTS

Table | Raw Response

Showing 1 to 10 of 10 entries

Search: [   ]   Show 50 entries

	actor_uri	actor_name
1	lmdb_actor:2	"Chris Morris"
2	lmdb_actor:5	"P. J. O'Rourke"
3	lmdb_actor:12	"Willis O'Brien"
4	lmdb_actor:25	"Graham Greene"
5	lmdb_actor:58	"Edward F. Cline"
6	lmdb_actor:64	"Dennis O'Keefe"
7	lmdb_actor:81	"Carroll O'Connor"
8	lmdb_actor:88	"Jerry O'Connell"
9	lmdb_actor:106	"Robert Taylor"
10	lmdb_actor:114	"Paul Gross"

Showing 1 to 10 of 10 entries

1.3

Get names of actors who starred in more than 18 films with the same director (who is not the actor). Show the name of the director and the number of films as well (total of 3 columns).

Execution result:

```
34              lmdb:director ?film_director .
35          ?film_director lmdb:director_name ?director_name .
36          FILTER (?actor_name != ?director_name)
37      }

38
39      GROUP BY ?actor_uri ?film_director
40      HAVING (?count > 18)
41    }
42    ?actor_uri lmdb:actor_name ?actor_name .
43    ?film_director lmdb:director_name ?director_name .
44
45  }
46
47  LIMIT 10
```

QUERY RESULTS

[ ] Table Raw Response [↓]

Showing 1 to 5 of 5 entries   Search: [   ]   Show 50 [↕] entries

actor_name	director_name	count
1	"Kenneth Williams"	"Gerald Thomas"	"26"^^xsd:integer
2	"Edna Purviance"	"Charlie Chaplin"	"36"^^xsd:integer
3	"Moe Howard"	"Jules White"	"23"^^xsd:integer
4	"Joan Sims"	"Gerald Thomas"	"19"^^xsd:integer
5	"Stan Laurel"	"James Parrott"	"21"^^xsd:integer

Showing 1 to 5 of 5 entries

1.4

> Get the names of countries, the number of comedy films (films with genre "Comedy") and the population (in millions) for each country. The results should be sorted by the number of comedy films in descending order (total of 3 columns).

I only counted the number of `comedy` movies, other kind of comedy like `romantic comedy`, `Screwball comedy`, etc. are not included.

There are more than 200 countries or regions in `lmdb.nt`, but only a couple dozen countries have comedy movies. Since we need "number of comedy films", I will not ouput country without a comedy movie.

Execution result:

```
25
26 ▼ WHERE {
27        ?film_uri a lmdb:film ;
28             lmdb:genre lmdb_film_genre:71 ; # comedy
29             lmdb:country ?movie_coutry .
30        ?movie_coutry lmdb:country_name ?country_name ;
31        lmdb:country_population ?country_population .
32        BIND(xsd:integer(?country_population) / 1000000 AS ?million_people)
33
34
35  }
36  GROUP BY ?country_name ?million_people
37  ORDER BY DESC(?comedy_count)
38  LIMIT 10
```

QUERY RESULTS

Table   Raw Response

Showing 1 to 10 of 10 entries

Search: _____   Show 50 entries

country_name	comedy_count	million_people
1	"United States"	"692"^^xsd:integer	"303.824"^^xsd:decimal
2	"United Kingdom"	"109"^^xsd:integer	"60.943"^^xsd:decimal
3	"Canada"	"93"^^xsd:integer	"33.679"^^xsd:decimal
4	"France"	"29"^^xsd:integer	"64.094"^^xsd:decimal
5	"Australia"	"17"^^xsd:integer	"20.6"^^xsd:decimal
6	"Italy"	"16"^^xsd:integer	"58.145"^^xsd:decimal
7	"Hong Kong"	"13"^^xsd:integer	"6.898686"^^xsd:decimal
8	"Japan"	"13"^^xsd:integer	"127.288"^^xsd:decimal
9	"Germany"	"11"^^xsd:integer	"82.369"^^xsd:decimal
10	"Spain"	"7"^^xsd:integer	"40.491"^^xsd:decimal

Showing 1 to 10 of 10 entries

1.5

> Get the names of female actors who were born after 1970 but before 1980. Show their date of birth (total of 2 columns).

> • Note that you do not have the data about gender and date-of-birth for actors in the provided graph. You will need to retrieve these values from an additional SPARQL endpoint using a federated query extension (the SERVICE keyword, read more about it here: https://www.w3.org/TR/sparql11-federated-query/).

Execution result:

```
19  PREFIX lmdb_producer: <http://data.linkedmdb.org/producer/>
20  PREFIX lmdb_writer: <http://data.linkedmdb.org/writer/>
21  PREFIX myns: <http://myns.com/>
22  PREFIX dbo: <http://dbpedia.org/ontology/>
23  prefix dbp: <http://dbpedia.org/property/>
24
25
26  SELECT DISTINCT ?actor_name ?birthDate
27 ▼ WHERE {
28 ▼     service <http://dbpedia.org/sparql> {
29          SELECT DISTINCT ?actor_DB_uri ?birthDate ?gender ?DBname
30 ▼         WHERE {
31              ?actor_DB_uri a dbo:Actor ;
32                  foaf:gender ?gender ;
```

QUERY RESULTS

Table | Raw Response | ⬇

Showing 1 to 10 of 10 entries

Search: Show 50 entries

actor_name	birthDate
1 "Rosanne Wong"	"1979-06-30"^^xsd:date
2 "Shu Qi"	"1976-04-16"^^xsd:date
3 "Anne Heung"	"1974-10-02"^^xsd:date
4 "Mavis Fan"	"1977-02-27"^^xsd:date
5 "Sonija Kwok"	"1974-07-22"^^xsd:date
6 "Zhang Ziyi"	"1979-02-09"^^xsd:date
7 "Zhou Xun"	"1974-10-18"^^xsd:date
8 "Gigi Leung"	"1976-03-25"^^xsd:date
9 "Josie Ho"	"1974-12-26"^^xsd:date
10 "Miriam Yeung"	"1974-02-03"^^xsd:date

Showing 1 to 10 of 10 entries

Task 2 Wikidata

2.1

> Get URIs and names of actors who are also musicians and were part of (P361) a band
> sometime between 1960 to 2010. Also, show their date of birth. If an actor doesn't have date
> of birth, the value can be null, empty string, or missing (total of 3 columns).

My approach is limited by the information that provided by Wikidata, in that I am directly pulling
the artist's time joined in a band from the artist page , If Wikidata only provides the start and end
dates of the band, but not when the artist joined, then I will ignore the artist.

Complying with the requirements here, if the time is exactly 1960 or exactly 2010, I did not count.

I've noticed that certain artists have more than one `date of birth` information. For this kind of
artist, there will be more than one record that appears in the results.

Execution result:

```
1   SELECT DISTINCT ?actor ?actorLabel ?DOB
2   WHERE {
3     ?actor wdt:P106/wdt:P279* wd:Q33999 . # P:occupation/subClassOf O:actor
4     ?actor wdt:P106/wdt:P279* wd:Q639669 . # P:occupation/subClassOf O:musicians
5
6     ?actor p:P361 ?statement . # part of
7     ?statement ps:P361 ?band . # part of
8     ?statement pq:P580 ?starttime.
9     ?statement pq:P582 ?endtime.
10
11    ?band wdt:P31/wdt:P279* wd:Q2088357 ; # P:instance of / subclass of O:musical ensemble
12
13    FILTER(YEAR(?starttime)>1960 && YEAR(?endtime)<2010)
14
15    OPTIONAL {
16      ?actor wdt:P569 ?DOB .
17    }
18
19    SERVICE wikibase:label { bd:serviceParam wikibase:language "en". }
20
21  }
22
23  LIMIT 10
24
25
```

10 results in 8418 ms   </> Code   ⬇ Download ▾   🔗 Link ▾

actor	actorLabel	DOB
🔍 wd:Q3983826	Teo Teocoli	25 February 1945
🔍 wd:Q265179	Rita Lee	31 December 1947
🔍 wd:Q2632	Ringo Starr	7 July 1940
🔍 wd:Q364881	Trevor Rabin	13 January 1954
🔍 wd:Q3752129	Franz Di Cioccio	21 January 1946
🔍 wd:Q3752129	Franz Di Cioccio	20 January 1946
🔍 wd:Q6793241	Yuval Dor	10 December 1950
🔍 wd:Q2599	Paul McCartney	18 June 1942
🔍 wd:Q36153	Beyoncé	4 September 1981
🔍 wd:Q6971687	Ami Mandelman	24 August 1950

2.2

> Get URIs and names of films that include at least 4 cast members (P161) who have been nominated after the year 2000 for the Academy Award for Best Actor (Q103916). Also, show the number of such actors for each film (total of 3 columns).

Not all `point of time` qualifier are provided under the "nominated for " property. Each Academy Awards instance did not provide nomination information either. Thus the number of my query result ~~my~~ may lower than reality.

Execution result:

```
1  SELECT ?film ?name (?num AS ?film_num)
2
3  WHERE {
4      {
5          SELECT ?film (COUNT(?cast_member) AS ?num)
6          WHERE {
7              ?film wdt:P31 wd:Q11424 ; # instance of ; film
8                    wdt:P161 ?cast_member .   # cast member
9
10             ?cast_member p:P1411 ?statement . # nominated for
11             ?statement ps:P1411 wd:Q103916 .  # nominated for ; Academy Award for Best Ac
12             ?statement pq:P585 ?nom_time.  # point in time
13             FILTER (YEAR(?nom_time) > 2000).
14         }
15         GROUP BY ?film
16         HAVING (?num >= 4)
17     }
18     SERVICE wikibase:label { bd:serviceParam wikibase:language "en" .
19                              ?film rdfs:label ?name .}
20 }
21
```

8 results in 936 ms   </> Code   ⬇ Download ▾   𝒮 Link ▾

film	name	film_num
🔍 wd:Q25394556	Touch of Evil	4
🔍 wd:Q7756638	The People Speak	5
🔍 wd:Q681962	Tinker Tailor Soldier Spy	4
🔍 wd:Q969828	Welcome to Hollywood	4
🔍 wd:Q208108	Catch Me If You Can	4
🔍 wd:Q660894	What's Eating Gilbert Grape	4
🔍 wd:Q1392744	The Wolf of Wall Street	5
🔍 wd:Q9013673	American Hustle	4